Malaysian Journal of Mathematical Sciences 18(1): 167–184 (2024)
https://doi.org/10.47836/mjms.18.1.10

Malaysian Journal of Mathematical Sciences

Journal homepage: https://mjms.upm.edu.my

A New Hybrid Three-Term LS-CD Conjugate Gradient In Solving Unconstrained Optimization Problems

Ishak M. A. I.*[1] and Marjugi S. M.[1]

[1] *Department of Mathematics and Statistics, Faculty of Science, Universiti Putra Malaysia, 43400 Serdang, Selangor, Malaysia*

E-mail: maqiiliqmal@gmail.com
**Corresponding author*

Received: 19 September 2023
Accepted: 3 January 2024

Abstract

The Conjugate Gradient (CG) method is renowned for its rapid convergence in optimization applications. Over the years, several modifications to CG methods have emerged to improve computational efficiency and tackle practical challenges. This paper presents a new three-term hybrid CG method for solving unconstrained optimization problems. This algorithm utilizes a search direction that combines Liu-Storey (LS) and Conjugate Descent (CD) CG coefficients and standardizes it using a spectral which acts as a scheme for the choices of the conjugate parameters. This resultant direction closely approximates the memoryless Broyden-Fletcher-Goldfarb-Shanno (BFGS) quasi-Newton direction, known for its bounded nature and compliance with the sufficient descent condition. The paper establishes the global convergence under standard Wolfe conditions and some appropriate assumptions. Additionally, the numerical experiments were conducted to emphasize the robustness and superior efficiency of this hybrid algorithm in comparison to existing approaches.

Keywords: unconstrained optimization; three-term conjugate gradient; memoryless quasi-Newton method; line search; global convergence.

1 Introduction

Consider the large-scale unconstrained optimization problem,

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

The function $f : \mathbb{R}^n \to \mathbb{R}$ is continuously differentiable and its gradient $g_k := \nabla f(x_k)$ exhibits Lipschitz continuity. The Newton, quasi-Newton method, and their respective alternatives have been proposed as viable ways for tackling unconstrained optimization problems [16]. However, these approaches are not considered optimal for addressing large-scale problems due to the requirement of computing and storing the Hessian matrix throughout each iteration. The singularity of the Hessian matrix occurs when the aforementioned approaches are unsuccessful. Consequently, the development of the Conjugate Gradient (CG) method was motivated by the need to address these challenges, given its advantages in terms of simplicity of implementation, Hessian-free approach, and minimal storage requirements [22].

The conjugate gradient approach utilises an iterative algorithm to solve equation (1) and generate a sequence $\{x_k\}$ in the following manner,

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

where α_k is the positive step length and the search direction d_k is given by,

$$d_k = \begin{cases} -g_k, & \text{if } k = 0, \\ -g_k + \beta_k d_{k-1}, & \text{if } k > 0. \end{cases} \tag{3}$$

The step length α_k is determined by evaluating the appropriate line search. The step length fulfills the standard Wolfe line search, whenever

$$f(x_k + \alpha_k d_k) - f(x_k) \le \eta \alpha_k g_k^T d_k, \tag{4}$$

$$g(x_k + \alpha_k d_k)^T d_k \ge \rho g_k^T d_k, \tag{5}$$

where $0 < \eta < \rho < 1$. Sufficient descent condition is one the important conditions for global convergence of CG methds that can facilitate the convergence structure. The search direction generated by the algorithm satisfies the sufficient descent condition, where there exists a $c > 0$ such that

$$g_k^T d_k \le -c\|g_k\|^2, \qquad c > 0. \tag{6}$$

Meanwhile, β_k represents the conjugate gradient parameter which plays a crucial role in shaping the overall convergence criteria and numerical efficiency of different conjugate gradient methods. The most well-known conjugate gradient methods include Hestenes-Stiefel (HS) [23], Polak-Ribiere-Polyak (PRP) [34, 35], Liu-Storey (LS) [30], Dai-Yuan (DY) [14], Fletcher-Reeves (FR) [20], and Conjugate Descent (CD) [21]. These methods are described as follows:

$$\beta_k^{\text{HS}} = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}, \qquad \beta_k^{\text{PRP}} = \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}, \qquad \beta_k^{\text{LS}} = \frac{g_k^T y_{k-1}}{-g_{k-1}^T d_{k-1}}. \tag{7}$$

$$\beta_k^{\text{DY}} = \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}}, \qquad \beta_k^{\text{FR}} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}, \qquad \beta_k^{\text{CD}} = \frac{\|g_k\|^2}{-g_{k-1}^T d_{k-1}} \tag{8}$$

respectively, where $y_{k-1} = g_k - g_{k-1}$ and $\|.\|$ denotes the Euclidean norm in $\mathbb{R}^n$.

According to Babaie-Kafaki and Ghanbari [11], the schemes employing a common numerator $g_k^T y_{k-1}$ tend to exhibit superior practical performance. However, Andrei [3, 4] stated that these schemes may not consistently converge due to interference and exhibit contrasting characteristics when compared to schemes employing a common numerator $\|g_k\|^2$. Babaie-Kafaki and Mahdavi-Amiri [12] stated that in pursuit of enhancing the efficacy of these strategies and avoid potential jamming, researchers expressed a keen interest in combining the schemes from both groups. From a theoretical standpoint, Hager and Zhang [22] assert that global convergence theorems for schemes using a common numerator $\|g_k\|^2$ only necessitate the Lipchitz assumption, unlike other choices of update parameters, which require boundedness assumptions. Powell [36] also highlighted that jamming is the primary factor contributing to the bad practical performance of the FR method. Babaie-Kafaki [10] stated that when a poor direction and a small step are generated between x_k and x_{k-1}, subsequently direction d_k and step length α_k are likely to be poor as well, unless a gradient restart is employed. Nevertheless, schemes using a common numerator exhibit an inherent approximate restart feature that addresses the issue of jamming, Babaie-Kafaki [9]. Based on Andrei [5, 6], the newly computed search direction d_k closely aligns with the steepest descent direction $-g_k$ when a small value of β_k is generated due to the small step s_{k-1}, in which the gradient difference y_{k-1} in the numerator approaches zero.

The CD method exhibits a close relation to FR scheme when employing an exact line search. One important difference between FR and CD methods is that the sufficient descent for CD holds for a strong Wolfe condition in which the constraints $c > 1/2$ for FR but unnecessary for CD. Hager and Zhang [22] emphasized that the CD method is globally convergent for a line search that satisfies the generalized Wolfe conditions with $\eta < 1$ and $\rho = 0$. Djordjevic [17] mentioned that there is limited research concerning the choice of β_k^{LS}, except the work conducted by Liu and Storey [30]. Nevertheless, it is anticipated that the analytical techniques developed for the PRP method can be effectively applied to the LS method, Hager and Zhang [22]. Similarly, Dai [13] showed that for an exact line search, the LS scheme is also identical to PRP. Following that, there are many works has been done regarding the hybridization of LS and CD method. Yang [41] introduced to the hybrid CG method known as LSCD under Wolfe line search, they proved the global convergence of the method. Again Djordjevic [17] proposed a new hybrid CG parameter that computed as convex combination of β_k^{LS} and β_k^{CD} in which satisfied both conjugacy condition and strong Wolfe line search conditions. Recently, Sahilu [37] also used the idea of convex combination proposed by Djordjevic [17] and hybridized by using Secant Equation which given as follows,

$$\beta_k^{\text{CLCS}} = (1-\theta_k)\beta_k^{\text{LS}} + \theta_k \beta_k^{\text{CD}}.$$

where θ_k is the hybridization scalar parameter satisfying $\theta_k \in [0, 1]$. It is obvious that $\beta_k^{\text{CLCS}} = \beta_k^{\text{LS}}$ as if $\theta_k = 0$, and $\beta_k^{\text{CLCS}} = \beta_k^{\text{CD}}$ as if $\theta_k = 1$. However, β_k^{CLCS} is a proper convex combination of β_k^{LS} and β_k^{CD} as if $0 < \theta_k < 1$. The hybrid computational able to yield such outperform or comparable results with known conjugate gradient algorithms.

Wang [39] introduced a spectral method that offers an optimal step length strategy within the gradient method. This approach serves as a novel way of determining the conjugate parameters and the newly computed search direction satisfies both the sufficient descent and spectral conditions. Global convergence under certain appropriate assumptions is subsequently established. The spectral parameter θ_k is defined as follows

$$\theta_k = \max\{\min\{\alpha_k^*, \bar{\rho}_k\}, \rho_k\}, \tag{9}$$

where $\alpha_k^* = -\dfrac{s_{k-1}^T g_{k-1}}{\varsigma \|y_{k-1}\|^2 \rho_k}$, $\bar{\rho}_k = \dfrac{\|s_{k-1}\|^2}{s_{k-1}^T y_{k-1}}$, $\rho_k = \dfrac{s_{k-1}^T y_{k-1}}{\|y_{k-1}\|^2}$ and ς is positive value.

Inspired by the idea of determining the suitable choice for the conjugate parameters introduced by Wang [39] and the problems discussed by Andrei [3, 4], Babaie-Kafaki [11] and other works into consideration in addressing issues of convergence and jamming. The key motivation of this paper is to prevent jamming by considering a combination of norms of $\|g_k\|^2$, $\|s_{k-1}\|^2$, and $\|y_{k-1}\|^2$. This modification computes the maximum of these norms which acting as a new alternative parameter that dynamically adjusts the CG update. Equation (16) introduces a crucial decision point in the CG update process. It hinges on the value of ω_k calculated in Equation (15). If ω_k equals $\|y_{k-1}\|^2$, the update direction becomes y_{k-1}, otherwise, it remains g_k. This decision is essential in avoiding jamming and maintaining convergence during the iterations. The rationale behind these equations is to combine the strengths of different CG schemes and adapt the update parameters dynamically to address jamming issues. By assessing the norms and switching between update directions based on the value of ω_k, these equations enhance the performance of CG methods as discussed by various authors in the provided literature.

In enhancing the traditional two-term direction previously discussed, researchers have developed hybrid and three-term CG methods aimed at improving their computational efficiency. One approach, proposed by Andrei [7] involving alteration of the inverse Hessian approximation within the Broyden-Fletcher-Goldfarb-Shanno (BFGS) formula while ensuring that the search direction adheres to the principles of descent and conjugacy. Another innovative method, introduced by Liu and Li [29], is a hybrid CG method that combines features of the LS and DY methods through a convex combination. This approach results in a search direction that satisfies both the Dai-Liao (DL) conjugacy condition and the Newton direction, with the added advantage of achieving global convergence through strong Wolfe line search. Xu and Kong [40] presented two hybrid algorithms that combine the PRP method with FR and the HS method with DY, respectively. Both of these hybrid methods yield descent directions and achieve global convergence through Wolfe line search. Dong [19] have devised a modified HS method that not only adheres to the descent criterion but also closely approximates the Newton method. The incorporation of the conjugacy condition aids in determining the hybridization parameter, ultimately resulting in the establishment of global convergence for general functions based on specific assumptions. Min Li [27] has suggested a three-term PRP CG method that closely resembles the memoryless BFGS quasi-Newton method. This method reverts to the classical PRP approach when exact line search conditions are met and the descent criterion is satisfied irrespective of line search considerations. Adequate line search strategies contribute to its global convergence and numerical results indicate its effectiveness in solving unconstrained optimization problems.

Additionally, Min Li [26] has introduced a nonlinear CG algorithm that generates a search direction akin to the memoryless BFGS quasi-Newton method. Notably, this search direction also meets the descent condition and under the framework of a strong Wolfe line search, global convergence has been established for both strongly convex and nonconvex functions. Abubakar [1] have presented a CG hybrid three-term algorithm wherein the search direction is determined using the limited memory BFGS method. This method manages to satisfy both the criteria of sufficient descent and trust region. It has been proven to achieve global convergence under specific conditions and has demonstrated efficiency when compared to some previously proposed methods. In addition, Kumam [25] and Deepho [15] also have introduced modifications to the CG hybrid three-term approaches, involving combinations of HS and LS as well as CD and DY provided a scaled preconditioner to the hybrid parameters. These modifications leverage existing conjugate gradient parameters, yielding positive results in solving a variety of test problems for both approaches. The similar concept was implemented in [31] and [2] with various combinations between conjugate parameters.

Inspires from the concepts elucidated in [1, 15, 25], we introduced a new CG hybrid three-term approach designed for addressing the problem denoted as (1). Referred to as the Three-Term LS-CD (TTLC) method, new approach amalgamates the three-term LS and CD directions. Furthermore, the direction closely mirrors that of the memoryless BFGS quasi-Newton method and adheres to trust region principles. We establish the global convergence of this method under both Wolfe line search conditions. The unique advantage and originality of our proposed method lie in its ability to encompass the favorable properties exhibited by both LS and CD directions. The significant contributions made by Andrei and Babaie-Kafaki in the realm of hybridization through convex combinations and Djordjevic motivated us to extend their approaches to access and combine the strength of the LS and CD CG update parameters. This paper is managed as follows. Section 2 presents the new proposed method. Convergence analyses are shown in Section 3. Numerical test results are reported in Section 4. Finally, conclusions are made in Section 5.

2 Algorithm and Theoretical Results

In this section, we commence by outlining our formulation, followed by the presentation of our proposed algorithm. In a prior study by Kumam [25], they introduced a CG hybrid three-term method denoted as HTTHSLS, which incorporates the following search direction,

$$d_k^{\text{HTTHSLS}} = -g_k + \left(\frac{g_k^T y_{k-1}}{v_k} - \frac{\|y_{k-1}\|^2 g_k^T d_{k-1}}{v_k^2}\right) d_{k-1} + t_k \frac{g_k^T d_{k-1}}{w_k} y_{k-1}, \qquad k \geq 1, \tag{10}$$

where,

$$v_k = \max\left(\mu\|d_{k-1}\|\|y_{k-1}\|, -d_{k-1}^T g_{k-1}, d_{k-1}^T y_{k-1}\right), \qquad \mu > 1, \qquad 0 \leq t_k \leq \bar{t}_k < 1.$$

Likewise, Deepho [15] introduced a CG hybrid three-term algorithm denoted as TTCDDY, featuring a search direction with the following structure,

$$d_k^{\text{TTCDDY}} = -g_k + \left(\frac{g_k^T g_k}{w_k} - \frac{\|g_k\|^2 g_k^T d_{k-1}}{w_k^2}\right) d_{k-1} - t_k \frac{g_k^T d_{k-1}}{w_k} g_k, \qquad k \geq 1, \tag{11}$$

where,

$$w_k = \max\left(\mu\|d_{k-1}\|\|g_k\|, -d_{k-1}^T g_{k-1}, d_{k-1}^T y_{k-1}\right), \qquad \mu > 1, \qquad 0 \leq t_k \leq \bar{t}_k < 1.$$

Both the HTTHSLS and TTCDDY methods are in compliance with the sufficient descent conditions and have been proved to globally converge under specific assumptions. Numerical results indicate that these hybrid approaches surpass their predecessors in terms of performance. Inspired by the HTTHSLS and TTCDDY methods, we introduce an innovative three-term hybrid CG algorithm that incorporates the LBFGS Quasi-Newton algorithm, integrating spectral standardization techniques introduced by Wang [39]. Following this, Subsequently, we will revisit the memoryless BFGS method proposed by Shanno [38] and Nocedal [33], wherein the search direction can be expressed as follows,

$$d_k^{\text{BFGS}} = -\left(I - \frac{s_{k-1}^T y_{k-1}}{s_{k-1}^T y_{k-1}} - \frac{y_{k-1}^T s_{k-1}}{s_{k-1}^T y_{k-1}} + \frac{s_{k-1} y_{k-1}^T y_{k-1} s_{k-1}}{s_{k-1}^T y_{k-1}} + \frac{s_{k-1} s_{k-1}^T}{s_{k-1}^T y_{k-1}}\right) g_k,$$

The equation $s_{k-1} = x_k - x_{k-1} = \alpha_{k-1} d_{k-1}$ holds, where I represents the identity matrix. By simplifying the d_k^{BFGS}, it can be expressed as,

$$d_k^{\text{BFGS}} = -g_k + \left(\frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}} - \frac{\|y_{k-1}\|^2 g_k^T d_{k-1}}{(d_{k-1}^T y_{k-1})^2}\right) d_{k-1} + \frac{g_k^T d_{k-1}}{d_{k-1}^T y_{k-1}} (y_{k-1} - s_{k-1}), \qquad k \geq 1. \tag{12}$$

By revisiting the proposed three-term LS and CD CG method by Kumam [25] and Deepho [15] respectively, which can be defined as,

$$d_k^{\text{TLS}} = -g_k + \left(\frac{g_k^T y_{k-1}}{-g_{k-1}^T d_{k-1}} - \frac{\|y_{k-1}\|^2 g_k^T d_{k-1}}{(-g_{k-1}^T d_{k-1})^2} \right) d_{k-1} + t_k \frac{g_k^T d_{k-1}}{-g_{k-1}^T d_{k-1}} y_{k-1}, \tag{13}$$

$$d_k^{\text{TCD}} = -g_k + \left(\frac{g_k^T g_k}{-g_{k-1}^T d_{k-1}} - \frac{\|g_k\|^2 g_k^T d_{k-1}}{(-g_{k-1}^T d_{k-1})^2} \right) d_{k-1} - t_k \frac{g_k^T d_{k-1}}{-g_{k-1}^T d_{k-1}} g_k. \tag{14}$$

We were inspired by the idea of spectral approach introduced by Wang [39] in establishing a standardization for both parameters. This modification involves replacing the terms associated with $\|g_k\|^2$, $\|s_{k-1}\|^2$, and $\|y_{k-1}\|^2$ to enable the selection of appropriate values for the conjugate parameter and search direction,

$$\omega_k = \max\left\{ \min\{\|g_k\|^2, \|s_{k-1}\|^2\}, \|y_{k-1}\|^2 \right\}, \tag{15}$$

where,

$$u_k = \begin{cases} y_{k-1} & \text{if } \omega_k = \|y_{k-1}\|^2, \\ g_k & \text{otherwise.} \end{cases} \tag{16}$$

Note that, the search direction $d_k^{\text{TTLC}} = d_k^{\text{TLS}}$ as if $u_k = y_{k-1}$, otherwise $d_k^{\text{TTLC}} = d_k^{\text{TCD}}$. Since the standardization of both search directions in equations (13) and (14) using equations (15) and (16) will be similar to the search direction of TTLC, the standardized search direction can be defined as follows,

$$d_k^{\text{TTLC}} = -g_k + \left(\frac{g_k^T u_k}{-g_{k-1}^T d_{k-1}} - \frac{\|u_k\|^2 g_k^T d_{k-1}}{(-g_{k-1}^T d_{k-1})^2} \right) d_{k-1} + t_k \frac{g_k^T d_{k-1}}{-g_{k-1}^T d_{k-1}} u_k. \tag{17}$$

To solve the problem of finding the univariate minimum, it becomes necessary to determine the parameter t_k,

$$\min_{t \in \mathbb{R}} \|(y_{k-1} - s_{k-1}) - t u_k\|^2 . \tag{18}$$

Let $A_k = (y_{k-1} - s_{k-1}) - t u_k$, then

$$\begin{aligned} A_k A_k^T &= \Big[(y_{k-1} - s_{k-1}) - t u_k\Big]\Big[(y_{k-1} - s_{k-1}) - t u_k\Big]^T \\ &= t^2 u_k u_k^T - t\Big[u_k^T (y_{k-1} - s_{k-1}) + (y_{k-1} - s_{k-1})^T u_k\Big] + (y_{k-1} - s_{k-1})(y_{k-1} - s_{k-1})^T. \end{aligned}$$

Let $B_k = y_{k-1} - s_{k-1}$, then

$$\begin{aligned} A_k A_k^T &= t^2 u_k u_k^T - t(u_k^T B_k + B_k^T u_k) + B_k B_k^T \\ \text{tr}(A_k A_k^T) &= t^2 \|u_k\|^2 - t\Big(\text{tr}(u_k^T B_k) + \text{tr}(B_k^T u_k)\Big) + \|B_k\|^2 \\ &= t^2 \|u_k\|^2 - 2t u_k^T B_k + \|B_k\|^2. \end{aligned}$$

By taking the derivative of the previous expression with respect to t_k and equating it to zero, we derive the following result,

$$2t\|u_k\|^2 - 2u_k^T B_k = 0.$$

This yields,

$$t = \frac{u_k^T(y_{k-1} - s_{k-1})}{\|u_k\|^2}. \tag{19}$$

Therefore, we choose t_k to be

$$t_k = \min\Big\{\bar{t}, \max\{0, t\}\Big\}, \tag{20}$$

where $0 \leq t_k \leq \bar{t} < 1$.

In accordance with the search direction stated in equations (17), we introduce a novel search direction for the CG hybrid three-term method which as follows,

$$d_0 = -g_0, \qquad d_k^{\text{TTLC}} = -g_k + \beta_k^{\text{TTLC}} d_{k-1} + \gamma_k^{\text{TTLC}} u_k, \qquad k \geq 1, \tag{21}$$

where

$$\beta_k^{\text{TTLC}} = \frac{g_k^T u_k}{-g_{k-1}^T d_{k-1}} - \frac{\|u_k\|^2 g_k^T d_{k-1}}{(-g_{k-1}^T d_{k-1})^2}, \qquad \gamma_k^{\text{TTLC}} = t_k \frac{g_k^T d_{k-1}}{-g_{k-1}^T d_{k-1}}. \tag{22}$$

Next, we describe algorithm of the proposed method.

Algorithm 2.1 Hybrid Three-Term LS-CD (TTLC)

Step 0: Choose an initial point $x_0 \in \mathbb{R}^n$, $\epsilon > 0$ $0 < \eta < \rho < 1$, $\bar{t} \in (0, 1)$. Set $k = 0$ and $d_0 = -g_0$.
Step 1: If $\|g_k\| \leq \epsilon = 10^{-6}$, stop; else, go to Step 2.
Step 2: Compute the standardization parameter u_k using (15) and (16).
Step 3: Calculate the conjugate gradient parameter β_k and γ_k using (22).
Step 4: Calculate the search direction d_k using (21)
Step 5: Compute the step length α_k using (4) and (5).
Step 6: Determine the next point $x_{k+1} = x_k + \alpha_k d_k$ and compute $g(x_{k+1})$, s_{k-1} and y_{k-1}.
Step 8: Set $k = k + 1$ and go to Step 1.

3 Convergence Analysis

In this section, we will establish the global convergence analysis of the TTLC method based on the subsequent assumptions

Assumption 1. *The set $\mathcal{H} = \{\mathbf{x} \in \mathbb{R}^n : f(\mathbf{x}) \leq f(\mathbf{x}_0)\}$, is bounded, with a starting point, x_0.*

Assumption 2. *Suppose there exists a neighborhood $\mathcal{H}$ of J where the gradient of f is Lipschitz continuous and continuously differentiable. In this neighborhood, we can find $L > 0$ such that for all x,*

$$\|g(x) - g(j)\|^2 \leq L\|x - j\|, \qquad j \in J.$$

Assuming Assumptions 1 and 2 hold, we can conclude that there exist positive constants A_1 and A_2 for all $x \in J$, such that,

$$\|x\| \leq A_1, \qquad \|g(x)\| \leq A_2.$$

Additionally, the sequence of function values $\{f(x_k)\}$ decreases as long as the sequence $\{x_k\}$ belonging to J is decreasing. Therefore, assuming the objective function has a lower bound and that Assumptions 1 and 2 are satisfied.

Following that, we outline the sufficient descent condition for the TTLC method.

Lemma 3.1. *The search direction d_k in (21) requires to satisfy (6) with $c = \left(1 - \frac{1}{4}\left(1+\bar{t}\right)^2\right)$.*

Proof. By multiplying both sides of (21) with g_k^T, we obtain

$$\begin{aligned} g_k^T d_k &= -\|g_k\|^2 + \frac{g_k^T u_k}{-g_{k-1}^T d_{k-1}} g_k^T d_{k-1} \quad - \frac{\|u_k\|^2}{(-g_{k-1}^T d_{k-1})^2}(g_k^T d_{k-1})^2 + t_k \frac{g_k^T u_k}{-g_{k-1}^T d_{k-1}} g_k^T d_{k-1} \\ &= -\|g_k\|^2 + (1+t_k)\frac{g_k^T u_k}{-g_{k-1}^T d_{k-1}} g_k^T d_{k-1} - \frac{\|u_k\|^2}{(-g_{k-1}^T d_{k-1})^2}(g_k^T d_{k-1})^2. \end{aligned} \tag{23}$$

We derive a_k and b_k by applying the inequality $a_k^T b_k \leq \frac{1}{2}\left(\|a_k\|^2 + \|b_k\|^2\right)$,

$$(1+t_k)\frac{g_k^T u_k}{-g_{k-1}^T d_{k-1}} g_k^T d_{k-1} \leq \frac{1}{4}(1+t_k)^2\|g_k\|^2 + \frac{\|u_k\|^2}{(-g_{k-1}^T d_{k-1})^2}(g_k^T d_{k-1})^2. \tag{24}$$

Substitute (24) into (23), we obtain

$$\begin{aligned} g_k^T d_k &\leq -\|g_k\|^2 + \frac{1}{4}(1+t_k)^2\|g_k\|^2 + \frac{\|u_k\|^2}{(-g_{k-1}^T d_{k-1})^2}(g_k^T d_{k-1})^2 - \frac{\|u_k\|^2}{(-g_{k-1}^T d_{k-1})^2}(g_k^T d_{k-1})^2 \\ &= -\|g_k\|^2 + \frac{1}{4}(1+t_k)^2\|g_k\|^2 \\ &\leq -\left(1 - \frac{1}{4}(1+\bar{t})^2\right)\|g_k\|^2. \end{aligned}$$

The proof is completed. □

Remark 3.1. *Lemma 3.1 shows that the TTLC method always satisfies the sufficient descent condition without requiring a line search.*

Dai and Yuan [14] showed that all conjugate gradient method under Wolfe line search holds.

Theorem 3.1. [14] *Given that Assumptions 1 and 2 are satisfied, and provided that conditions (4) and (5) are met, if*

$$\sum_{k=0}^{\infty} \frac{1}{\|d_k\|^2} = +\infty.$$

Then,

$$\lim_{k\to\infty} \inf \|g_k\| = 0. \tag{25}$$

Proof. By contradiction, assume that equation (25) is not met. In this case, there exists a positive scalar ξ such that,

$$\|g_k\| \geq \xi, \qquad \forall k > 0. \tag{26}$$

Lemma 3.2. *If $\{d_k\}$ is defined by (21), there exists $\lambda_1 > 0$ such that $\|d_k\| \le \|g_k\|\lambda_1$.*

Recalling the d_k^{TTLC} from (22) for $u_k = y_{k-1}$ when $\omega_k = \|y_{k-1}\|^2$,

$$\begin{aligned}
\left\|d_k^{\text{TTLC}}\right\| &\le \left\|-g_k + \beta_k^{\text{TTLC}} d_{k-1} + \gamma_k^{\text{TTLC}} y_{k-1}\right\| \\
&\le \|-g_k\| + |\beta_k^{\text{TTLC}}|\|d_{k-1}\| + |\gamma_k^{\text{TTLC}}|\|y_{k-1}\| \\
&= \|g_k\| + \left|\frac{g_k^T y_{k-1}}{-g_{k-1}^T d_{k-1}} - \frac{\|y_{k-1}\|^2 g_k^T d_{k-1}}{(-g_{k-1}^T d_{k-1})^2}\right| \|d_{k-1}\| + t_k \left|\frac{g_k^T d_{k-1}}{-g_{k-1}^T d_{k-1}}\right| \|y_{k-1}\| \\
&\le \|g_k\| + \left(\frac{\|g_k\|\|y_{k-1}\|}{\|g_{k-1}\|\|d_{k-1}\|} + \frac{\|y_{k-1}\|^2\|g_k\|\|d_{k-1}\|}{(\|g_{k-1}\|\|d_{k-1}\|)^2}\right)\|d_{k-1}\| + t_k\left(\frac{\|g_k\|\|d_{k-1}\|}{\|g_{k-1}\|\|d_{k-1}\|}\right)\|y_{k-1}\| \\
&\le \|g_k\| + \left(\frac{\alpha_{k-1}\|g_k\|\|d_{k-1}\|}{\mu\alpha_{k-1}\|d_{k-1}\|^2} + \frac{\alpha_{k-1}^2\|g_k\|\|d_{k-1}\|^3}{\mu^2\alpha_{k-1}^2\|d_{k-1}\|^4}\right)\|d_{k-1}\| \\
&\quad + t_k\left(\frac{\|g_k\|\|d_{k-1}\|}{\mu\alpha_{k-1}\|d_{k-1}\|^2}\right)\alpha_{k-1}\|d_{k-1}\| \\
&= \|g_k\| + \left(\|g_k\|\frac{1}{\mu} + \|g_k\|\frac{1}{\mu^2}\right) + \|g_k\| t_k\left(\frac{1}{\mu}\right) \\
&\le \|g_k\|\left(1 + \frac{1}{\mu} + \frac{1}{\mu^2} + \frac{\bar{t}}{\mu}\right).
\end{aligned}$$

In which $\lambda_1 = \|g_k\|\left(1 + \frac{1}{\mu} + \frac{1}{\mu^2} + \frac{\bar{t}}{\mu}\right)$, where $\|d_k\| \le \|g_k\|\lambda_1$.

The same proof technique is applied in another scenario where $u_k = g_k$ holds true, provided that $\omega_k \neq \|y_{k-1}\|^2$. Consequently, the sequence $\|d_k\|$ produced by the TTLC method possesses an upper bound. □

Next, we introduce the renowned Zoutendijk condition [42], a crucial element for the global convergence analysis of the TTLC method.

Lemma 3.3. [42] *Suppose that Assumptions 1 and 2 are satisfied, and the sequence $\{x_k\}$ is generated by (2), d_k satisfies the sufficient descent condition and α_k is computed by the standard Wolfe line search, then*

$$\sum_{k=0}^{\infty} \frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < +\infty. \tag{27}$$

Based on Lemma 3.1 and condition (4), for $\alpha_k > 0$, $\eta > 0$, $0 \le \bar{t} \le 1$, we obtain

$$\begin{aligned}
f(x_k + \alpha_k d_k) &\le f(x_k) + \eta\alpha_k g_k^T d_k \\
&\le f(x_k) - \eta\alpha_k\left(1 - \frac{1}{4}(1+\bar{t})^2\right)\|g_k\|^2 \\
&\le f(x_k).
\end{aligned}$$

By elaborating on the above outcome and contemplating Assumption 1, we have

$$f(x_{k+1}) \le f(x_k) + \eta\alpha_k g_k^T d_k \le f(x_k) \le f(x_{k-1}) \le \ldots \le f(x_0) < +\infty.$$

Incorporating condition (5) by adding $-g_k^T d_k$ gives,

$$g(x_k + \alpha_k d_k)^T d_k - g_k^T d_k \ge \rho g_k^T d_k - g_k^T d_k = -(1-\rho)\, g_k^T d_k.$$

Using Lemma 3.1, along with condition (5) and Assumption 2, it deduces as follows,

$$-(1-\rho)g_k^T d_k \le (g_{k+1}-g_k)^T d_k \le \|g_{k+1}-g_k\|\|d_k\| \le \alpha_k L\|d_k\|^2. \tag{28}$$

By multiplying above inequality with $-\eta g_k^T d_k$ and combine with (4), we obtain

$$\frac{\eta(1-\rho)}{L}\frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} \le -\eta\alpha_k g_k^T d_k \le f(x_k)-f(x_{k+1}),$$

and

$$\frac{\eta(1-\rho)}{L}\sum_{k=0}^{\infty}\frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} \le (f(x_0)-f(x_1))+(f(x_1)-f(x_2))+\ldots \le f(x_0) < +\infty.$$

As previously mentioned, the sequence $f(x_k)$ is limited within certain bounds. This implies that,

$$\sum_{k=0}^{\infty}\frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} < +\infty.$$

Inequality (26) in conjunction with (6) leads to the conclusion that,

$$g_k^T d_k \le -\left(1-\frac{1}{4}(1+\bar{t})^2\right)\|g_k\|^2 \le -\left(1-\frac{1}{4}(1+\bar{t})^2\right)\|\xi\|^2. \tag{29}$$

By squaring both sides and dividing equation (29) by $\|d_k\|^2$, where $\|d_k\| \ne 0$, we obtain

$$\sum_{k=0}^{\infty}\frac{\left(g_k^T d_k\right)^2}{\|d_k\|^2} \ge \left(1-\frac{1}{4}(1+\bar{t})^2\right)^2\sum_{k=0}^{\infty}\frac{\|\xi\|^4}{\|d_k\|^2} = +\infty. \tag{30}$$

As it conflicts with the Zoutendijk condition (27), the theorem is validated.

4 Numerical Experiments

In this section, we conduct an analysis of the performance of our novel TTLC CG algorithm on 150 test functions sourced from Andrei [8], Moré [32], and Jamil [24]. The newly proposed method, denoted as TTLC, will be compared against several other methods, including NHS+ [26], HTTHSLS [25], TTCDDY [15], HTT [1], TTPRLS [2], HTHP [31] and TTRMIL [28]. All the comparative methods were implemented and executed using Matlab R2021B with Intel® Core™ i5-9300H processor, 16 GB RAM, and 64-bit Windows 11 on a personal laptop. The comparisons are made based on reductions in terms of the number of iterations and central processing unit times that denoted as NOI and CPU time, respectively. These for each test functions cover a wide range of dimensions, spanning from 2 to 1,000,000 as detailed in Table 1.

Table 1: List of test functions and their dimensions.

No.	Functions	Dimensions	Initial Points
1	Extended White & Holst	50,000	(1.1, . . . , 1.1)
2	Extended White & Holst	100,000	(1.1, . . . , 1.1)
3	Extended White & Holst	1,000,000	(1.1, . . . , 1.1)

Continued on next page

Continued from previous page

No.	Functions	Dimensions	Initial Points
4	Extended Rosenbrock	50,000	$(0.1, \dots, 0.1)$
5	Extended Rosenbrock	100,000	$(0.1, \dots, 0.1)$
6	Extended Rosenbrock	1,000,000	$(0.1, \dots, 0.1)$
7	Extended Freudenstein and Roth	1,000	$(-2, \dots, -2)$
8	Extended Freudenstein and Roth	50,000	$(-2, \dots, -2)$
9	Extended Freudenstein and Roth	100,000	$(-2, \dots, -2)$
10	Extended Beale	1,000	$(1, \dots, 1)$
11	Extended Beale	50,000	$(1, \dots, 1)$
12	Extended Beale	100,000	$(1, \dots, 1)$
13	Raydan 1	10	$(1.1, \dots, 1.1)$
14	Raydan 1	50	$(1.1, \dots, 1.1)$
15	Raydan 1	100	$(1.1, \dots, 1.1)$
16	Extended Tridiagonal 1	10	$(-2.1, \dots, -2.1)$
17	Extended Tridiagonal 1	50	$(-2.1, \dots, -2.1)$
18	Extended Tridiagonal 1	10	$(-2.1, \dots, -2.1)$
19	Diagonal 4	1,000	$(0.1, \dots, 0.1)$
20	Diagonal 4	5,000	$(0.1, \dots, 0.1)$
21	Diagonal 4	50,000	$(0.1, \dots, 0.1)$
22	Extended Himmelblau	1,000	$(5, \dots, 5)$
23	Extended Himmelblau	50,000	$(5, \dots, 5)$
24	Extended Himmelblau	100,000	$(5, \dots, 5)$
25	FLETCHCR	100	$(-5, \dots, -5)$
26	FLETCHCR	5,000	$(-5, \dots, -5)$
27	FLETCHCR	50,000	$(-5, \dots, -5)$
28	Extended Powell	100	$(8, \dots, 8)$
29	Extended Powell	1,000	$(8, \dots, 8)$
30	NONSCOMP	2	$(10, 10)$
31	NONSCOMP	4	$(10, \dots, 10)$
32	NONSCOMP	10	$(10, \dots, 10)$
33	Extended DENSCHNB	1,000	$(1, \dots, 1)$
34	Extended DENSCHNB	50,000	$(1, \dots, 1)$
35	Extended DENSCHNB	100,000	$(1, \dots, 1)$
36	Extended Penalty Function U52	5	$(5, \dots, 5)$
37	Extended Penalty Function U52	10	$(5, \dots, 5)$
38	Extended Penalty Function U52	50	$(5, \dots, 5)$
39	Hager	5	$(1, \dots, 1)$
40	Hager	10	$(1, \dots, 1)$
41	Hager	50	$(1, \dots, 1)$
42	Booth	2	$(5, 5)$
43	Booth	2	$(10, 10)$
44	Sum Squares	1,000	$(0.1, \dots, 0.1)$
45	Sum Squares	10,000	$(0.1, \dots, 0.1)$
46	Sum Squares	100,000	$(0.1, \dots, 0.1)$
47	Zirilli or Aluffie-Petini's	2	$(1, 1)$
48	Zirilli or Aluffie-Petini's	2	$(-1, -1)$
49	Leon	2	$(-2, -2)$
50	Leon	2	$(-2, -2)$
51	Cube	2	$(4, 4)$
52	Cube	50	$(4, \dots, 4)$

Continued on next page

Continued from previous page

No.	Functions	Dimensions	Initial Points
53	Cube	100	$(4, \ldots, 4)$
54	Extended Maratos	10	$(-0.5, \ldots, -0.5)$
55	Extended Maratos	50	$(-0.5, \ldots, -0.5)$
56	Extended Maratos	100	$(-0.5, \ldots, -0.5)$
57	Generalized Tridiagonal 1	5	$(15, \ldots, 15)$
58	Generalized Tridiagonal 1	10	$(15, \ldots, 15)$
59	Generalized Tridiagonal 1	100	$(15, \ldots, 15)$
60	Trecanni	2	$(-1, 0.5)$
61	Trecanni	2	$(-5, 10)$
62	Zettl	2	$(0, 0)$
63	Zettl	2	$(10, 10)$
64	Shallow	1,000	$(1.001, \ldots, 1.001)$
65	Shallow	50,000	$(1.001, \ldots, 1.001)$
66	Shallow	100,000	$(1.001, \ldots, 1.001)$
67	Generalized Quartic	100	$(1.001, \ldots, 1.001)$
68	Generalized Quartic	5,000	$(1.001, \ldots, 1.001)$
69	Generalized Quartic	10,000	$(1.001, \ldots, 1.001)$
70	Quadratic QF2	10	$(0.5, \ldots, 0.5)$
71	Quadratic QF2	100	$(0.5, \ldots, 0.5)$
72	Quadratic QF2	1,000	$(0.5, \ldots, 0.5)$
73	Six Hump Camel	2	$(-1.5, -2)$
74	Six Hump Camel	2	$(-5, -10)$
75	Three Hump Camel	2	$(-1.5, -2)$
76	Three Hump Camel	2	$(-5, -10)$
77	Dixon and Price	1,000	$(0.5, \ldots, 0.5)$
78	Dixon and Price	10,000	$(0.5, \ldots, 0.5)$
79	Dixon and Price	100,000	$(0.5, \ldots, 0.5)$
80	POWER	10	$(3, \ldots, 3)$
81	POWER	50	$(3, \ldots, 3)$
82	POWER	500	$(3, \ldots, 3)$
83	Quadratic QF1	100	$(1, \ldots, 1)$
84	Quadratic QF1	1,000	$(1, \ldots, 1)$
85	Quadratic QF1	10,000	$(1, \ldots, 1)$
86	Generalized Tridiagonal 2	10	$(4, \ldots, 4)$
87	Generalized Tridiagonal 2	50	$(4, \ldots, 4)$
88	Generalized Tridiagonal 2	500	$(4, \ldots, 4)$
89	Extended Quadratic Penalty QP3	5	$(1, \ldots, 1)$
90	Extended Quadratic Penalty QP3	10	$(1, \ldots, 1)$
91	Extended Quadratic Penalty QP3	100	$(1, \ldots, 1)$
92	Extended Quadratic Penalty QP2	5	$(1, \ldots, 1)$
93	Extended Quadratic Penalty QP2	50	$(1, \ldots, 1)$
94	Extended Quadratic Penalty QP2	500	$(1, \ldots, 1)$
95	Extended Quadratic Penalty QP1	5	$(2, \ldots, 2)$
96	Extended Quadratic Penalty QP1	10	$(2, \ldots, 2)$
97	Extended Quadratic Penalty QP1	100	$(2, \ldots, 2)$
98	QUARTICM	1,000	$(4, \ldots, 4)$
99	QUARTICM	50,000	$(4, \ldots, 4)$
100	QUARTICM	100,000	$(4, \ldots, 4)$
101	Sphere	1,000	$(1, \ldots, 1)$

Continued on next page

Continued from previous page

No.	Functions	Dimensions	Initial Points
102	Sphere	10,000	$(1, \ldots, 1)$
103	Sphere	100,000	$(1, \ldots, 1)$
104	Quartic	4	$(0.5, \ldots, 0.5)$
105	Quartic	4	$(0.5, \ldots, 0.5)$
106	Matyas	2	$(1, 1)$
107	Matyas	2	$(20, 20)$
108	Diagonal 2	2	$(30, 30)$
109	Diagonal 2	5	$(30, \ldots, 30)$
110	Diagonal 2	10	$(30, \ldots, 30)$
111	Colville	4	$(1.2, \ldots, 1.2)$
112	Colville	4	$(-0.5, \ldots, -0.5)$
113	Price Function 4	2	$(-2, 3)$
114	Price Function 4	2	$(3, -2)$
115	Perturbed Quadratic	2	$(1, 1)$
116	Perturbed Quadratic	2	$(5, 5)$
117	Perturbed Quadratic	2	$(10, 10)$
118	Extended Hiebert	1,000	$(5, \ldots, 5)$
119	Extended Hiebert	10,000	$(5, \ldots, 5)$
120	Extended Hiebert	100,000	$(5, \ldots, 5)$
121	Linear Perturbed	100	$(0.1, \ldots, 0.1)$
122	Linear Perturbed	5,000	$(0.1, \ldots, 0.1)$
123	Linear Perturbed	50,000	$(0.1, \ldots, 0.1)$
124	Extended Block-Diagonal BD1	100	$(1.02, \ldots, 1.02)$
125	Extended Block-Diagonal BD1	5,000	$(1.02, \ldots, 1.02)$
126	Extended Block-Diagonal BD1	50,000	$(1.02, \ldots, 1.02)$
127	DENSCHNA	1,000	$(-1, \ldots, -1)$
128	DENSCHNA	10,000	$(-1, \ldots, -1)$
129	DENSCHNA	100,000	$(-1, \ldots, -1)$
130	DENSCHNB	100	$(10, \ldots, 10)$
131	DENSCHNB	5,000	$(10, \ldots, 10)$
132	DENSCHNB	50,000	$(10, \ldots, 10)$
133	DENSCHNC	100	$(1.5, \ldots, 1.5)$
134	DENSCHNC	5,000	$(1.5, \ldots, 1.5)$
135	DENSCHNC	50,000	$(1.5, \ldots, 1.5)$
136	DENSCHNF	100	$(50, \ldots, 50)$
137	DENSCHNF	5,000	$(50, \ldots, 50)$
138	DENSCHNF	50,000	$(50, \ldots, 50)$
139	HIMMELBG	10	$(1.5, \ldots, 1.5)$
140	HIMMELBG	50	$(1.5, \ldots, 1.5)$
141	HIMMELBG	100	$(1.5, \ldots, 1.5)$
142	HIMMELBH	10	$(0.8, \ldots, 0.8)$
143	HIMMELBH	50	$(0.8, \ldots, 0.8)$
144	HIMMELBH	100	$(0.8, \ldots, 0.8)$
145	DIAG-AUP1	10	$(-1, \ldots, -1)$
146	DIAG-AUP1	1,000	$(-1, \ldots, -1)$
147	DIAG-AUP1	10,000	$(-1, \ldots, -1)$
148	Strait	1,000	$(2, \ldots, 2)$
149	Strait	100,000	$(2, \ldots, 2)$
150	Strait	1,000,000	$(2, \ldots, 2)$

The numerical comparisons were conducted objectively using the standard Wolfe line search, where the parameter values for our proposed method are $\eta = 0.0001$, $\rho = 0.09$, and $\bar{t} = 0.3$, while the parameter values used for NHS+, HTTHSLS, TTCDDY, HTT, TTPRLS, HTHP and TTRMIL were kept consistent as specified in their respective studies. When $\|g_k\| \leq 10^{-6}$, all methods were terminated and would fail if the optimal value was not reached or the number of iterations exceeded 10,000. For the step length, α_k will be chosen when the search number of the standard Wolfe line search is more than 6. The overall numerical results for the all methods including the number of iterations and central processing unit times are provided at OVERALL NR DATA. Further assessment and visual representation of the results were carried out using the performance profile tool developed by Dolan and Mor'e [18], as depicted in Figure 1 and Figure 2, respectively.

Based on the numerical findings and the visual representations in Figure 1 and Figure 2, the proposed TTLC approach demonstrates several notable advantages. Specifically, it exhibits a high level of effectiveness in addressing 57% of the tested problems, showcasing superior efficiency compared to alternative methodologies under consideration. Moreover, the numerical performance of the TTLC method maintains a remarkable degree of stability, primarily attributed to the well-considered parameter choices outlined in equations (15), (16), and (22). When examining the numerical results from the two comparative analyses and their respective performance profiles, all five methods, in this context, have proven to be practically efficacious, particularly within the framework of these specific sets of numerical experiments. The efficacy of each approach is discernible by referencing Figure 1 and Figure 2, wherein the NHS+ method successfully resolves 91% of the problems, while HTTHSLS, TTCDDY, HTT, TTPRLS, HTHP and TTRMIL achieve 97%, 92%, 89%, 95%, 91% and 84%, respectively and TTLC attains a perfect 100%. From this standpoint, the TTLS method emerges as the most effective among the compared methodologies. Furthermore, it is worth highlighting that the TTLC method exhibits robust performance, especially when confronted with challenging problem instances.

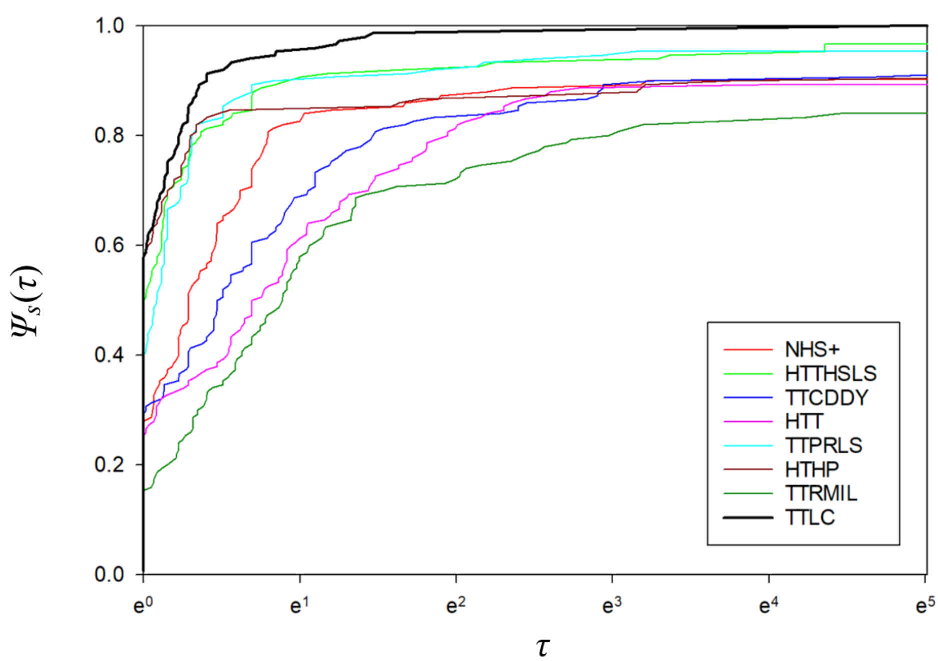

Figure 1: Performance profiles on NOI.

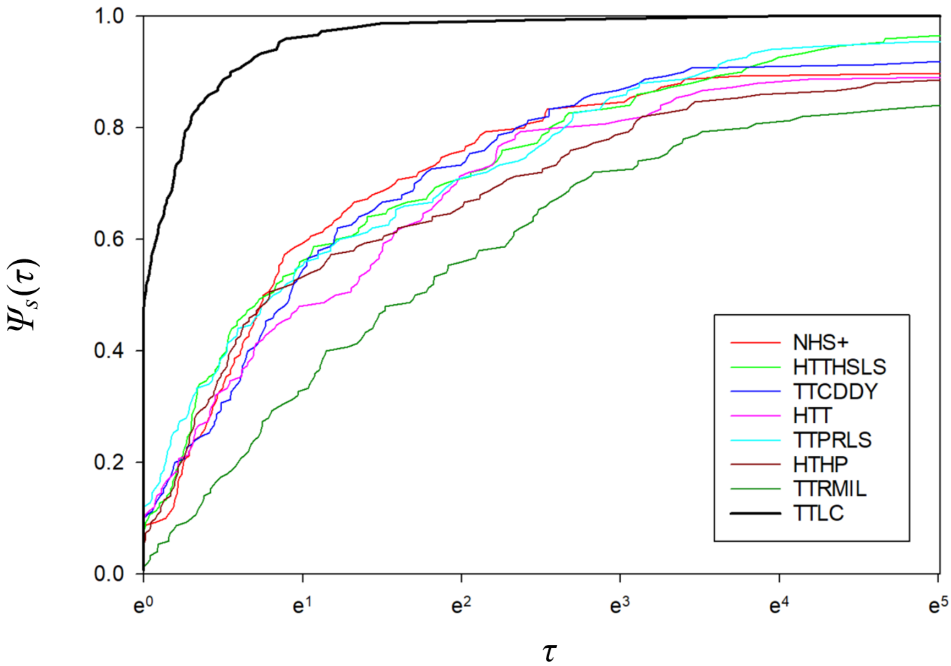

Figure 2: Performance profiles on CPU time.

5 Conclusions

In this paper, a new hybrid three-term CG algorithm is developed by combining the classical three-term LS and CD CG method by using standardization parameter. The standardization parameter is independent for any line searches. Regardless of whether a line search is employed, the search direction of the algorithm exhibits a satisfactory descent behavior and remains within defined bounds. Furthermore, the determination of step lengths is achieved through standard Wolfe line search. Demonstrating its effectiveness under certain assumptions, the global convergence is rigorously established and it owns the sufficient descent property independent of any line search technique. Based on the empirical evidence garnered from experimental numerical results which includes 150 test functions with various dimensions, it becomes evident that this innovative hybrid approach surpasses other existing methods in terms of both efficiency and robustness which has been visualized in the performance profiles. Therefore, the proposed method offers more effective and stable convergence across most of the problem scenarios examined.

Acknowledgement The authors are grateful to the editor and the anonymous reviewers for their comments and suggestions which improved this paper substantially. This paper was also presented at the 14^{th} International Fundamental Science Congress (14^{th} IFSC). Special thanks to the Department of Mathematics and Statistics, Faculty of Science, Universiti Putra Malaysia.

Conflicts of Interest The authors declare that they have no conflict of interest.

References

[1] A. B. Abubakar, P. Kumam, M. Malik, P. Chaipunya & A. H. Ibrahim (2021). A hybrid FR–DY conjugate gradient algorithm for unconstrained optimization with application in portfolio selection. *AIMS Mathematics*, *6*(6), 6506–6527. https://doi.org/10.3934/math.2021383.

[2] A. B. Abubakar, P. Kumam, M. Malik & A. H. Ibrahim (2022). A hybrid conjugate gradient based approach for solving unconstrained optimization and motion control problems. *Mathematics and Computers in Simulation*, *201*, 640–657. https://doi.org/10.1016/j.matcom.2021.05.038.

[3] N. Andrei (2008). Another hybrid conjugate gradient algorithm for unconstrained optimization. *Numerical Algorithm*, *47*, 143–156. https://doi.org/10.1007/s11075-007-9152-9.

[4] N. Andrei (2008). A hybrid conjugate gradient algorithm for unconstrained optimization as convex combination of Hestenes–Steifel and Dai–Yuan. *Studies in Informatics and Control*, *17*(1), 57–70.

[5] N. Andrei (2008). An unconstrained optimization test functions collection. *Advanced Modeling and Optimization*, *10*(1), 147–161. https://citeseerx.ist.psu.edu/document?repid=rep1&type=pdf&doi=0aa7264b4c5b14ddf091bfdc328c4fcb4049d4f4.

[6] N. Andrei (2009). Hybrid conjugate gradient algorithm for unconstrained optimization. *Journal of Optimization Theory and Applications*, *141*(2), 249–264. https://doi.org/10.1007/s10957-008-9505-0.

[7] N. Andrei (2013). A simple three-term conjugate gradient algorithm for unconstrained optimization. *Journal of Computational and Applied Mathematics*, *241*, 19–29. https://doi.org/10.1016/j.cam.2012.10.002.

[8] N. Andrei (2020). *Nonlinear Conjugate Gradient Methods for Unconstrained Optimization*. Springer, Cham 1st edition. https://doi.org/10.1007/978-3-030-42950-8.

[9] S. Babaie-Kafaki (2013). A hybrid conjugate gradient method based on quadratic relaxation of Dai–Yuan hybrid conjugate gradient parameter. *Optimization*, *62*(7), 929–941. https://doi.org/10.1080/02331934.2011.611512.

[10] S. Babaie-Kafaki, M. Fatemi & N. Mahdavi-Amiri (2011). Two effective hybrid conjugate gradient algorithms based on modified BFGS updates. *Numerical Algorithms*, *58*(3), 315–331. https://doi.org/10.1007/s11075-011-9457-6.

[11] S. Babaie-Kafaki, R. Ghanbari & N. Mahdavi-Amiri (2010). Two new conjugate gradient methods based on modified secant equations. *Journal of Computational and Applied Mathematics*, *234*(5), 1374–1386. https://doi.org/10.1016/j.cam.2010.01.052.

[12] S. Babaie-Kafaki & N. Mahdavi-Amiri (2013). Two modified hybrid conjugate gradient methods based on a hybrid secant equation. *Mathematical Modelling and Analysis*, *18*(1), 32–52. https://doi.org/10.3846/13926292.2013.756832.

[13] Y. H. Dai (2001). New properties of nonlinear conjugate gradient method. *Numerische Mathematik*, *89*, 83–98. https://doi.org/10.1007/PL00005464.

[14] Y. H. Dai & Y. Yuan (1999). A nonlinear conjugate gradient method with a strong global convergence property. *SIAM Journal on Optimization*, *10*(1), 177–182. https://doi.org/10.1137/S1052623497318992.

[15] J. Deepho, A. B. Abubakar, M. Malik & I. K. Argyros (2022). Solving unconstrained optimization problems via hybrid CD–DY conjugate gradient methods with applications. *Journal of Computational and Applied Mathematics*, *405*, Article ID: 113823. https://doi.org/10.1016/j.cam.2021.113823.

[16] J. E. Dennis Jr & J. J. Moré (1977). Quasi–Newton methods, motivation and theory. *SIAM Review*, *19*(1), 46–89. https://www.jstor.org/stable/2029325.

[17] S. S. Djordjević (2019). New hybrid conjugate gradient method as a convex combination of LS and FR methods. *Acta Mathematica Scientia*, *39*, 214–228. https://doi.org/10.1007/s10473-019-0117-6.

[18] E. D. Dolan & J. J. Moré (2002). Benchmarking optimization software with performance profiles. *Mathematical Programming*, *91*(2), 201–213. https://doi.org/10.1007/s101070100263.

[19] X. L. Dong, D. R. Han, R. Ghanbari, X. L. Li & Z. F. Dai (2017). Some new three–term Hestenes–Stiefel conjugate gradient methods with affine combination. *Optimization*, *66*(5), 759–776. https://doi.org/10.1080/02331934.2017.1295242.

[20] R. Fletcher & C. M. Reeves (1964). Function minimization by conjugate gradients. *The Computer Journal*, *7*(2), 149–154. https://doi.org/10.1093/comjnl/7.2.149.

[21] R. Fletcher (2013). *Practical Methods of Optimization*. John Wiley & Sons, New York 2nd edition. https://doi.org/10.1002/9781118723203.

[22] W. W. Hager & H. Zhang (2006). A survey of nonlinear conjugate gradient methods. *Pacific Journal of Optimization*, *2*(1), 35–58. https://people.clas.ufl.edu/hager/files/cg_survey.pdf.

[23] M. R. Hestenes & E. Stiefel (1952). Methods of conjugate gradients for solving linear systems. *Journal of Research of the National Bureau of Standards*, *49*(6), 409–436. https://doi.org/10.6028/JRES.049.044.

[24] M. Jamil & X. S. Yang (2013). A literature survey of benchmark functions for global optimization problems. *International Journal of Mathematical Modelling and Numerical Optimisation*, *4*(2), 150–194. https://doi.org/10.1504/IJMMNO.2013.055204.

[25] P. Kumam, A. B. Abubakar, M. Malik, A. H. Ibrahim, N. Pakkaranang & B. Panyanak (2023). A hybrid HS–LS conjugate gradient algorithm for unconstrained optimization with applications in motion control and image recovery. *Journal of Computational and Applied Mathematics*, *433*, Article ID: 115304. https://doi.org/10.1016/j.cam.2023.115304.

[26] M. Li (2018). A modified Hestense–Stiefel conjugate gradient method close to the memoryless BFGS quasi-Newton method. *Optimization Methods and Software*, *33*(2), 336–353. https://doi.org/10.1080/10556788.2017.1325885.

[27] M. Li (2020). A three term Polak–Ribiére–Polyak conjugate gradient method close to the memoryless BFGS quasi-Newton method. *Journal of Industrial and Management Optimization*, *16*(1), 245–260. https://doi.org/10.3934/jimo.2018149.

[28] J. K. Liu, Y. M. Feng & L. M. Zou (2018). Some three-term conjugate gradient methods with the inexact line search condition. *Calcolo*, *55*(2), Article ID: 16. https://doi.org/10.1007/s10092-018-0258-3.

[29] J. K. Liu & S. J. Li (2014). New hybrid conjugate gradient method for unconstrained optimization. *Applied Mathematics and Computation*, *245*, 36–43. https://doi.org/10.1016/j.amc.2014.07.096.

[30] Y. Liu & C. Storey (1991). Efficient generalized conjugate gradient algorithms, part 1: Theory. *Journal of Optimization Theory and Applications*, *69*, 129–137. https://doi.org/10.1007/BF00940464.

[31] M. Malik, I. M. Sulaiman, A. B. Abubakar, G. Ardaneswari & Sukono (2023). A new family of hybrid three-term conjugate gradient method for unconstrained optimization with application to image restoration and portfolio selection. *AIMS Mathematics*, *8*(1), 1–28. https://doi.org/10.3934/math.2023001.

[32] J. J. Moré, B. S. Garbow & K. E. Hillstrom (1981). Testing unconstrained optimization software. *ACM Transactions on Mathematical Software*, *7*(1), 17–41. https://doi.org/10.1145/355934.355936.

[33] J. Nocedal (1980). Updating quasi-Newton matrices with limited storage. *Mathematics of Computation*, *35*(151), 773–782. https://doi.org/10.1090/S0025-5718-1980-0572855-7.

[34] E. Polak & G. Ribiere (1969). Note sur la convergence de méthodes de directions conjugées. *Revue Française D'informatique et de Recherche Opérationnelle. Série Rouge*, *3*(R1), 35–43. http://www.numdam.org/item/M2AN_1969__3_1_35_0/.

[35] B. T. Polyak (1969). The conjugate gradient method in extremal problems. *USSR Computational Mathematics and Mathematical Physics*, *9*(4), 94–112. https://doi.org/10.1016/0041-5553(69)90035-4.

[36] M. J. D. Powell (1984). Nonconvex minimization calculations and the conjugate gradient method. In D. F. Griffiths (Ed.), *Numerical Analysis*, pp. 122–141. Springer, Berlin, Heidelberg. https://doi.org/10.1007/BFb0099521.

[37] N. Salihu, M. R. Odekunle, M. Y. Waziri & A. S. Halilu (2020). A new hybrid conjugate gradient method based on secant equation for solving large scale unconstrained optimization problems. *Iranian Journal of Optimization*, *12*(1), 33–44. https://dorl.net/dor/20.1001.1.25885723.2020.12.1.4.0.

[38] D. F. Shanno (1978). Conjugate gradient methods with inexact searches. *Mathematics of Operations Research*, *3*(3), 244–256. https://www.jstor.org/stable/3689494.

[39] L. Wang, M. Cao, F. Xing & Y. Yang (2020). The new spectral conjugate gradient method for large-scale unconstrained optimisation. *Journal of Inequalities and Applications*, *2020*(1), Article ID: 111. https://doi.org/10.1186/s13660-020-02375-z.

[40] X. Xu & F. Y. Kong (2016). New hybrid conjugate gradient methods with the generalized Wolfe line search. *SpringerPlus*, *5*(1), Article ID: 881. https://doi.org/10.1186/s40064-016-2522-9.

[41] X. Yang, Z. Luo & X. Dai (2013). A global convergence of LS–CD hybrid conjugate gradient method. *Advances in Numerical Analysis*, *2013*, Article ID: 517452. https://doi.org/10.1155/2013/517452.

[42] G. Zoutendijk (1970). Nonlinear programming computational methods. In J. Abadie (Ed.), *Integer and Nonlinear Programming*, pp. 37–86. North–Holland, Amsterdam.