Kurdistan Journal of Applied Research (KJAR)
Print-ISSN: 2411-7684 | Electronic-ISSN: 2411-7706

Original Article

Check for updates

A Novel Conjugate Gradient Algorithm as a Convex Combination of Classical Conjugate Gradient Methods

Sara Sahib Mohammed Zaki [a*], **Hawraz Nadhim Jabbar** [b], **Sozan Saber Haider** [c]

[a] Network Department, Computer Science Institute, Sulaimani Polytechnic University, Sulaymaniyah, Iraq.
[b] Mathematics Department, College of Science, Kirkuk University, Kirkuk, Iraq.
[c] Statistics and Informatics Department, College of Administration and Economics, University of Sulaimani, Sulaymaniyah, Iraq.

Submitted: 21 January 2025
Revised: 1 March 2025
Accepted: 25 March 2025

*** Corresponding Author**:
sara.sahib@spu.edu.iq

Keywords: Unconstrained optimization, Hybrid conjugate Gradient, Convex combination, Global convergence, Sufficient descent condition.

How to cite this paper: S. S. Mohammed Zaki, H. N. Jabbar, S. S. Haider," A Novel Conjugate Gradient Algorithm as a Convex Combination of Classical Conjugate Gradient Methods", KJAR, vol. 10, no. 1, pp: 83-98, June 2025, doi: 10.24017/science.2025.1.6

Abstract: Conjugate gradient (CG) algorithms are constructive for handling large-scale nonlinear optimization problems. One optimization technique intended to address unconstrained optimization issues effectively is the hybrid conjugate gradient algorithm. The hybrid conjugate gradient algorithm aims to improve convergence properties while keeping computations simple by merging features from other conjugate gradient techniques. In this paper, a new hybrid conjugate gradient algorithm is proposed and analyzed, which is obtained as a convex combination of the Dai-Yuan, Hestenes-Stiefel and Harger-Zhan conjugate gradient methods. The primary objective is to improve convergence efficiency and computational performance. The proposed algorithm is designed to reduce the number of iterations and computational costs compared to traditional CG methods. Numerical experiments on standard unconstrained optimization criteria show that the hybrid method achieves faster convergence, often requiring much fewer iterations to reach a specified gradient norm tolerance or objective function value. Additionally, the per-iteration computational cost remains competitive, as the convex combination framework introduces minimal overhead. Theoretical analysis proves the global convergence of the algorithm under standard assumptions. The results highlight the superior performance of the hybrid method in terms of the number of iterations and the total computational cost, especially for large-scale and unconditional problems. This work advances the development of efficient and robust CG algorithms, offering a practical solution for unconstrained optimization challenges.

1. Introduction

Optimization has witnessed significant advancements in recent years, particularly in developing efficient algorithms for solving large-scale unconstrained optimization problems. Among these, conjugate gradient (CG) methods have emerged as a cornerstone due to their simplicity, low memory requirements, and strong convergence properties [1]. These methods are particularly well-suited for problems where the objective function is smooth and convex, making them indispensable in various scientific and engineering applications, including machine learning, computational physics, and numerical analysis [2].

The classical conjugate gradient methods, such as the Fletcher-Reeves (FR), Polak-Ribière (PR) and Hestenes-Stiefel (HS) algorithms, have been extensively studied and applied [3]. Each method has its strengths and weaknesses, often leading to varying performance depending on the problem. For instance, the FR method is known for its global convergence properties, while the PR and HS methods often exhibit faster convergence in practice but may suffer from numerical instability. Recent research has focused on enhancing the performance of these classical methods by exploring new strategies that combine their desirable features [4]. One promising approach is the convex combination of different CG methods, which aims to leverage each method's strengths while mitigating their limitations. This approach has led to development of novel algorithms that exhibit improved convergence rates and robustness across a broader range of problems.

Unconstrained optimization minimizes an objective function that depends on real variables without any limitations or restrictions on the values these variables can take [5]. Essentially, it involves solving general optimization problems where no constraints are imposed on the permissible range of the variables [6]; consider the following unconstrained optimization problem:

$$\min\{f(x), x \in R^n\} \qquad (1)$$

where $f: R^n \longrightarrow R$ is a continuously differentiable function; the authors define the gradient as $g_k = \nabla f(x_k)$ The conjugate gradient methods are among the best optimization techniques for solving large-scale problems.

Generally, to solve this problem, starting from an initial point $x_0 \in R^n$ [7], a conjugate gradient algorithm generates a sequence of points $\{x_k\}$ as follows:

$$x_{k+1} = x_k + \alpha_k d_k \qquad (2)$$

Where α_k is the step size selected by using line search and the directions d_k are generated as:

$$d_{k+1} = -g_{k+1} + \beta_k d_k, \quad d_0 = -g_0 \qquad (3)$$

β_k is known as the conjugate gradient coefficient; the different choices for this coefficient correspond to different conjugate gradient methods. Some of these methods are: HS [8], FR [3], PRP [9], CD [10], LS [11], DY [12], and HZ [13]:

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

$$\beta_k^{LS} = \frac{g_{k+1}^T y_k}{-g_k^T d_k} \qquad \beta_k^{DY} = \frac{||g_{k+1}||^2}{y_k^T d_k} \qquad \beta_k^{HZ} = \frac{g_{k+1}^T y_k}{d_k^T y_k} - 2\frac{||y_k||^2}{d_k^T y_k}\frac{g_{k+1}^T d_k}{d_k^T y_k}$$

where $y_k = g_{k+1} - g_k$, and $\|.\|$ denotes the Euclidean norm.

In this paper, authors use strong Wolfe line search, which is determined by the subsequent criteria [14]:

$$f(x_k + \alpha_k d_k) - f(x_k) \le \rho \alpha_k g_k^T d_k \qquad (4)$$

$$\sigma g_k^T d_k \le g_{k+1}^T d_k \le -\sigma g_k^T d_k \qquad (5)$$

Where $0 < \rho < \sigma < 1$, the hybrid algorithm is a fundamental class of conjugate gradient techniques. Moreover, because hybrid schemes capitalize on the factors that make them up, they outperform standard conjugate gradient approaches regarding computational performance and have more reliable convergence characteristics.

Academics are interested in hybrid or mixed conjugate gradient approaches. For instance, Andrei [15] proposed a new hybrid conjugate gradient method based on the convex combination of HS and DY which is defined as:

$$\beta_k^C = (1-\theta_k)\beta_k^{HS} + \theta_k \beta_k^{DY} \qquad \textit{where} \qquad 0 \le \theta_k \le 1$$

In addition, Sabrina et al [16], proposed the following hybrid method:

$$\beta_k^{hyb} = \alpha_k\beta_k^{FR} + \theta_k\beta_k^{PRP} + (1-\alpha_k-\theta_k)\beta_k^{DY} \text{ where } 0 \le \theta_k \le 1$$

Furthermore, Abbo and Hameed [6] suggested a new hybrid method which is defined as follows:

$$\beta_k^{NK1} = \Upsilon\beta_k^{LS} + (1-\Upsilon)\beta_k^{DY}$$

This paper proposes a new hybrid conjugate gradient method based on a convex combination of the DY, HS, and HZ conjugate gradient algorithms to solve unconstraint optimization problems.
The remainder of this paper is organized as follows: under section 2, the authors introduce some previously proposed approaches, present the hybrid conjugate gradient method, and obtain the parameters δ and γ through various techniques. The authors demonstrate that under mild conditions, the chosen method with the Wolfe line search produces directions that meet the sufficient descent criteria, the algorithm will be proposed, and the newly chosen method's descent condition and convergence features will be analyzed. The authors obtained results after implementing some functions presented in section 3, and numerical analysis was performed, and section 4 provides analysis of the results and significance of this study along with the study's limitations, and final concluding remarks are presented in section 5.

2. Material and Method

2.1. *Proposed Method*

In this paper, a convex combination of the DY, HS and HZ conjugate gradient algorithms is proposed. The conjugate gradient parameter:

$$\beta_k^{hSH} = \delta_k\beta_k^{DY} + \Upsilon_k\beta_k^{HS} + (1-\delta_k-\Upsilon_k)\beta_k^{HZ} \tag{6}$$

Consequently, the direction d_k is given by:

$$d_{k+1} = -g_{k+1} + \beta_k d_k, \qquad d_0 = -g_0 \tag{7}$$

The parameters δ_k, Υ_k satisfying $0 \le \delta_k, \Upsilon_k \le 1$ are selected, as explained later. Note that

- If $\delta_k = 1$ and $\Upsilon_k = 0$, then $\beta_k^{hSH} = \beta_k^{DY}$
- If $\delta_k = 0$ and $\Upsilon_k = 1$, then $\beta_k^{hSH} = \beta_k^{HS}$
- If $\delta_k = 0$ and $\Upsilon_k = 0$, then $\beta_k^{hSH} = \beta_k^{HZ}$
- If $\delta_k = 0$ and $0 < \Upsilon_k < 1$, then $\beta_k^{hSH} = \Upsilon_k\beta_k^{HS} + (1-\Upsilon_k)\beta_k^{HZ}$ which is convex combination of β_k^{HS} and β_k^{HZ}
- If $\Upsilon_k = 0$ and $0 < \delta_k < 1$, then $\beta_k^{hSH} = \delta_k\beta_k^{DY} + (1-\delta_k)\beta_k^{HZ}$ which is convex combination of β_k^{DY} and β_k^{HZ} [17]
- If $(1-\delta_k-\Upsilon_k) = 0$ and $0 \le \delta_k, \Upsilon_k \le 1$, then $\Upsilon_k = 1-\delta_k$, hence $\beta_k^{hSH} = \delta_k\,\beta_k^{DY} + (1-\delta_k)\,\beta_k^{HS}$ which is convex combination of β_k^{DY} and β_k^{HS} [18]
- If $\delta_k \in (0,1)$, $\Upsilon_k \in (0,1)$ and $0 < \delta_k + \Upsilon_k < 1$, then a new hybrid conjugate gradient method is obtained as a convex combination of DY, HS and HZ.

From 6 and 7 it is evident that:

$$d_k = \begin{cases} -g_k & \text{if } k = 0 \\ -g_{k+1} + \delta_k \frac{\|g_{k+1}\|^2}{d_k^T y_k} d_k + \Upsilon_k \frac{g_{k+1}^T y_k}{d_k^T y_k} d_k + (1-\delta_k-\Upsilon_k)\left(\frac{g_{k+1}^T y_k}{d_k^T y_k} - 2\frac{\|y_k\|^2}{d_k^T y_k}\frac{g_{k+1}^T d_k}{d_k^T y_k}\right) d_k, & \text{if } k \ge 1 \end{cases} \tag{8}$$

The conventional conjugacy requirement is applied to select the parameters δ_k, Υ_k that is $d_{k+1}^T y_k = 0$ thus, the following lemma is obtained:
Lemma 1: [6] if the condition $d_{k+1}^T y_k = 0$ is satisfied at each iteration, then:

$$1-\delta_k = \frac{2\|y_k\|^2 g_{k+1}^T d_k (\Upsilon_k + 1)}{-g_{k+1}^T g_k d_k^T y_k - 2\|y_k\|^2 g_{k+1}^T d_k} \qquad 0 < \Upsilon_k < 1 \tag{9}$$

2- $\Upsilon_k = (\frac{g_k\, y_k}{2\|y_k\|^2} - 1)\,\delta_k$ $\quad 0 < \delta_k < 1 \quad$ (10)

Proof 1: from (8):

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

By multiplying both sides by y_k:

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

If $d_{k+1}^T y_k = 0$

$$0 = -g_{k+1}^T y_k + \delta_k \|g_{k+1}\|^2 + \Upsilon_k g_{k+1}^T y_k + (1-\delta_k-\Upsilon_k)\left(g_{k+1}^T y_k - 2\|y_k\|^2 \frac{g_{k+1}^T d_k}{d_k^T y_k}\right)$$

Following some algebraic computation, the above equation becomes:

$$\delta_k = \frac{2\|y_k\|^2 g_{k+1}^T d_k (\Upsilon_k + 1)}{-g_{k+1}^T g_k d_k^T y_k - 2\|y_k\|^2 g_{k+1}^T d_k} \qquad 0 < \Upsilon_k < 1$$

The parameter Υ can be outside [0,1] then:

- If $\delta_k < 0$ then $\delta_k = 0$ is assumed
- If $\delta_k > 1$ then $\delta_k = 1$ is assumed
- If $\delta_k + \Upsilon_k \geq 1$ then $\delta_k + \Upsilon_k = 1$ is assumed

Proof 2: from 9 and after some algebra similar to part 1:

$$\Upsilon_k = \frac{(-g_{k+1}^T g_k d_k^T y_k - 2\|y_k\|^2 g_{k+1}^T d_k)\delta_k + 2\|y_k\|^2 g_{k+1}^T d_k}{2\|y_k\|^2 g_{k+1}^T d_k} \qquad 0 < \delta_k < 1$$

The parameter Υ_k can be outside [0,1] then:

- If $\Upsilon_k < 0$ then $\Upsilon_k = 0$ is assumed
- If $\Upsilon_k > 1$ then $\Upsilon_k = 1$ is assumed
- If $\delta_k + \Upsilon_k \geq 1$ then $\delta_k + \Upsilon_k = 1$ is assumed

2.2. *Algorithm (hSSH1) and (hSH)*[1]

Step 1: Initialization: given $x_0 \in R^n$ and the parameters $0 < \rho < \sigma < 1$, compute: $f(x_0)$, $g_0 = \nabla f(x_0)$. Consider $d_0 = -g_0$, set the initial guess is $\Upsilon_k = 0.5$ *or* $\delta_k = 0.5$

Step 2: If $\|g_k\| \leq 10^{-6}$, then stop. Else go to next.

Step 3: Calculate the step size α_k under strong Wolfe condition (4) and (5)

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

Step 5: Compute δ_k as in equation (9) or Υ_k as in equation (10)

Step 6: Calculate β_k^{hSH} as in equation (6)

Step 7: Search direction $d_{k+1} = -g_{k+1} + \beta_k^{hSH} d_k$ if the restart criterion of Powell $|g_{k+1}^T g_k| \geq 0.2\|g_{k+1}\|^2$ is satisfied, then restart, i.e., set $d_{k+1} = -g_{k+1}$ otherwise define $d_{k+1} = d$

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

2.3. *Sufficient Descent Condition and Convergence*

2.3.1. *Sufficient Descent Condition*

To demonstrate that the proposed method satisfies the sufficient descent condition, the following assumptions are needed:

Assumption 1 [19]: the level set $T = \{x \in R^n : f(x) \leq f(x_0)\}$ is bounded, i.e., there is a constant $B > 0$ such that $\|x\| \leq B$ for all $x \in T$ (11)

---

[1] (hSSH1) and (hSH) denote hybrid algorithms named after the contributing authors: Sara, Sozan, and Hawraz for hSSH1; Sara and Hawraz for hSH.

Assumption 2 [20]: in a neighborhood N of T, f is continuously differentiable and its gradient is Lipschitz continuous, i.e., $\exists L \geq 0$ such that $\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|$ for all $x, y \in N$ (12)

According to the assumption (1) and (2) on f, there is a constant $\tau \geq 0$ such that:
$\|\nabla f(x)\| \leq \tau$ for all $x \in T$
The search direction determined by the novel approach meets sufficient descent criterion, as demonstrated by the following theorem:

Theorem 1 [21]: let $\{ g_k \}$ and $\{d_k\}$ be generated by the proposed method, then d_k satisfies the sufficient descent condition:

$$g_k^T d_k \leq -c\|g_k\|^2 \text{ for all } k \geq 0, c > 0 \quad (13)$$

Proof: by applying mathematical induction, the fact that the search direction should satisfy the following descent condition is demonstrated:
When $k = 0$ that is $d_0 = -g_0$
Hence $g_0^T d_0 = -\|g_0\|^2$

The condition is satisfied when $k = 0$

Now if $k \geq 1$:

$$d_{k+1} = -g_{k+1} + \beta_k^{hSH} d_k$$
$$= -g_{k+1} + (\delta_k \beta_k^{DY} + \Upsilon_k \beta_k^{HS} + (1 - \delta_k - \Upsilon_k)\beta_k^{HZ})d_k$$
$$= -(\delta_k g_{k+1} + \Upsilon_k g_{k+1} + (1 - \delta_k - \Upsilon_k)g_{k+1}) + (\delta_k \beta_k^{DY} + \Upsilon_k \beta_k^{HS} + (1 - \delta_k - \Upsilon_k)\beta_k^{HZ})d_k$$

After some algebraic computations:

$$d_{k+1} = \delta_k d_{k+1}^{DY} + \Upsilon_k d_{k+1}^{HS} + (1 - \delta_k - \Upsilon_k)d_{k+1}^{HZ} \quad (14)$$

By multiplying both sides by g_{k+1}^T:

$$g_{k+1}^T d_{k+1} = \delta_k g_{k+1}^T d_{k+1}^{DY} + \Upsilon_k g_{k+1}^T d_{k+1}^{HS} + (1 - \delta_k - \Upsilon_k)g_{k+1}^T d_{k+1}^{HZ} \quad (15)$$

The seven cases are proven as follows:
Case 1 [12]: if $\delta_k = 1, \Upsilon_k = 0$, then

$g_{k+1}^T d_{k+1} = g_{k+1}^T d_{k+1}^{DY}$ the sufficient descent condition for DY has to be proven:

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

Since $g_{k+1} d_k^T = 0$ then g_{k+1} is orthogonal to d_k
Hence

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

Then the sufficient descent condition becomes:

$$g_{k+1}^T d_{k+1}^{DY} = -\|g_{k+1}\|^2 \leq -c_1 \|g_{k+1}\|^2 \qquad c_1 \in (0,1) \quad (16)$$

Case 2 [8]: If $\delta_k = 0, \Upsilon_k = 1$

$g_{k+1}^T d_{k+1} = g_{k+1}^T d_{k+1}^{HS}$

The sufficient descent condition for HS has to be proven:

$g_{k+1}^T d_{k+1} \leq -c_2 \|g_{k+1}\|^2$ where c_2 is a small positive number

$$\beta_k^{HS} = \frac{g_{k+1}^T y_k}{y_k^T d_k}$$

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

Multiplying both sides by g_{k+1}^T

$g_{k+1}^T d_{k+1} = g_{k+1}^T (-g_{k+1} + \beta_k^{HS} d_k)$

$= -\|g_{k+1}\|^2 + \beta_k^{HS} g_{k+1}^T d_k$

By substituting β_k^{HS}

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

since $g_{k+1} d_k^T = 0$ (Exact line search)

$$g_{k+1}^T d_{k+1}^{HS} = -\|g_{k+1}\|^2 \leq -c_2 \|g_{k+1}\|^2 \quad c_2 \in (0,1) \tag{17}$$

Case 3 [13]: if $\delta_k = 0$, $\Upsilon_k = 0$

$$g_{k+1}^T d_{k+1} = g_{k+1}^T d_{k+1}^{HZ} \leq -c_3 \|g_{k+1}\|^2 \tag{18}$$

Where $c_3 = 7/8$

Case 4 [22]: if $\delta_k = 0$ and $0 < \Upsilon_k < 1$

$\beta_k^{hHZHS} = \Upsilon_k \beta_k^{HS} + (1-\Upsilon_k)\, \beta_k^{HZ}$ *where* $\Upsilon \in [0,1]$

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

$d_{k+1}^{hHZHS} = (1-\Upsilon_k) d_{k+1}^{HZ} + \Upsilon_k d_{k+1}^{HS}$

By multiplying by g_{k+1}^T:

$g_{k+1}^T d_{k+1}^{hHZHS} = (1-\Upsilon_k) g_{k+1}^T d_{k+1}^{HZ} + \Upsilon_k g_{k+1}^T d_{k+1}^{HS}$

Three cases are obtained:

If $\Upsilon_k = 0$ then:

$g_{k+1}^T d_{k+1}^{hHZHS} = g_{k+1}^T d_{k+1}^{HZ}$

W.W. Hager and H. Zhang proved [22] that d_{k+1}^{HZ} satisfies the sufficient descent condition i.e. there is $c_3 = \frac{7}{8} > 0$

$g_{k+1}^T d_{k+1}^{HS} = -\|g_{k+1}\|^2 + \frac{g_{k+1}^T y_k}{d_k^T y_k} (g_{k+1}^T d_k) g_{k+1}^T d_{k+1}^{HZ} \leq -c_3 \|g_{k+1}\|^2$

If $\Upsilon_k = 1$ then:

$g_{k+1}^T d_{k+1}^{hHZHS} = g_{k+1}^T d_{k+1}^{HS}$

Clearly from strong Wolf condition that:

$(-\sigma - 1) d_k^T g_k \geq d_k^T y_k = d_k^T (g_{k+1} - g_k)$

$\geq (\sigma - 1) d_k^T g_k$

With substitution:

$g_{k+1}^T d_{k+1}^{HS} \leq -c_2 \|g_{k+1}\|^2$

Where $\sigma < \frac{1}{2}$

Finally, for $0 < \Upsilon_k < 1$

$g_{k+1}^T d_{k+1}^{hHZHS} = (1-\Upsilon_k) g_{k+1}^T d_{k+1}^{HZ} + \Upsilon_k g_{k+1}^T d_{k+1}^{HS}$

$\leq -(c_3(1-\Upsilon_k) + c_2 \Upsilon_k) \|g_{k+1}\|^2$

There is $\mu, \lambda \in \mathbb{R}$ where $0 < \mu < \Upsilon_k < \lambda < 1$ that gives

$$g_{k+1}^{T} d_{k+1}^{hHZHS} \leq -c_4 \|g_{k+1}\|^2 \quad (19)$$

Where $c_4 = (c_3(1-\lambda) + c_2\mu) > 0$

Case 5 [17]: if $\Upsilon_k = 0 \; and \; 0 < \delta_k < 1$ then:

$$\beta_k^{hHZDY} = \delta_k \beta_k^{DY} + (1-\delta_k)\, \beta_k^{HZ}$$

The sufficient descent condition for the convex combination for DY and HZ has to be proven:
Here $d_0 = -g_0$, $d_{k+1} = -g_{k+1} + \beta_k d_k$, let the step size α_k be generated by strong Wolfe condition i.e.

$$f(x_k + \alpha_k d_k) - f(x_k) \leq \sigma \alpha_k \nabla f(x_k^T) d_k$$

$$\sigma \nabla f(x_k^T) d_k \leq \nabla f(x + \alpha_k d_k)^T d_k \leq -\sigma \nabla f(x_k^T) d_k$$

$g_k^T d_k \leq -c\|g_k\|^2$ must be demonstrated

By using mathematical induction, it's clear that:

$g_k^T d_k \leq -c\|g_k\|^2$ for $k = 0$

Assume that its true for k

For $k = k+1$

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

$$= -(1-\delta_k) g_{k+1} - \delta_k g_{k+1} + (1-\delta_k) \beta_k^{HZ} d_k + \delta_k \beta_k^{DY} d_k$$

Implies

$$d_{k+1}^{hHZDY} = (1-\delta_k) d_{k+1}^{HZ} + \delta_k d_{k+1}^{DY}$$

Multiplying both sides by g_{k+1}^T

$$g_{k+1}^T d_{k+1}^{hHZDY} = (1-\delta_k) g_{k+1}^T d_{k+1}^{HZ} + \delta_k g_{k+1}^T d_{k+1}^{DY}$$

Two cases are obtained:

- **If $\delta_k = 0$ then $g_{k+1}^T d_{k+1}^{hHZDY} = g_{k+1}^T d_{k+1}^{HZ}$ (which is proved)**
- **If $\delta_k = 1$ then $g_{k+1}^T d_{k+1}^{hHZDY} = g_{k+1}^T d_{k+1}^{DY}$**

Since $(-\sigma - 1) d_k^T g_k \geq d_k^T y_k = d_k^T (g_{k+1} - g_k) \geq (\sigma - 1) d_k^T g_k$

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

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

$$\leq -(1 - \tfrac{\sigma}{1-\sigma}) \|g_{k+1}\|^2$$

Hence $g_{k+1}^T d_{k+1}^{DY} \leq -c_1 \|g_{k+1}\|^2$ where $\sigma \leq \frac{1}{2}, 0 < \delta_k < 1$

$$g_{k+1}^T d_{k+1}^{hHZDY} = (1-\delta_k) g_{k+1}^T d_{k+1}^{HZ} + \delta_k g_{k+1}^T d_{k+1}^{DY}$$

$$\leq -(c_3(1-\delta_k) + c_1 \delta_k \|g_{k+1}\|^2)$$

Then there exists $\mu, \lambda \in R$ where $0 < \mu < \delta_k < \lambda < 1$ which gives:

$$g_{k+1}^T d_{k+1}^{new} \leq -c_5 \|g_{k+1}\|^2 \quad (20)$$

When $c_5 = (c_3(1-\lambda) + c_1\mu)$

Case 6 [18]: if $(1 - \delta_k - \Upsilon_k) = 0$ when $0 < \delta_k, \Upsilon_k < 1$ then $\Upsilon_k = 1 - \delta_k$

$$\beta_k^{hHSDY} = \delta_k \beta_k^{DY} + (1-\delta_k)\, \beta_k^{HS}$$

$$g_{k+1}^T d_{k+1} \le -c_6 \|g_{k+1}\|^2 \quad (21)$$

where $c_6 = c_u c_v,\ c_u = \left(\frac{s_k^T g_{k+1}}{g_k^T g_{k+1}}\right),\ c_v = \left(\frac{s_k^T g_k}{y_k^T s_k}\right)$

Case 7: If $0 < \delta_k < 1$ and $0 < \Upsilon_k < 1$ and $0 < \delta_k + \Upsilon_k < 1$ then a new hybrid conjugate gradient method is obtained as a convex combination of DY, HS and HZ.
It has to be proven that the direction satisfies the sufficient descent condition at each iteration i.e.

$g_{k+1}^T d_{k+1} \le -c\|g_{k+1}\|^2$ when $(0 < \delta_k < 1$ and $0 < \Upsilon_k < 1)$

$d_{k+1} = -g_{k+1} + \beta_k^{hSH} d_k$ when β is a convex combination parameter of DY, HS and HZ

$\beta_k^{hSH} = \lambda_1 \beta_k^{DY} + \lambda_2 \beta_k^{HS} + \lambda_3 \beta_k^{HZ}$ when $\lambda_1, \lambda_2, \lambda_3 > 0$ and $\lambda_1 + \lambda_2 + \lambda_3 = 1$

This ensures that β_k^{hSH} is a weighted average of the individual conjugate gradient parameters inherited properties from each.
Now from $g_{k+1}^T d_{k+1} \le -c\|g_{k+1}\|^2$ and $d_{k+1} = -g_{k+1} + \beta_k^{hSH} d_k$

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

Since β_k^{hSH} is a convex combination of the parameters $\beta_k^{DY}, \beta_k^{HS}$ and β_k^{HZ} and since each individual method have well-known descent properties and under standard conditions (such as using line search that satisfies Wolfe conditions), each technique satisfies sufficient descent condition.

$g_{k+1}^T d_{k+1} \le -c\|g_{k+1}\|^2$ and $\beta_k^{hSH} = \lambda_1 \beta_k^{DY} + \lambda_2 \beta_k^{HS} + \lambda_3 \beta_k^{HZ}$

$$g_{k+1}^T d_{k+1} \le \lambda_1(-c_1\|g_k\|^2) + \lambda_2(-c_2\|g_k\|^2) + \lambda_3(-c_3\|g_k\|^2)$$

Hence

$$g_{k+1}^T d_{k+1} \le -(\lambda_1 c_1 + \lambda_2 c_2 + \lambda_3 c_3)\|g_{k+1}\|^2$$

Since $\lambda_1 + \lambda_2 + \lambda_3 = 1$

Let $c_7 = \lambda_1 c_1 + \lambda_2 c_2 + \lambda_3 c_3$ then:

$$g_{k+1}^T d_{k+1} \le -c_7\|g_{k+1}\|^2 \quad (22)$$

When $c_7 > 0$ and is a constant derived from the convex combination of the individual descent conditions.

2.3.2. *Convergence Analysis*

The conjugate gradient methods of global convergence are frequently demonstrated using the Zoutendijk criterion [23]. Furthermore, the Zoutendijk requirement is met by the proposed approach under the strong Wolfe condition, as demonstrated by the following lemma:
Lemma 2 [24]: Consider the assumption (1) and (2) hold and from (2) where $\boldsymbol{d_k}$ is a descent direction and $\propto_{\boldsymbol{k}}$ is the step size determined by strong Wolf conditions, then, Zoutendijk condition

$$\sum_{k\ge1} \frac{(g_{k+1}^T d_{k+1})^2}{\|d_{k+1}\|^2} < \infty \quad \text{holds.} \quad (23)$$

The following theorem proves the novel method's global convergence:
Theorem 2 [25]: Suppose the assumption (1) and (2) hold and $\{\boldsymbol{x_k}\}$ be generated by the new algorithm, then:

$$\lim_{k\to\infty} inf\|g_{k+1}\| = 0 \quad (24)$$

Proof: this theorem is proved by using contradiction. Suppose $\boldsymbol{\lim_{k\to\infty} inf}\|\boldsymbol{g_{k+1}}\| = \boldsymbol{0}$ is not true, then there exist C > 0 such that $\|\boldsymbol{g_{k+1}}\| \ge \boldsymbol{C}$ for all k≥1 from theorem (1):

$$g_{k+1}^T d_{k+1} \le -K\|g_{k+1}\|^2 \text{ for all K>0}$$

From Lipschitz rule:
$\|y_k\| = \|g_{k+1} - g_k\| \le L\|x_{k+1} - x_k\| \le LD$

Where $D = \max\{\|x-y\|: x, y \in N\}$ is diameter of N

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

$\|d_{k+1}\| \le \|g_{k+1}\| + |\beta_k^{hSH}|\,\|d_k\|$

From (7) and according to assumptions 1 and 2, the strong wolf conditions, it is concluded that α_k which is obtained in the proposed method is not equal to zero, i.e., there exists a constant $\lambda > 0$ such that

And since:

$$\beta_k^{hSH} = \delta_k\beta_k^{DY} + \Upsilon_k\beta_k^{HS} + (1-\delta_k-\Upsilon_k)\,\beta_k^{HZ} \quad \text{where } 0 < \delta_k, \Upsilon_k < 1 \text{ and } 0 < 1-\delta_k-\Upsilon_k < 1 \text{ then}$$

$$|\beta_k^{hSH}| \le |\beta_k^{DY}| + |\beta_k^{HS}| + |\beta_k^{HZ}|$$

$$= \frac{\|g_{k+1}\|^2}{y_k^T d_k} + \frac{g_{k+1}^T y_k}{y_k^T d_k} + \frac{g_{k+1}^T y_k}{d_k^T y_k} - 2\frac{\|y_k\|^2}{d_k^T y_k}\frac{g_{k+1}^T d_k}{d_k^T y_k}$$

$$\le \frac{\|g_{k+1}\|^2}{\|y_k^T\|\|d_k\|} + \frac{\|g_{k+1}^T\|\|y_k\|}{\|y_k^T\|\|d_k\|} + \frac{\|g_{k+1}^T\|\|y_k\|}{\|d_k^T\|\|y_k\|} - 2\frac{\|y_k\|^2}{\|d_k^T\|\|y_k\|}\frac{\|g_{k+1}^T\|\|d_k\|}{\|d_k^T\|\|y_k\|}$$

And since:

$g_{k+1}^T d_{k+1} \le -K\|g_{k+1}\|^2$, $\|\nabla f(x)\| \le \Upsilon$ and since $y_k\| \le LD$

$$|\beta_k^{hSH}| \le \frac{\Upsilon^2}{KLD} + \frac{\Upsilon LD}{KLD} + \frac{\Upsilon LD}{KLD} - 2\frac{L^2D^2\Upsilon K}{(KLD)^2} = M$$

$\alpha_k \ge \lambda$ for all , $k \ge 0$ then,

$\frac{1}{\alpha_k} \le \frac{1}{\lambda}$ hence

$$\|d_{k+1}\| \le \|g_{k+1}\| + |\beta_k^{hSH}|\,\|d_k\| \le \|g_{k+1}\| + \frac{|\beta_k^{hSH}|\,\|x_{k+1}-x_k\|}{\alpha_k} \le \Upsilon + \frac{MD}{\lambda} = W$$

Hence

$$\|d_{k+1}\| \le W \text{ then } \sum_{k\ge1}\frac{1}{\|d_k\|^2} = \infty,\ k \ge 0$$

From Zoutendijk condition,

$\sum_{k\ge1}\frac{(g_{k+1}^T d_{k+1})^2}{\|d_{k+1}\|^2} < \infty$ and since $\|g_{k+1}\| \ge C$ and $g_{k+1}^T d_{k+1} \le -K\|g_k\|^2$

$K^2C^4 \sum_{k\ge1}\frac{1}{\|d_k\|^2} \le \sum_{k\ge1}\frac{k^2\|g_k\|^4}{\|d_k\|^2} \le \infty$

Which is in contradiction with $\sum_{k\ge1}\frac{1}{\|d_{k+1}\|^2} = \infty$

3. Results

3.1. Numerical Analysis

In this section, authors present numerical experiment results obtained by testing the new algorithm hSH with HS, DY and HZ conjugate gradient algorithms on a set of 96 unconstrained optimization test problems, in which the problems 1-53 are taken from the constrained and unconstrained testing environment library [26]. The others come from the unconstrained problem collections [27, 28]. The dimensions of the test problems vary from 500 to 500000. For the sake of fairness, all the comparison methods use the strong Wolfe line search to compute the step-length α_k, and the relevant parameters are set to ϱ=0.0001 and σ=0.9 and the hybridization parameter $\delta_k = \gamma_k = 0.5$. The strategy described in [29,30] is adopted to compute the initial step length for the proposed method. The termination criterion is (1) $\|g_k\|_\infty \le$10^(-6) or (2) NOI> 2000, where "NOI" represents the number of iterations. When (2) does happen, the relevant algorithm is claimed to be invalid for the corresponding test problem and denote it by "NAN". All codes are written in Matlab (as a tool for data analysis) 2024b and run on a Lenovo PC with a 3.60 GHz Central Processing Unit (CPU) processor and 8 GB RAM memory and Windows 10 operation system. Comparisons of these methods are given in the following context. Let e.g. fi^{H1} and fi^{H2} be the optimal values found by H_1andH_2, for problem i=1,...96, respectively. It is considered that in the particular problem i, the performance of H_1 was better than the performance of H_2 if

$$\left|f_i^{H1} - f_i^{H2}\right| < 10^{-3} \qquad (25)$$

The number of iterations (NOI), the number of function-gradient evaluations (NOF), or the CPU time of H_1 methods is less than those of H_2 methods, respectively, to obtain complete comparisons, the profile of Dolan and More [31] is used to evaluate and compare the performance of the set of methods. In the first set of numerical experiments, the performance of the proposed hSSH1 algorithm is compared with equation 9 when $\gamma_k = 0.5$ versus the HS, DY, and HZ conjugate gradient algorithms. The table 1. represents, the numerical results based on the NOI and CPU time [32, 33].

Table 1: Test results comparison based on the NOI and CPU time.

Function/Size	DY	HS	HZ	hSSH1
	Ite/Tcpu/Grad.	**Ite/Tcpu/Grad.**	**Ite/Tcpu/Grad.**	**Ite/Tcpu/Grad.**
cosine/5000	10/0.232/6.82e-07	14/0.069/2.54e-07	12/0.059/4.52e-07	12/0.076/2.88e-07
cosine/50000	11/0.297/3.63e-07	16/0.351/1.19e-07	12/0.269/4.12e-07	11/0.311/1.16e-07
cosine/500000	141/16.357/9.77e-07	104/13.935/8.96e-07	17/3.521/3.76e-07	13/2.996/9.07e-07
dixmaana/15000	12/0.467/8.32e-07	18/0.689/1.81e-07	13/0.436/3.85e-07	10/0.426/9.74e-09
dixmaana/150000	10/4.136/2.06e-07	17/6.107/1.20e-07	12/4.231/3.16e-07	10/3.635/1.11e-07
dixmaanb/15000	9/0.426/4.71e-07	16/0.492/7.18e-08	13/0.454/9.12e-07	9/0.375/6.12e-07
dixmaanb/150000	10/4.251/1.19e-07	17/5.425/1.23e-07	12/4.608/7.54e-07	10/4.031/2.01e-08
dixmaanc/15000	10/0.463/1.21e-07	15/0.505/1.10e-07	13/0.446/3.15e-07	9/0.355/2.72e-07
dixmaanc/150000	11/4.249/1.01e-07	18/6.021/5.31e-07	12/4.467/3.90e-07	10/4.154/6.65e-07
dixmaand/15000	10/0.425/3.86e-07	17/0.524/3.40e-07	13/0.427/5.21e-07	10/0.395/2.63e-07
dixmaand/150000	12/4.524/7.02e-07	15/4.980/4.88e-07	14/4.701/5.99e-07	10/4.131/6.17e-07
dixmaane/15000	680/6.862/9.92e-07	825/9.365/9.69e-07	918/12.062/9.44e-07	734/6.831/9.90e-07
dixmaane/150000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
dixmaanf/15000	443/4.543/9.84e-07	734/8.503/9.79e-07	824/11.152/8.40e-07	609/5.863/9.99e-07
dixmaanf/150000	1261/81.842/9.87e-07	NaN/NaN/NaN	NaN/NaN/NaN	1361/124.107/9.95e-07
dixmaang/15000	NaN/NaN/NaN	727/8.664/9.47e-07	734/9.682/9.86e-07	549/5.857/9.94e-07
dixmaang/150000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	1727/165.880/9.56e-07
dixmaanh/15000	NaN/NaN/NaN	678/7.959/8.47e-0	694/9.340/9.89e-07	546/5.560/9.58e-07
dixmaanh/150000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	1757/167.650/8.87e-07
dixmaani/15000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
dixmaanj/15000	NaN/NaN/NaN	1104/12.905/9.18e-07	1270/16.919/8.67e-07	766/7.099/8.33e-07
dixmaanj/150000	NaN/NaN/NaN	1866/206.411/8.37e-07	NaN/NaN/NaN	1880/168.385/9.26e-07
dixmaank/15000	1248/8.820/9.91e-07	944/10.700/9.75e-07	1112/14.049/9.59e-07	972/9.257/4.54e-07
dixmaank/150000	1950/22977.379/9.91e-07	1528/170.107/9.36e-07	1730/210.919/7.62e-07	1614/141.934/9.98e-07
dixmaanl/15000	1142/8.376/9.47e-07	1208/17.778/8.90e-07	966/12.292/9.61e-07	956/8.990/9.74e-07
dixmaanl/150000	1598/97.081/9.82e-07	1599/220.563/9.70e-07	1523/188.267/9.84e-07	1215/109.071/6.43e-07
dixon3dq/500	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
dqdrtic/5000	32/0.042/5.20e-07	48/0.039/8.38e-07	31/0.032/4.60e-07	30/0.030/9.09e-07
dqdrtic/50000	42/0.178/7.03e-07	38/0.145/5.21e-07	36/0.136/5.94e-07	29/0.119/2.82e-07
dqrtic/5000	38/0.358/9.62e-07	41/0.336/4.42e-07	49/0.391/8.28e-07	17/0.193/3.91e-07
dqrtic/50000	94/6.235/7.32e-07	28/2.132/2.42e-07	107/6.332/9.77e-07	21/2.023/1.46e-07
edensch/5000	26/0.220/7.77e-07	41/0.465/7.46e-07	27/0.234/5.93e-07	29/0.320/8.59e-07
edensch/50000	25/2.307/8.75e-07	39/3.945/9.26e-07	54/5.957/9.45e-07	36/3.293/9.11e-07
edensch/500000	30/24.466/7.92e-07	33/27.952/9.72e-07	25/22.080/5.00e-07	38/38.691/5.89e-07
eg2/500	NaN/NaN/NaN	982/0.941/9.67e-07	NaN/NaN/NaN	NaN/NaN/NaN
fletchcr/5000	32/0.057/7.35e-07	62/0.074/5.77e-07	77/0.090/6.53e-07	206/0.284/9.59e-07

Table 1: continue

fletchcr/50000	61/0.336/6.30e-07	150/0.992/9.38e-07	45/0.272/7.33e-07	53/0.347/5.61e-07
fletchcr/500000	140/9.347/3.56e-07	260/19.613/8.01e-07	145/10.283/9.85e-07	174/14.300/8.28e-07
freuroth/500	NaN/NaN/NaN	968/0.572/9.66e-07	NaN/NaN/NaN	552/0.320/9.78e-07
genrose/5000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
himmelbg/50000	2/0.047/0.00e+00	2/0.038/0.00e+00	2/0.030/0.00e+00	2/0.028/0.00e+00
himmelbg/500000	3/0.282/0.00e+00	3/0.320/6.82e-70	3/0.285/0.00e+00	3/0.297/0.00e+00
liarwhd/5000	479/0.286/8.87e-07	34/0.030/1.11e-07	33/0.026/1.95e-07	43/0.039/6.08e-07
liarwhd/50000	83/0.316/9.43e-07	40/0.163/1.38e-07	33/0.125/2.98e-07	84/0.376/1.36e-07
penalty 1/5000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	139/20.686/3.12e-07
penalty 1/50000	74/1351.136/2.96e-07	NaN/NaN/NaN	41/660.443/1.03e-07	36/592.580/2.50e-07
quartc/5000	38/0.536/9.62e-07	41/0.458/4.42e-07	49/0.462/8.28e-07	17/0.266/3.91e-07
quartc/50000	94/11.177/7.32e-07	28/4.264/2.42e-07	107/11.180/9.77e-07	21/3.788/1.46e-07
quartc/500000	133/104.972/7.84e-07	41/29.785/3.53e-07	189/109.494/7.38e-07	25/23.851/8.95e-08
tridia/500	561/0.177/1.00e-06	861/0.237/9.92e-07	955/0.276/8.42e-07	741/0.220/9.95e-07
tridia/5000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
woods/50000	NaN/NaN/NaN	139/0.369/6.71e-07	205/0.518/6.60e-07	166/0.554/3.05e-07
woods/500000	NaN/NaN/NaN	128/3.622/4.52e-07	159/4.175/6.64e-07	115/4.506/1.74e-07
bdexp/5000	2/0.036/0.00e+00	2/0.010/0.00e+00	2/0.009/0.00e+00	2/0.008/0.00e+00
bdexp/50000	2/0.058/0.00e+00	2/0.058/0.00e+00	2/0.058/0.00e+00	2/0.058/0.00e+00
bdexp/500000	2/0.632/1.55e-120	2/0.634/1.72e-120	2/0.628/1.54e-120	2/0.637/1.55e-120
exdenschnf/5000	17/0.049/1.71e-07	16/0.021/6.50e-07	17/0.014/5.99e-07	16/0.015/3.73e-07
exdenschnf/50000	29/0.109/8.73e-07	20/0.101/5.25e-07	18/0.089/1.52e-07	17/0.093/4.85e-07
exdenschnb/500000	77/1.509/8.97e-07	14/0.586/1.60e-07	12/0.521/3.79e-07	12/0.577/7.48e-07
genquartic/5000	13/0.039/2.28e-07	15/0.013/2.01e-07	13/0.010/7.01e-07	10/0.010/2.27e-08
genquartic/50000	11/0.060/2.09e-07	17/0.081/8.29e-07	12/0.064/4.30e-07	13/0.077/1.66e-07
genquartic/500000	12/0.675/6.87e-07	16/0.779/1.66e-07	14/0.704/3.05e-07	13/0.805/5.86e-07
biggsb1/500	1260/0.287/9.09e-07	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
sine/5000	NaN/NaN/NaN	138/0.337/7.77e-07	NaN/NaN/NaN	NaN/NaN/NaN
sine/50000	420/7.518/8.87e-07	NaN/NaN/NaN	25/0.711/5.51e-07	31/0.862/6.15e-07
sine/500000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
fletcbv3/500	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
nonscomp/5000	61/0.066/8.20e-07	65/0.045/6.39e-07	37/0.027/8.23e-07	35/0.030/9.28e-07
nonscomp/50000	NaN/NaN/NaN	177/0.582/7.75e-07	41/0.141/7.33e-07	53/0.185/7.62e-07
power1/500	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
raydan1/500	178/0.096/8.44e-07	187/0.066/9.16e-07	199/0.065/9.76e-07	183/0.070/9.05e-07
raydan1/5000	643/0.589/9.76e-07	643/0.626/9.86e-07	755/0.751/8.33e-07	626/0.580/9.63e-07
raydan2/5000	8/0.036/1.94e-07	36/0.132/5.23e-08	8/0.028/1.94e-07	8/0.031/1.94e-07
raydan2/50000	10/0.268/1.91e-07	30/0.902/5.89e-08	10/0.273/1.90e-07	10/0.276/1.90e-07
raydan2/500000	11/3.033/5.80e-07	56/17.582/2.70e-07	11/3.084/3.95e-08	11/3.130/1.83e-07
diagonal1/500	1275/1.107/9.60e-07	504/0.398/9.62e-07	392/0.286/9.33e-07	427/0.367/9.77e-07
diagonal1/5000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
diagonal2/5000	989/1.812/9.97e-07	469/0.907/7.72e-0	427/0.798/9.60e-07	461/0.925/9.51e-07
diagonal2/50000	NaN/NaN/NaN	NaN/NaN/NaN	1728/25.397/8.83e-07	NaN/NaN/NaN
diagonal3/500	1171/0.880/9.80e-0	833/0.598/9.93e-07	NaN/NaN/NaN	1008/0.761/8.51e-07
diagonal3/5000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN

Table 1: continue				
bv/500	NaN/NaN/NaN	1112/2.750/9.36e-07	802/2.233/8.83e-07	1095/2.456/6.41e-07
bv/5000	0/0.041/2.00e-07	0/0.041/2.00e-07	0/0.042/2.00e-07	0/0.042/2.00e-07
singx/1000	1198/11.578/9.63e-07	214/2.679/7.93e-07	140/1.667/4.09e-07	182/2.313/6.70e-07
singx/5000	NaN/NaN/NaN	91/16.804/1.93e-07	103/20.968/2.66e-07	204/42.777/1.47e-07
osb2/11	NaN/NaN/NaN	403/0.157/9.37e-07	456/0.171/6.17e-07	513/0.209/8.71e-07
pen1/500	718/1.783/1.80e-07	65/0.522/1.53e-07	97/0.723/2.80e-07	57/0.458/2.23e-07
pen1/5000	NaN/NaN/NaN	142/64.182/5.65e-08	50/21.173/4.42e-07	62/28.010/5.81e-07
pen2/500	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
rosex/500	90/0.481/3.32e-07	44/0.341/4.47e-07	39/0.250/1.70e-07	60/0.412/4.74e-07
rosex/5000	93/19.816/8.30e-07	39/12.305/7.40e-07	32/10.085/9.11e-07	56/17.780/4.17e-07
trid/500	46/0.200/7.04e-07	34/0.152/7.48e-07	32/0.134/7.82e-07	33/0.183/6.26e-07
trid/5000	474/44.755/9.57e-07	38/6.742/8.70e-07	37/6.174/6.92e-07	72/10.882/7.72e-07

Figures 1 and 2, illustrate the performance profiles of the new method hSSH1 versus HS, DY and HZ based on the NOI and CPU time.

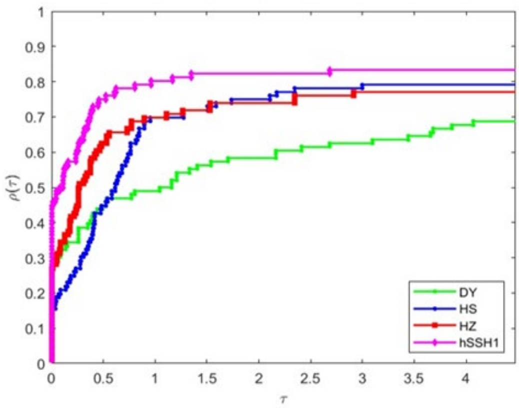

Figure 1: Performance profiles using the iteration number.

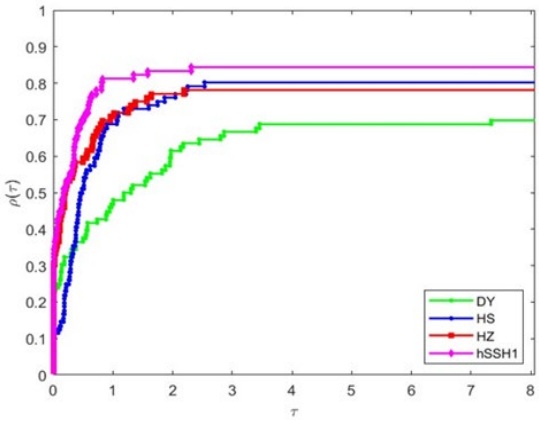

Figure 2: Performance profiles using the CPU time.

In the second set of numerical experiments, the performance of the new algorithm hSH is compared with eq. (10) δ_k=0.5 versus the HS, DY and HZ conjugate gradient algorithms. Table 2. illustrates the numerical results based on the NOI and CPU time, respectively.

Table 2. Second set of test results comparison based on the NOI and CPU time.

Function/Size	DY	HS	HZ	hSSH2
	Ite/Tcpu/Grad.	Ite/Tcpu/Grad.	Ite/Tcpu/Grad.	Ite/Tcpu/Grad.
cosine/6000	11/0.201/6.11e-07	11/0.092/6.11e-07	11/0.088/6.11e-07	11/0.106/6.11e-07
cosine/100000	11/0.929/2.77e-07	11/1.088/1.96e-07	11/0.900/2.92e-07	11/1.222/2.64e-07
cosine/800000	13/8.594/6.81e-07	13/9.139/6.81e-07	13/9.802/6.81e-07	13/14.215/6.81e-07
dixmaana/6000	9/0.567/6.51e-08	9/0.439/6.51e-08	9/0.419/6.51e-08	9/0.432/6.51e-08
dixmaana/90000	10/6.466/6.48e-07	10/8.091/6.48e-07	10/6.885/6.48e-07	10/6.455/6.48e-07
dixmaanb/24000	8/1.820/3.45e-07	8/1.565/3.45e-07	8/1.569/3.45e-07	8/1.730/3.45e-07
dixmaanb/48000	10/3.508/1.37e-07	10/3.805/1.37e-07	10/3.107/1.37e-07	10/4.661/1.37e-07
dixmaanc/2700	9/0.366/7.40e-08	9/0.263/7.40e-08	9/0.307/7.40e-08	9/0.229/7.40e-08
dixmaanc/27000	10/2.380/8.39e-08	10/2.250/8.39e-08	10/3.254/8.39e-08	10/2.632/8.39e-08
dixmaand/12000	10/1.100/8.17e-07	10/0.974/8.17e-07	10/0.948/8.17e-07	10/0.954/8.17e-07
dixmaand/90000	10/6.375/7.97e-07	10/6.315/7.97e-07	10/5.726/7.97e-07	10/6.237/7.97e-07

Table 2: continue				
dixmaane/2400	322/2.503/9.61e-07	356/2.863/9.40e-07	402/3.315/8.79e-07	315/2.508/9.95e-07
dixmaane/48000	1480/152.408/9.62e-07	1376/158.812/8.72e-07	1828/226.346/7.75e-07	1144/66.063/9.98e-07
dixmaanf/15000	508/19.100/9.28e-07	776/30.426/7.79e-07	813/32.866/8.50e-07	569/16.063/9.07e-07
dixmaanf/60000	1454/166.262/8.98e-07	1367/176.373/8.26e-07	1397/198.060/9.30e-07	1038/73.402/9.75e-07
dixmaang/12000	528/14.711/9.23e-07	522/14.964/8.35e-07	741/23.772/9.48e-07	474/12.927/9.25e-07
dixmaang/90000	1134/193.514/7.93e-07	1547/323.612/8.86e-07	1760/402.882/9.25e-07	1270/128.925/9.68e-07
dixmaanh/6000	892/14.904/3.24e-07	926/16.477/9.99e-07	1013/17.217/6.95e-07	891/14.338/4.93e-07
dixmaanh/30000	1050/61.610/9.37e-07	794/53.330/9.80e-07	903/65.752/9.24e-07	694/27.666/9.66e-07
dixmaani/360	1813/3.507/9.35e-07	1704/3.271/8.19e-07	NaN/NaN/NaN	1723/2.460/9.98e-07
dixmaanj/15000	935/31.392/9.68e-07	855/29.960/9.86e-07	886/35.517/9.30e-07	1062/22.861/9.77e-07
dixmaanl/24000	847/54.423/8.16e-07	919/60.314/9.87e-07	817/51.560/7.20e-07	830/56.566/9.75e-07
dixon3dq/150	1326/1.210/9.76e-07	1282/1.162/6.37e-07	1749/1.688/8.48e-07	1103/0.922/9.95e-07
dixon3dq/1500	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
dqrtic/5000	17/0.723/3.91e-07	17/0.609/3.91e-07	17/0.655/3.91e-07	17/0.491/3.91e-07
dqrtic/150000	23/16.349/3.72e-07	23/15.471/3.72e-07	23/15.302/3.72e-07	23/15.546/3.72e-07
edensch/7000	37/1.390/6.65e-07	28/0.854/9.85e-07	24/0.680/9.88e-07	27/0.809/4.83e-07
edensch/40000	27/4.935/3.73e-07	37/6.608/4.94e-07	31/5.659/5.02e-07	34/6.178/7.43e-07
fletchcr/1000	45/0.117/9.54e-07	83/0.140/4.34e-07	40/0.052/9.76e-07	72/0.109/7.81e-07
fletchcr/50000	60/1.115/4.58e-07	49/0.917/5.58e-07	94/1.871/9.57e-07	84/1.739/1.54e-07
fletchcr/200000	151/10.438/7.43e-07	113/8.054/8.08e-07	141/9.700/7.09e-07	123/8.850/6.49e-07
freuroth/460	639/1.208/8.14e-07	431/0.780/5.41e-07	611/1.091/9.41e-07	585/1.017/9.93e-07
genrose/10000	115/0.397/5.50e-07	102/0.310/9.23e-07	149/0.461/9.99e-07	105/0.324/9.16e-07
himmelbg/70000	2/0.162/2.12e-174	2/0.102/2.12e-174	2/0.101/2.12e-174	2/0.102/2.12e-174
himmelbg/240000	2/0.357/6.78e-07	2/0.338/6.78e-07	2/0.274/6.78e-07	2/0.324/6.78e-07
penalty 1/4000	36/12.260/7.35e-07	192/53.614/2.86e-07	35/10.842/6.47e-08	40/9.515/5.09e-07
penalty 1/10000	13/23.011/3.63e-08	13/21.821/3.72e-07	13/19.860/3.60e-08	13/22.164/3.60--08
quartc/4000	17/0.395/2.01e-07	17/0.374/2.01e-07	17/0.577/2.01e-07	17/0.380/2.01e-07
quartc/80000	22/8.017/3.10e-07	22/8.079/3.10e-07	22/8.162/3.10e-07	22/964.731/3.10e-07
quartc/500000	25/38.576/1.15e-07	25/36.398/1.15e-07	25/36.035/1.15e-07	25/38.458/1.15e-07
tridia/300	595/0.348/9.95e-07	640/0.416/9.68e-07	643/0.351/9.03e-07	527/0.218/9.54e-07
tridia/2000	1463/0.848/9.65e-07	1997/1.460/9.50e-07	NaN/NaN/NaN	1546/1.292/8.97e-07
bdexp/50000	2/0.129/0.00e+00	2/0.106/0.00e+00	2/0.083/0.00e+00	2/0.108/0.00e+00
bdexp/500000	2/1.403/1.55e-120	2/1.290/1.55e-120	2/1.102/1.55e-120	2/1.234/1.55e-120
exdenschnf/9000	20/0.715/6.22e-07	16/0.569/6.58e-07	16/0.555/4.68e-07	17/0.525/3.47e-07
exdenschnf/280000	17/1.236/1.13e-07	18/1.302/2.22e-07	19/1.564/1.69e-08	16/1.343/7.56e-07
exdenschnf/600000	20/2.892/9.20e-08	20/3.003/3.78e-07	17/2.649/7.28e-07	17/2.957/9.39e-07
exdenschnb/6000	14/0.092/3.72e-07	12/0.029/8.32e-07	14/0.039/3.01e-07	14/0.041/2.02e-07
exdenschnb/24000	14/0.092/7.43e-07	13/0.094/1.30e-07	14/0.107/6.03e-07	14/0.099/4.04e-07
exdenschnb/300000	14/0.778/8.56e-07	13/0.784/4.49e-07	13/0.742/8.04e-07	17/1.007/1.32e-07
genquartic/9000	12/0.110/5.58e-07	12/0.037/5.58e-07	12/0.039/5.58e-07	12/0.061/5.58e-07
genquartic/90000	12/0.349/4.62e-07	12/0.349/1.59e-07	12/0.352/7.67e-07	12/0.410/4.09e-07
genquartic/500000	15/1.924/3.94e-07	15/2.065/5.01e-07	12/1.541/5.83e-07	15/2.070/4.76e-07
biggsb1/300	1373/1.083/9.67e-07	1387/1.128/8.38e-07	1427/1.352/9.37e-07	1097/0.873/8.81e-07
sine/100000	16/2.554/8.05e-07	17/2.852/7.90e-07	18/2.806/7.33e-07	21/3.234/1.88e-07
sine/250000	127/34.115/5.57e-07	31/12.221/5.16e-07	27/8.651/8.96e-07	105/35.070/4.24e-07

Table 2: continue

sine/500000	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
fletcbv3/100	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN	NaN/NaN/NaN
nonscomp/5000	41/0.289/4.02e-07	40/0.165/5.76e-07	37/0.169/9.89e-07	43/0.166/7.57e-07
nonscomp/80000	41/0.621/6.88e-07	41/0.659/8.61e-07	43/1.067/9.35e-07	40/0.576/7.28e-07
power1/150	1639/2.816/9.54e-07	1995/3.515/9.90e-07	NaN/NaN/NaN	1530/2.579/8.74e-07
raydan1/500	184/0.405/8.89e-07	199/0.492/8.25e-07	224/0.535/8.08e-07	170/0.378/9.35e-07
raydan1/5000	618/1.928/7.88e-07	757/2.269/9.10e-07	778/2.569/7.16e-07	592/2.994/9.59e-07
raydan2/2000	6/0.079/6.10e-08	6/0.051/6.10e-08	6/0.046/6.10e-08	6/0.049/6.10e-08
raydan2/20000	8/0.419/3.89e-07	8/0.360/3.89e-07	8/0.249/3.89e-07	8/0.256/3.89e-07
raydan2/500000	11/8.501/1.83e-07	12/9.783/8.22e-08	11/8.791/1.83e-07	11/10.989/1.83e-07
bv/2000	96/14.286/9.53e-07	102/13.060/9.61e-07	86/14.192/9.11e-07	109/11.945/8.70e-07
bv/20000	0/4.533/1.25e-08	0/2.347/1.25e-08	0/2.531/1.25e-08	0/2.678/1.25e-08
ie/500	9/18.944/1.45e-07	9/16.093/1.45e-07	9/16.622/1.45e-07	9/14.120/1.45e-07
ie/1500	10/118.637/3.10e-07	10/125.435/3.10e-07	10/139.482/3.10e-07	10/91.283/3.10e-07
singx/1000	161/6.598/7.31e-07	193/7.205/7.64e-07	377/13.437/7.35e-07	98/4.060/5.01e-07
singx/2000	110/13.621/2.29e-07	241/27.190/7.41e-07	330/36.645/9.48e-07	134/18.040/5.51e-07
lin/100	1/0.059/4.36e-14	1/0.031/4.36e-14	1/0.026/4.36e-14	1/0.021/4.36e-14
lin/1000	11/46.039/1.74e-07	16/81.663/2.63e-07	11/65.130/1.74e-07	11/62.825/1.74e-07
osb2/11	437/0.822/8.90e-07	616/0.829/9.33e-07	807/1.009/6.24e-07	433/0.534/8.64e-07
rosex/500	47/1.067/4.92e-07	62/1.350/4.74e-08	57/1.260/3.64e-07	53/1.224/4.45e-07
rosex/1000	57/4.822/6.55e-07	58/5.085/7.76e-07	60/5.220/9.82e-07	48/4.086/2.52e-07
trid/500	32/0.434/5.93e-07	33/0.422/7.73e-07	32/0.430/8.21e-07	33/0.434/6.09e-07
trid/8000	34/52.245/7.30e-07	33/55.504/9.83e-07	37/59.698/9.30e-07	33/53.081/7.97e-07

Figures 3 and 4 show the performance profiles of the new method hSH versus HS, DY, and HZ based on the NOI and CPU time.

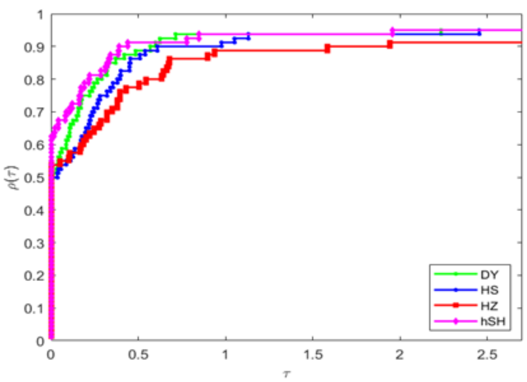

Figure 3: Performance profiles using the iteration number.

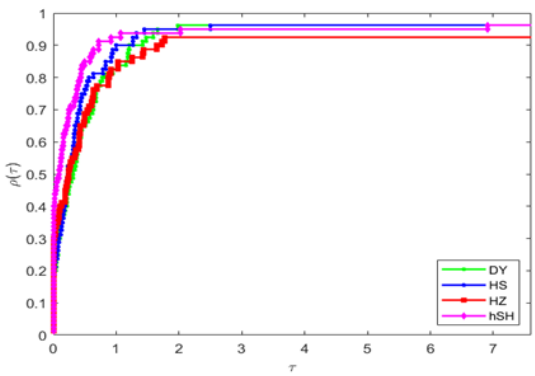

Figure 4: Performance profiles using the CPU time.

4. Discussion

A convex combination of the DY, HS, and HZ conjugate gradient methods were evaluated on a variety of benchmark optimization problems. By utilizing the advantages of each approach, the convex combination improved convergence speed and resilience compared to the individual approaches, especially for ill-conditioned and non-convex problems. Although HZ demonstrated stability in various landscapes, DY performed exceptionally well in ill-conditioned settings. The comparison of the new hybrid conjugate gradient method with existing methods in terms of NOI and CPU time (as presented in the tables above) is essential to demonstrate its efficiency and practical applicability. The iteration count directly reflects the convergence speed, and the new method's ability to converge in fewer iterations indicates faster convergence, which is particularly beneficial for large-scale optimization problems. Additionally, CPU time is a critical metric for real-world applications, as it accounts for the actual computational cost. The new method's lower CPU time demonstrates its scalability and efficiency on modern hardware. By comparing the proposed method with established techniques, its novelty and superior performance are highlighted, making it a viable alternative for solving complex optimization problems.

Moreover, regarding the study's limitations, the primary challenge was identifying and selecting the most effective features from each of the HS, DY, and HZ methods. This required a comprehensive analysis to determine which characteristics of these methods contributed most significantly to the overall performance, ensuring that the strengths of each approach were properly leveraged.

5. Conclusions

Conjugate gradient techniques are widely used to solve unconstrained optimization problems, particularly large-scale optimization problems. The hybrid approach, which combines traditional methodologies, is one of the most beneficial techniques. To develop a novel, effective technique, this research presents a new hybrid approach that computes parameters as a convex combination of three parameters: DY, HS, and HZ.

The practical results presented in this study demonstrate that the proposed strategy is faster and more effective than the alternative methods.

The authors recommend that future research explore the application of this new algorithm in training Feedforward Neural Networks (FFNNs) and consider its integration with fuzzy logic, time series analysis, and finite difference methods for further enhancement and broader applicability.

Author contributions: **Sara Sahib Mohammed Zaki**: Conceptualization, Data curation, Formal Analysis, Investigation, Methodology, Validation, Visualization, Writing – original draft, **Sozan Saber Haider**: Writing – review & editing, Software, **Hawraz Nadhim Jabbar**: Writing – review & editing, Software

Data availability: Data will be available upon reasonable request.

Conflicts of interest: The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Funding: The authors did not receive support from any organization for the submitted work.

References

[1] B. A. Hassan, I. A. R. Moghrabi, A. L. Ibrahim, and H. N. Jabbar, "Improved Conjugate Gradient Methods for Unconstrained Minimization Problems and Training Recurrent Neural Network," *Engineering Reports*, vol. 7, no. 2, pp. 1-13, Feb. 2025, doi: 10.1002/eng2.70019.

[2] D. H. Omar, A. L. Ibrahim, M. M. Hassan, B. G. Fathi, and D. A. Sulaiman, "Enhanced Conjugate Gradient Method for Unconstrained Optimization and Its Application in Neural Networks," *European Journal of Pure and Applied Mathematics*, vol. 17, no. 4, pp. 2692-2705, Oct. 2024, URL: https://www.ejpam.com/index.php/ejpam/article/view/5354/1727.

[3] R. Fletcher, and C. Reeves, "Function minimization by conjugate gradients," *The Computer Journal*, vol. 7, pp. 149-154, 1964, DOI:10.1093/COMJNL/7.2.149.

[4] S. A. Hamad, D. H. Omar, D. A. Sulaiman, and A. L. Ibrahim, "A New Conjugate Gradient Method Based on Logistic Mapping for Unconstrained Optimization and Its Application in Regression Analysis," *Science Journal of University of Zakho*, vol. 12, no. 4, pp. 484-489, Nov. 2024, doi: 10.25271/sjuoz.2024.12.4.1310.

[5] J. E. Dennis, and R. B. Schnabel, Numerical Methods for Unconstrained Optimization and Nonlinear Equations, Philadelphia, PA, USA: SIAM, 1983, ISBN:978-0-89871-364-0.

[6] K. K. Abbo, and N. H. Hameed, "New Hybrid Conjugate Method as a Convex Combination of Liu-Storey and Dixon Methods," *Journal of Multidisciplinary Modeling and Optimization*, vol. 1, no. 2, pp.91-99, 2018, URL: https://journals.indexcopernicus.com/search/article?articleId=2631251.

[7] Y. N. Huda, and I. A. Huda, "A Modification of Dai-Yuan's Conjugate Gradient Algorithms for Solving Unconstrained Optimization," *Bulletin of the South Ural State University. Ser. Mathematical Modelling, Programming & Computer Software*, vol. 15, no. 3, pp. 127-133, 2022, DOI:10.14529/mmp220309.

[8] M. R. Hestenes, and E. Stiefel, "Method of conjugate gradient for solving linear systems," *Journal of Research National Bureau Standards*, vol. 49, pp. 409-436, 1952, DOI:10.6028/JRES.049.044.

[9] S. S. Djordjevic, "New hybrid conjugate gradient method as a convex combination of FR and PRP Methods," *Filomat*, vol. 30, no. 11, pp. 3083-3100, 2016, DOI:10.2298/FIL1611083D.

[10] R. Fletcher, Practical Methods of Optimization: Unconstrained Optimization, vol. 1, 2nd ed. New York, NY, USA: Willey, 1997, ISBN:9780471915478.

[11] Y. Liu, and C. Storey, "Efficient generalized conjugate gradient algorithms," *Journal of Optim*ization *Theory and Application*, vol. 69, no. 1, pp. 129-137, 1991, DOI: 10.1007/BF00940464.

[12] Y. Dai, and Y. Yuan, "A nonlinear conjugate gradient with a strong global convergence property," *SIAM J. Optim.*, vol. 10, pp. 177-182, 2000, DOI:10.1137/S1052623497318992.

[13] W. W. Hager, and H. Zhang, "A survey of nonlinear conjugate gradient methods," *Pacific journal of Optimization*, vol.2, pp. 35-58, 2006, URL: https://people.clas.ufl.edu/hager/files/cg_survey.pdf.

[14] 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-Oper. Res*, vol. 58, pp. 613-627, 2024, DOI:10.1051/ro/2023196.

[15] N. Andrei, "Another nonlinear conjugate gradient algorithm for unconstrained optimization," *Optimization Methods and Software*, vol. 24, no.1, pp. 89-104, 2009, DOI: 10.1080/10556780802393326.

[16] 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, DOI: DOI:10.1051/ro/2022109.

[17] A. Hallal, M. Belloufi, and B. Sellami, "A new hybrid CG method as convex combination," *Mathematical Foundations of Computing*, vol. 7, no. 4, pp. 522-530, 2024, Doi:10.3934/mfc.2023028.

[18] Y. Yu, Y. Wang, R. Deng, and Y. Yin, "New DY-HS hybrid conjugate gradient algorithm for solving optimization problem of unsteady partial differential equations with convection term," *Mathematical and Computers in Simulation*, vol. 208, no.4, pp. 677-701, 2023, DOI: 10.1016/j.matcom.2023.01.033.

[19] S. S. Djordjevic, "New Hybrid Conjugate Gradient Method as a Convex Combination of LS and CD methods," *Filomat*, vol. 31, no. 6, pp. 1813-1825, 2017, DOI:10.2298/FIL1706813D.

[20] N. S. Mohamed, M. Mamat, M. Rivaie, and S. M. Shaharudin, "A new hybrid coefficient of conjugate gradient method," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 18, no. 3, pp. 1454-1463, 2020, DOI:10.11591/ijeecs.v18.i3.pp1454-1463.

[21] A. Hallal, M. Belloufi, and B. Sellami, "An Efficient New Hybrid CG-Method as Convex Combination of DY and CD and HS Algorithms," *RAIRO-Operations Research*, vol. 56, no.6, pp. 4047-4056, 2022, DOI: 10.1051/ro/2022200.

[22] W. W. Hager, and H. Zhang, "A New Conjugate Gradient Method with Guaranteed Descent and an Efficient Line Search," *SIAM Journal on Optimization.*, vol. 16, no. 1, pp. 170-192, 2005, DOI:10.1137/030601880.

[23] B. A. Hassan, Z. M. Abdullah, and S. A. Hussein, "A new family of conjugate gradient methods to solve unconstrained optimization problems," *Journal of Information and Optimization Sciences,* vol. 43, no. 4, pp. 811-820, Aug. 2022, doi: 10.1080/02522667.2022.2103299.

[24] N. Andrei, "Another hybrid conjugate gradient algorithm for unconstrained optimization," *Numerical Algorithms*, vol. 47, pp. 143-156, 2008, DOI:10.1007/s11075-007-9152-9.

[25] Y. Salih, M. A. Hamoda, Sukono, and M. Mamat, "The convergence properties of new hybrid conjugate gradient method," *IOP Conf. Series: Materials Science and Engineering*, vol. 567, 2019, DOI:10.1088/1757-899X/567/1/012031.

[26] M. J. D. Powell, "Restart procedures of the conjugate gradient method," *Mathematical Programming*, vol. 12, no.1, pp. 241-254, 1977, DOI: 10.1007/BF01593790.

[27] D. Touati-Ahmed, and C. Storey, "Efficient hybrid conjugate gradient techniques," *Journal of Optimization Theory and Applications*, vol. 64, pp. 379-397, 1990, DOI:10.1007/BF00939455.

[28] G. Zoutendijk, "Nonlinear programming computational methods," in *Integer and Nonlinear Programming*, J. Abadie, Ed. Amsterdam, Netherlands: North-Holland, 1970, pp. 37-86, ISBN:0-444-10000-8.

[29] X. Jiang, W. Liao, J. Yin, and J. Jian, "A new family of hybrid three-term conjugate gradient methods with applications in image restoration," *Numerical Algorithms*, vol. 91, pp. 161-191, 2022, DOI:10.1007/s11075-022-01258-2.

[30] S. Badreddine, Y. Laskri, and R. Benzine, "A new two-parameter family of nonlinear conjugate gradient methods," *Optimization*, vol. 64, no. 4, pp. 1-17, 2013, DOI:10.1080/02331934.2013.830118.

[31] E. D. Dolan, and J. J. More, "Benchmarking optimization software with performance profiles," *Mathematical Programming*, vol. 91, pp. 201-213, 2002, DOI:10.1007/s101070100263.

[32] J. J. More, B. S. Garbow, and K. E. Hillstrom, "Testing unconstrained optimization software," *ACM Transactions on Mathematical Software (TOMS)*, vol. 7, no. 1, pp. 17-41, 1981, DOI:10.1145/355934.355936.

[33] J. Mo, N. Gu, and Z. Wei, "Hybrid conjugate gradient methods for unconstrained optimization," *Optimization Methods and Software*, vol. 22, no. 2, pp. 297-307, 2007, DOI:10.1080/10556780500518653.