Nonlinear Dynamics and Systems Theory, **25** (3) (2025) 266–287

Modified Parameter of the Dai–Liao Conjugacy Condition of the Conjugate Gradient Method with Some Applications

A. Jaradat [1*], S. Masmali [2], A. Alhawarat [1], R. Sabra [2], S. Ismail [3] and A. S. Al-Jawarneh [1]

[1] *Department of Mathematics, Faculty of Arts and Science, Amman Arab University, Amman, 11953, Jordan.*
[2] *Department of Mathematics, College of Science, Jazan University, Jazan, Saudi Arabia.*
[3] *Financial Mathematics Program, Faculty of Science and Technology, Universiti Sains Islam Malaysia, Bandar Baru Nilai, 71800, Negeri Sembilan, Malaysia.*
[3] *Risk and Analytics Research Group, Faculty of Science and Technology, Universiti Sains Islam Malaysia.*

Received: December 25, 2024; Revised: June 4, 2025

Abstract: This study introduces a novel modification of the conjugate gradient (CG) method by refining the Dai–Liao conjugacy parameter and incorporating a restart property. The proposed method, which is established in the Hestenes–Stiefel framework, is designed to ensure global convergence and satisfies the sufficient descent condition for both convex and non-convex functions. Utilizing the Lipschitz constant as a foundation, the method's efficiency and robustness were benchmarked against CG Descent across over 200 functions from the CUTEst library. Numerical experiments revealed superior performance in terms of CPU time, iterations, gradient evaluations, and function evaluations. Additionally, practical applications in heat conduction and image restoration demonstrate the method's versatility and effectiveness.

Keywords: *conjugate gradient; inexact line search; conjugacy condition; global convergence; CUTEst library.*

Mathematics Subject Classification (2020): 49M37, 65K05, 90C30.

* Corresponding author: mailto:a.jaradat@aau.edu.jo

1 Introduction

Conjugate gradient (CG) methods have been widely used for solving nonlinear unconstrained optimization problems due to their low memory requirements for implementation. Moreover, CG methods have been used in many applications such as regression analysis, image restoration, electrical circuits, and many others.

The CG method is used to determine optimal solutions for the following optimization problem:

$$\min f(x), \quad x \in \mathbb{R}^n,$$

where $f : \mathbb{R}^n \to \mathbb{R}$ is a continuously differentiable function, and its gradient $\nabla f(x_k) = g_k = g(x_k)$ should exist. From the starting point (arbitrary or standard) $x_1 \in \mathbb{R}^n$, the CG method generates a sequence of vectors x_k by the iterative rule

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

in which x_k represents the present iteration and $\alpha_k > 0$ represents a step size obtained from the exact line search or an inexact line search. The search direction d_k of the CG method is defined by

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

where β_k is the update parameter. The following exact line search can be utilized to obtain the step size α_k:

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

However, Eq.(3) is computationally expensive because it requires unidimensional optimization to achieve the step size and many iterations to reach convergence. To avoid this problem, the inexact line search is a dominant approach in computing the step size. The most popular inexact line search is the strong Wolfe–Powell (SWP) line search [1,2], which is defined as

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

$$|g(x_k + \alpha_k d_k)^T d_k| \leq \sigma |g_k^T d_k| \tag{5}$$

so that $0 < \delta < \sigma < 1$.

A version of the Wolfe–Powell line search is the weak Wolfe–Powell (WWP) line search, which is defined by (4) and

$$g(x_k + \alpha_k d_k)^T d_k \geq \sigma g_k^T d_k.$$

The most famous classical formulae of the CG methods are the Hestenes–Stiefel (HS) [3], Fletcher–Reeves (FR) [4], and Polak–Ribiere–Polyak (PRP) [5] methods, which are defined by the following update parameters, respectively:

$$\beta_k^{HS} = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}}, \quad \beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}, \quad \text{and}$$

$$\beta_k^{PRP} = \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}, \quad \text{where} \quad y_{k-1} = g_k - g_{k-1}.$$

Powell in [6] provided a counterexample showing that there exists a non-convex function for which the PRP and HS methods fail to satisfy the convergence properties even if the exact line search is employed. Powell recommended the use of nonnegative paremeters β_k^{HS} and β_k^{PRP} to achieve the convergence properties of the CG method. Gilbert and Nocedal [7] proved that the nonnegative PRP or HS method defined by $\beta_k = \max\{\beta_k^{PRP}, 0\}$, is globally convergent with arbitrary line searches.

The descent condition (downhill condition) plays a crucial role in the convergence of the CG method and its robustness, and it is defined by

$$g_k^T d_k < 0. \tag{6}$$

Al-Baali [8] proposed another version of the downhill condition called the sufficient descent condition, which also plays a significant role in the convergence of the CG method. Al-Baali proposed the condition

$$g_k^T d_k \le -c\|g_k\|^2 \quad \forall k \in \mathbb{N} \tag{7}$$

to establish the global convergence properties of β_k^{FR}. More precisely, if there exists a constant $c > 0$ satisfying (7), then the search direction d_k guarantees the sufficient descent condition.

Based on the quasi-Newton method, the Broyden–Fletcher–Goldfarb–Shanno (BFGS) method and the limited-memory BFGS (LBFGS) method, and using Eq.(2), Dai and Liao [9] proposed the conjugacy condition

$$d_k^T y_{k-1} = -t g_k^T s_{k-1} \tag{8}$$

such that $s_{k-1} = x_k - x_{k-1}$ and $t \ge 0$. In the case of $t = 0$, Eq.(8) is considered as the classical conjugacy condition. Using Eqs. (2) and (8), Dai and Liao [9] proposed the following CG formula:

$$\beta_k^{DL} = \frac{g_k^T y_{k-1}}{d_{k-1}^T y_{k-1}} - t\frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}. \tag{9}$$

However, β_k^{DL} cannot satisfy the descent condition and convergence properties similar to β_k^{PRP} and β_k^{HS} because β_k^{DL} is not nonnegative in general. Thus, Dai and Liao [9] replaced the formula (9) by

$$\beta_k^{DL+} = \max\{\beta_k^{HS}, 0\} - t\frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}. \tag{10}$$

However, β_k^{DL+} cannot satisfy the descent property in some cases. Therefore, Dai and Liao [9] restarted (10) using a negative gradient (steepest descent) when β_k^{DL+} fails to satisfy inequality (7). Another method for determining the optimal parameter t was proposed by Babaie-Kafaki and Ghanbari [10,11], where they rewrote the search direction (Eq.(2)) with β_k^{DL}, and based on Perry [12], as follows: $d_{k+1} = -Q_{k+1} g_{k+1}$, where $Q_{k+1} = I - \frac{s_k y_k^T}{s_k^T y_k} + t\frac{s_k s_k^T}{s_k^T y_k}$. Babaie-Kafaki and Ghanbari [10] proposed the following adaptive choices for t:

$$t = \frac{s_k y_k^T}{\|s_k\|^2} + \frac{\|y_k\|}{\|s_k\|} \quad \text{and} \quad t = \frac{\|y_k\|}{\|s_k\|}.$$

Andrei in [13] proposed a CG method with the parameter

$$\beta_k^{DL*} = \max\left\{\frac{y_k^T g_k}{y_k^T s_k}, 0\right\} - t_k^* \frac{s_k^T g_{k+1}}{y_k^T s_k},$$

where $t_k^* = \frac{y_k^T s_k}{\|s_k\|^2}$. Hager and Zhang [14, 15] presented a modified CG parameter that satisfies the descent property for any inexact line search with $g_k^T d_k \leq -(7/8)\|g_k\|^2$. This new version of the CG method is globally convergent whenever the line search satisfies the WWP line search. This formula is expressed by

$$\beta_k^{HZ} = \max\{\beta_k^N, \eta_k\},$$

where $\beta_k^N = \frac{1}{d_k^T y_k}\left(y_k - 2d_k \frac{\|y_k\|^2}{d_k^T y_k}\right)^T g_k$, $\eta_k = -\frac{1}{\|d_k\| \min\{\eta, \|g_k\|\}}$, and $\eta > 0$ is a constant.

Note that, if $t = 2\frac{\|y_k\|^2}{s_k^T y_k}$, then $\beta_k^N = \beta_k^{DL}$. Zhang *et al.* [16] proposed a new parameter for Eq.(9) as follows:

$$t = \frac{\|y_k\|^2}{s_k^T y_k} - \frac{1}{4}\frac{s_k^T y_k}{\|s_k\|^2}.$$

Yao *et al.* [2] proposed three terms of CG with a new choice of t as follows:

$$d_{k+1} = -g_{k+1} + \left(\frac{g_k^T y_k - t_k g_{k+1}^T s_k}{y_k^T d_k}\right) d_k + \frac{g_{k+1}^T d_k}{y_k^T d_k} y_k.$$

Based on the SWP line search, Yao *et al.* [2] selected t_k to satisfy the descent condition

$$t_k > \frac{\|y_k\|^2}{y_k^T s_k}.$$

Yao *et al.* [2] also proposed a theorem stating that if t_k is close to $\frac{\|y_k\|^2}{y_k^T s_k}$, then the search direction results in a zigzag search path. Therefore, they selected the following choice for t_k:

$$t_k = 1 + 2\frac{\|y_k\|^2}{y_k^T s_k}.$$

Al-Baali *et al.* [17] proposed a new CG version called the G3TCG that offers many selections of CG parameters. They found that the G3TCG method is more efficient than β_k^{HZ} in some cases and competitive in some other cases.

In this research, we propose a new CG iterative formula based on a modified parameter of the Dai–Liao conjugacy condition of the CG method with the restart property. The convergence of the proposed modified CG method is analyzed under standard assumptions. Numerical experiments are performed to illustrate the superiority of the proposed method.

The highlighted results are achieved in the subsequent sections organized in the following manner. A novel CG formula is proposed in Section 2, as well as underlying motivation. The convergence analysis of the modified CG method is presented in Section 3. Section 4 includes the results of numerical experiments and their discussion.

2 Proposed CG Formula and Its Motivation

The CG method with β_k^{DL} cannot satisfy the descent condition, but β_k^{DL} inherits the conjugacy condition. To improve the properties of β_k^{DL}, we used β_k^{AZPRP} as presented by Alhawarat et al. [23] to propose a new nonnegative CG method that can satisfy the sufficient descent condition and global convergence properties with the SWP line search as follows:

$$\beta_k^{AZPRP} = \begin{cases} \frac{\|g_k\|^2 - \mu_k \left|g_k^T g_{k-1}\right|}{\|g_{k-1}\|^2}, & \|g_k\|^2 > \mu_k \left|g_k^T g_{k-1}\right|, \\ 0, & \text{otherwise.} \end{cases}$$

The proposed CG update parameter is a modification of β_k^{DL} and β_k^{HS}, with the restart criterion depending on the Lipschitz constant used in the study conducted by Alhawarat *et al.* [23]. The modified formula is expressed as

$$\beta_k^{AZHS} = \begin{cases} \frac{\|g_k\|^2 - \mu_k \left|g_k^T g_{k-1}\right|}{d_{k-1}^T y_{k-1}} - \frac{1}{\alpha_k} \mu_k \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}, & \|g_k\|^2 > \mu_k \left|g_k^T g_{k-1}\right|, \\ -\frac{1}{\alpha_k} \mu_k \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}, & \text{otherwise,} \end{cases} \tag{11}$$

where $\|\cdot\|$ represents the Euclidean norm and μ_k is defined as follows:

$$\mu_k = \frac{\|s_{k-1}\|}{\|y_{k-1}\|}.$$

In the first case of the equality (11), we can note that

$$\beta_k^{AZHS} \leq \frac{\|g_k\|^2}{d_{k-1}^T y_{k-1}} - \frac{1}{\alpha_k} \mu_k \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}. \tag{12}$$

It is worth noting that the formula (11) inherits the advantages of β_k^{DL}, β_k^{HS}, and β_k^{AZPRP}. Moreover, as we will see in the next sections, the new formula satisfies the descent condition and the global convergence properties. The usage of the proposed parameter β_k^{AZHS} in (11) leads to the novel CG method described in Algorithm 2.1.

Algorithm 2.1 CG method based on β_k^{AZHS}.

Step 1 Set a starting point x_1. The initial point can be arbitrary or standard for scientific functions. The initial search direction is the negative gradient, i.e., $d_1 = -g_1$. Let $k \leftarrow 1$.

Step 2 If the stopping condition is satisfied, then stop.

Step 3 Compute the search direction d_k based on Eq.(2) using Eq.(11).

Step 4 Compute the step size α_k using Eqs.(4) and (5).

Step 5 Update x_{k+1} based on Eq.(1).

Step 6 Set $k \leftarrow k + 1$ and go to Step 2.

3 Convergence Analysis of β_k^{AZHS}

To perform the convergence analysis of the modified CG method, we consider the following assumptions.

Assumption 1

A. The level set $\Phi = \{x|f(x) \leq f(x_1)\}$ is bounded. In other words, a positive constant B exists so that

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

B. In some neighborhood P of Φ, f is continuously differentiable, and its gradient is Lipschitz continuous. In other words, $\forall x, y \in P, \exists L > 0$ so that

$$\|g(x) - g(y)\| \leq L \|x - y\|.$$

This assumption implies that there exists a positive constant $\hat{\gamma}$ such that

$$\|g(x)\| \leq \widehat{\gamma}, \quad \forall x \in P.$$

Theorem 3.1 *Let the sequences $\{g_k\}$ and $\{d_k\}$ be obtained using Eqs.(1) and (2), and β_k^{AZHS}, where α_k is computed using the SWP line search in Eqs.(4) and (5). If $\sigma \in (0, 0.5)$, then the descent condition provided in (7) holds.*

Proof. The proof is carried out for two cases.

Case 1: $\|g_k\|^2 > \mu_k \left|g_k^T g_{k-1}\right|$.

This assumption implies

$$\beta_k^{AZHS} = \frac{\|g_k\|^2 - \mu_k \left|g_k^T g_{k-1}\right|}{d_{k-1}^T y_{k-1}} - \frac{1}{\alpha_k} \mu_k \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}.$$

Multiplying (2) by g_k^T, we can conclude that

$$\begin{aligned} g_k^T d_k &= g_k^T(-g_k + \beta_k d_{k-1}) = -\|g_k\|^2 + \beta_k g_k^T d_{k-1} \\ &\leq -\|g_k\|^2 + \frac{\|g_k\|^2}{\left|d_{k-1}^T y_{k-1}\right|} \left|g_k^T d_{k-1}\right| - \mu_k \frac{\left\|g_k^T d_{k-1}\right\|^2}{d_{k-1}^T y_{k-1}}. \end{aligned}$$

The usage of the SWP line search leads to the inequality

$$\frac{\left|g_k^T d_{k-1}\right|}{\left|d_{k-1}^T y_{k-1}\right|} \leq \frac{\sigma}{1 - \sigma}.$$

Thus,

$$g_k^T d_k \leq -\|g_k\|^2 + \frac{\sigma \|g_k\|^2}{(1 - \sigma)} = -\|g_k\|^2 \left(1 - \frac{\sigma}{1 - \sigma}\right) = -c\|g_k\|^2.$$

Case 2: $\|g_k\|^2 \leq \mu_k \left|g_k^T g_{k-1}\right|$.

This assumption implies

$$\beta_k^{AZHS} = -\frac{1}{\alpha_k} \mu_k \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}$$

and further

$$\begin{aligned} g_k^T d_k &= g_k^T(-g_k + \beta_k d_{k-1}) = -\|g_k\|^2 + \beta_k g_k^T d_{k-1} \\ &\le -\|g_k\|^2 + \left(-\frac{\mu_k}{\alpha_{k-1}} \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}\right) g_k^T d_{k-1} \\ &= -\|g_k\|^2 - \mu_k \frac{\left\|g_k^T d_{k-1}\right\|^2}{d_{k-1}^T y_{k-1}}. \end{aligned}$$

Since the SWP line search is used, it follows that $d_{k-1}^T y_{k-1} > 0$, and further

$$g_k^T d_k \le -c\|g_k\|^2,$$

which completes the proof. □

Lemma 3.1 shows that if $L > 1$, then equation (13) holds. Note that if $L \ll 1$, then $\|g_k\|^2 > \mu_k \left|g_k^T g_{k-1}\right|$ can not be satisfied.

Lemma 3.1 *If* $\|g_k\|^2 > \mu_k \left|g_k^T g_{k-1}\right|$ *and* $L > 1$, *then*

$$\|g_k\|^2 - \frac{1}{L}\left|g_k^T g_{k-1}\right| \le L\left|\|g_k\|^2 - \left|g_k^T g_{k-1}\right|\right|. \tag{13}$$

***Proof*.** The proof is performed using contradiction. Suppose

$$\|g_k\|^2 - \frac{1}{L}\left|g_k^T g_{k-1}\right| > L\left|\|g_k\|^2 - \left|g_k^T g_{k-1}\right|\right|,$$

and divide both sides by L:

$$\frac{\|g_k\|^2}{L} - \frac{1}{L^2}\left|g_k^T g_{k-1}\right| > \left|\|g_k\|^2 - \left|g_k^T g_{k-1}\right|\right|. \tag{14}$$

Using Assumption 1, the following relationship is derived:

$$\|g_k\|^2 > \mu_k \left|g_k^T g_{k-1}\right| > \frac{1}{L}\left|g_k^T g_{k-1}\right|.$$

If $L > 1$, we conclude that inequality (14) is not true, which results in a contradiction. Thus, inequality (13) holds. □

The following Lemma 3.2 indicates the step length always has a lower bound.

Lemma 3.2 [25]. *Suppose that the objective function satisfies Assumption 1. If the step length* α_k *fulfills the SWP line search conditions* (4) *and* (5), *then*

$$\alpha_k \ge \frac{(1-\sigma)\left|g_k^T d_k\right|}{L\|d_k\|^2}.$$

Lemma 3.3 *Let Assumption 1 hold. Consider any form of Eqs.*(1) *and* (2) *with the step size* α_k *satisfying the SWP line search, where the search direction* d_k *is descent. The following inequality is obtained:*

$$\sum_{k=0}^{\infty} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < \infty. \tag{15}$$

The condition presented in inequality (15) is called the Zoutendijk condition [25] and plays an important role in proving the convergence properties of the CG method. We use the contradiction technique with (15) to prove $\liminf_{k\to\infty} \|g_k\| = 0$.

Moreover, (15) holds for the exact and SWP line searches. By substituting (7) into (15), we obtain

$$\sum_{k=0}^{\infty} \frac{\|g_k\|^4}{\|d_k\|^2} < \infty. \tag{16}$$

Lemma 3.4 *If* $\|g_k\|^2 > \mu_k \left|g_k^T g_{k-1}\right|$ *is satisfied, then* $\mu_k = \frac{\|s_{k-1}\|}{\|y_{k-1}\|}$ *is bounded above and below.*

Proof. Since $\|g_k\|^2 > \mu_k \left|g_k^T g_{k-1}\right| > \frac{1}{L} \left|g_k^T g_{k-1}\right|$, based on Assumption 1, it follows that $0 < \mu_k \leq E$, where E denotes a positive constant. Moreover, if $y_{k+1} = 0$, this means $x_{k+1} = x_k$ and it is known that $x_{k+1} = x_k + \alpha_k d_k$. Thus, $\alpha_k d_k = 0$. However, by Lemma 3.2, we conclude that $\alpha_k > 0$. This means that $d_k = 0$. The usage of Theorem 3.1 and Lemma 3.3 leads to a contradiction. □

Dai *et al.* [26] presented the following Theorem 3.2, which is also useful for proving the global convergence properties of CG methods.

Theorem 3.2 *Suppose that Assumption 1 holds. Consider any CG method in the form of Eqs.*(1) *and* (2), *where* d_k *is a descent direction and* α_k *is obtained using the SWP line search. If*

$$\sum_{k\geq 1}^{\infty} \frac{1}{\|d_k\|^2} = \infty,$$

then

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

Global convergence properties for the convex functions

In the following theorem, if $f(x)$ is a uniformly convex function, then the CG method satisfies β_k^{AZHS} strong global convergence properties.

Theorem 3.3 *Suppose that Assumption 1 holds. Consider the CG method in the form of Eqs.*(1) *and* (2) *with* β_k^{AZHS}, $L > 1$, *and* d_k *as a descent direction, where* α_k *is obtained using the SWP line search. If* $f(x)$ *is a uniformly convex function, then* $\liminf_{k\to\infty} \|g_k\| = 0$.

Proof. Since the function $f(x)$ is uniformly convex, there exists a positive constant ϖ satisfying

$$\varpi \|x-y\|^2 \leq (\nabla f(x) - \nabla f(y))^T (x-y)$$

for all $x, y \in \mathcal{P}$. Thus,

$$d_{k-1} y_{k-1} \geq \varpi \alpha_{k-1} \|d_{k-1}\|^2 \tag{17}$$

and

$$\begin{aligned} \beta_k^{AZHS} &= \frac{\|g_k\|^2 - \mu_k \left|g_k^T g_{k-1}\right|}{d_{k-1}^T y_{k-1}} - \frac{\mu_k}{\alpha_{k-1}} \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}} \\ &\leq \frac{\|g_k\|^2 - \frac{1}{L} \left|g_k^T g_{k-1}\right|}{d_{k-1}^T y_{k-1}} - \frac{\mu_k}{\alpha_{k-1}} \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}. \end{aligned}$$

An application of inequalities (17) and (13) gives

$$\begin{aligned}
\beta_k^{AZHS} &\leq \frac{L\,\|g_k\|\,(\|g_k - g_{k-1}\|)}{\varpi\alpha_{k-1}\|d_{k-1}\|^2} + E\frac{\left|g_k^T s_{k-1}\right|}{\varpi\alpha_{k-1}^2\|d_{k-1}\|^2} \\
&\leq \frac{L\,\|g_k\|\,\|g_k - g_{k-1}\|}{\varpi\alpha_{k-1}\|d_{k-1}\|^2} + E\frac{\|g_k\|\,\|s_{k-1}\|}{\varpi\alpha_{k-1}^2\|d_{k-1}\|^2}.
\end{aligned}$$

Applying Assumption 1, we obtain

$$\begin{aligned}
\beta_k^{AZHS} &\leq \frac{L^2\,\|g_k\|\,\alpha_{k-1}\,\|d_{k-1}\|}{\varpi\alpha_{k-1}\|d_{k-1}\|^2} + E\frac{\|g_k\|\,\|d_{k-1}\|}{\varpi\alpha_{k-1}\|d_{k-1}\|^2} \\
&\leq \frac{L^2\,\|g_k\|}{\varpi\,\|d_{k-1}\|} + E\frac{\|g_k\|}{\varpi\alpha_{k-1}\,\|d_{k-1}\|}.
\end{aligned}$$

Based on Eq.(2), it can be obtained that

$$\begin{aligned}
\|d_k\| &\leq \|g_k\| + |\beta_k|\,\|d_{k-1}\| \\
&\leq \|g_k\| + \frac{\|g_k\|}{\|d_{k-1}\|}\left(\frac{L^2}{\varpi} + \frac{E}{\varpi\alpha_{k-1}}\right)\|d_{k-1}\| \\
&\leq \hat{\gamma}\left(1 + \left(\frac{L^2}{\varpi} + \frac{E}{\varpi\alpha_{k-1}}\right)\right).
\end{aligned}$$

Thus, Theorem 3.2 leads to the conclusion

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

and completes the proof. □

Global convergence for β_k^{AZHS} with the SWP line search for general functions

Using Property(*) and some lemmas, Gilbert and Nocedal [7] proved the global convergence of nonnegative PRP and HS methods. Because our modification is nonnegative and satisfies Property(*), by using the other lemmas presented below, we perform our proof in the same way as in [7]. This property is defined as follows.

Property(*)

Consider any CG method in the form of Eqs.(1) and (2). Assume

$$0 < \gamma \leq \|g_k\| \leq \hat{\gamma} \tag{18}$$

for all $k \geq 1$. The CG method then inherits Property(*) if for $\forall k$, there exist constants $b > 1$ and $\lambda > 0$ such that $|\beta_k| \leq b$ and $\|s_k\| \leq \lambda$, which implies that $|\beta_k| \leq \frac{1}{2b}$.

Lemma 3.5 shows that β_k^{AZHS} satisfies Property(*).

Lemma 3.5 *Consider a CG method in the form of Eqs.*(1) *and* (2) *using β_k^{AZHS} with $L > 1$. Lemma 3.1 holds true, then β_k^{AZHS} satisfies Property(*).*

Proof. Let $b = \frac{2L\alpha_{k-1}\widehat{\gamma}^2 + B\widehat{\gamma}}{\alpha_{k-1}L(1-\sigma)c\gamma^2} \geq 1$, and let $\lambda \leq \frac{(1-\sigma)c\gamma^2}{2(L^2+\frac{E}{\alpha_{k-1}})\hat{\gamma}b}$. Then the following inequality holds:

$$\beta_k^{AZHS} \leq \frac{\|g_k\|^2 - \mu_k\left|g_k^T g_{k-1}\right|}{d_{k-1}^T y_{k-1}} - \frac{\mu_k}{\alpha_{k-1}}\frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}.$$

Inequalities (13) and (18) are a basis for the inequalities

$$\begin{aligned}\beta_k^{AZHS} &\leq \frac{\|g_k\|^2 + \left|g_k^T g_{k-1}\right|}{d_{k-1}^T y_{k-1}} + \frac{E}{\alpha_{k-1}} \frac{\left|g_k^T s_{k-1}\right|}{d_{k-1}^T y_{k-1}} \\ &\leq \frac{2\hat{\gamma}^2}{(1-\sigma)c\gamma^2} + \frac{EB\hat{\gamma}}{\alpha_{k-1}L(1-\sigma)c\gamma^2} = \frac{2L\alpha_{k-1}\hat{\gamma}^2 + EB\hat{\gamma}}{\alpha_{k-1}(1-\sigma)c\gamma^2} \\ &= b > 1.\end{aligned}$$

Further, $\|s_k\| \leq \lambda$ gives

$$\begin{aligned}\beta_k^{AZHS} &\leq \frac{\|g_k\|^2 - \mu_k \left|g_k^T g_{k-1}\right|}{d_{k-1}^T y_{k-1}} - \frac{\mu_k}{\alpha_{k-1}} \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}} \\ &\leq \frac{L\|g_k\| \|g_k - g_{k-1}\|}{d_{k-1}^T y_{k-1}} + \frac{E}{\alpha_{k-1}} \frac{\|g_k\| \|s_{k-1}\|}{d_{k-1}^T y_{k-1}} \\ &\leq \frac{L^2 \|g_k\| \|s_{k-1}\|}{d_{k-1}^T y_{k-1}} + \frac{E}{\alpha_{k-1}} \frac{\|g_k\| \|s_{k-1}\|}{d_{k-1}^T y_{k-1}} \\ &\leq \frac{(L^2 + \frac{E}{\alpha_{k-1}}) \|g_k\| \|s_{k-1}\|}{d_{k-1}^T y_{k-1}} \\ &\leq \frac{(L^2 + \frac{E}{\alpha_{k-1}})\hat{\gamma}\lambda}{(1-\sigma)c\gamma^2} = \frac{1}{2b}.\end{aligned}$$

Thus, the proof is complete. □

Lemma 3.6 and Lemma 3.7 are similar to Lemma 4.1 and Lemma 4.2 presented by Gilbert and Nocedal in [7].

Lemma 3.6 *Assume that Assumption 1 holds and the sequences* $\{g_k\}$ *and* $\{d_k\}$ *are generated using Algorithm 1, where the step size* α_k *is computed via the SWP line search so that the sufficient descent condition holds. If* $\beta_k \geq 0$, *there exists a constant* $\gamma > 0$ *such that* $\|g_k\| > \gamma$ *for all* $k \geq 1$. *Then* $d_k \neq 0$ *and*

$$\sum_{k=0}^{\infty} \|u_{k+1} - u_k\|^2 < \infty,$$

where $u_k = \frac{d_k}{\|d_k\|}$.

Proof. The assumption $d_k = 0$, based on the sufficient descent condition, leads to $g_k = 0$. So, $d_k \neq 0$ as well as

$$\|g_k\| \geq \gamma, \quad \text{where } \gamma > 0. \tag{19}$$

Eq.(11) can be divided into two parts as follows:

$$\beta_k^{(1)} = \frac{\|g_k\|^2 - \mu_k \left|g_k^T g_{k-1}\right|}{d_{k-1}^T y_{k-1}}, \quad \beta_k^{(2)} = -\frac{\mu_k}{\alpha_{k-1}} \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}.$$

Then the following values can be defined:

$$\xi = \frac{\left\| -g_k + \beta_k^{(2)} d_{k-1} \right\|}{\|d_k\|}, \quad \zeta = \frac{\beta_k^{(1)} \|d_{k-1}\|}{\|d_k\|}.$$

From the definition of u_k, it can be derived that

$$u_k = \frac{d_k}{\|d_k\|} = \frac{-g_k + (\beta_k^{(1)} + \beta_k^{(2)}) d_{k-1}}{\|d_k\|} = \xi + \zeta \frac{d_{k-1}}{\|d_k\|} = \xi + \zeta u_{k-1}.$$

Since u_k is a unit vector, it follows that $\|\xi\| = \|u_k - \zeta u_{k-1}\| = \|\zeta u_k - u_{k-1}\|$. By using the triangle inequality and $\zeta > 0$, one concludes

$$\|u_k - u_{k-1}\| = 2\|\xi\|. \tag{20}$$

Using the definition of ξ, we obtain

$$\|\xi\| \|d_k\| = \left\| -g_k + \beta_{k-1}^{(2)} d_{k-1} \right\| \le \|g_k\| + \left\| \beta_{k-1}^{(2)} \right\| \|d_{k-1}\|. \tag{21}$$

By using the equations of SWP (Eq.(5)) and line search (Eq.(6)), one gets

$$d_{k-1}^T y_{k-1} \ge (\sigma - 1) g_{k-1}^T d_{k-1}, \quad \left| \frac{g_k^T d_{k-1}}{d_{k-1}^T y_{k-1}} \right| \le \left(\frac{\sigma}{1-\sigma} \right).$$

Thus,

$$\beta_k^{(2)} = -\frac{\mu_k}{\alpha_{k-1}} \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}} \le \frac{E}{\alpha_{k-1}} \frac{\left| g_k^T s_{k-1} \right|}{d_{k-1}^T y_{k-1}} \le \frac{E}{\alpha_{k-1}} \frac{\|g_k\| \|s_{k-1}\|}{d_{k-1}^T y_{k-1}}.$$

By using Eq.(21), we obtain the following:

$$\begin{aligned}\|\xi\| \|d_k\| = \left\| -g_k + \beta_{k-1}^{(2)} d_{k-1} \right\| &\le \|g_k\| + \frac{E}{\alpha_{k-1}} \left| \frac{g_k^T d_{k-1}}{d_{k-1}^T y_{k-1}} \right| \|s_{k-1}\| \\ &\le \gamma + \frac{E}{\alpha_{k-1}} \left(\frac{\sigma}{1-\sigma} \right) B.\end{aligned}$$

The application of Eq.(20) leads to

$$\|u_k - u_{k-1}\| = 2\|\xi\| = 2 \frac{\gamma + \frac{E}{\alpha_{k-1}} \left(\frac{\sigma}{1-\sigma} \right) B}{\|d_k\|},$$

$$\|u_k - u_{k-1}\|^2 = 4 \frac{\left(\gamma + \frac{E}{\alpha_{k-1}} \left(\frac{\sigma}{1-\sigma} \right) B \right)^2}{\|d_k\|^2}.$$

Utilizing Eq.(19), we obtain the following:

$$\sum_{k=1}^{\infty} \frac{1}{\|d_k\|^2} \le \infty,$$

which completes the proof. $\square$

Lemma 3.7 *Assume that Assumption 1 holds and the sequences $\{g_k\}$ and $\{d_k\}$ are generated using Algorithm 1, where α_k is computed via the WWP line search so that the sufficient descent condition given in Eq.*(7) *holds and consider that the method satisfies Property(*). Suppose also that Eq.*(19) *holds. Then there exists a constant $\lambda > 0$ so that for any $\Delta \in \mathbb{N}$ and any index k_0, there exists an index $k > k_0$ that satisfies the following inequality:*

$$\left|\kappa_{k,\Delta}^{\lambda}\right| > \frac{\lambda}{2},$$

where $\kappa_{k,\Delta}^{\lambda} = \{i \in \mathbb{N} : k \leq i \leq k+\Delta-1, \|s_i\| > \lambda\}$, $\mathbb{N}$ denotes the set of positive integers, and $\left|\kappa_{k,\Delta}^{\lambda}\right|$ denotes the number of elements in $\kappa_{k,\Delta}^{\lambda}$.

From Lemmas 3.5, 3.6 and 3.7, the convergence properties of Algorithm 1 with the SWP line search can be satisfied in a manner similar to that used in Theorem 3.6 presented by Gilbert and Nocedal [7]. Therefore, the proof of the following theorem is omitted.

Theorem 3.4 *Assume that the sequences $\{g_k\}$ and $\{d_k\}$ are generated using Eqs.*(1) *and* (2) *with the CG formula β_k^{AZHS}, and let the step length satisfy Eqs.*(4) *and* (5)*. If Lemmas 3.5, 3.6, and 3.7 are true, then* $\liminf_{k\to\infty}\|g_k\| = 0$.

Note that if Lemma 3.1 does not hold true, then it is enough to show that

$$\beta_k^{AZHS} = \frac{\|g_k\|^2 - \mu_k \left|g_k^T g_{k-1}\right|}{d_{k-1}^T y_{k-1}}$$

satisfies Property (*) similar to Lemma 3.3 in [23].

The following theorem shows that if the second case of equation (11) holds, i.e.,

$$\beta_k^{AZHS} = -\frac{1}{\alpha_k}\mu_k \frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}, \tag{22}$$

then we will obtain the result stated in Theorem 3.6.

Theorem 3.5 *Assume that Assumption 1 holds. Consider the conjugate gradient method in* (1) *and* (2) *with equation* (22)*, where d_k is a descent direction and α_k is obtained by the strong Wolfe line search. Then* $\liminf_{k\to\infty}\|g_k\| = 0$.

Proof. We will prove this theorem by contradiction. Suppose Theorem 3.5 is not true. Then equation (19) holds and

$$\begin{aligned}
\|d_k\|^2 &= \|g_k\|^2 - 2\beta_k g_k^T d_{k-1} + \beta_k^2\|d_{k-1}\|^2 \\
&\leq \|g_k\|^2 + 2\left|\beta_k\right|\left|g_k^T d_{k-1}\right| + \beta_k^2\|d_{k-1}\|^2 \\
&\leq \|g_k\|^2 + \frac{2E}{\alpha_k}\frac{\|g_k\|\,\|s_{k-1}\|}{(1-\sigma)\left|g_{k-1}^T d_{k-1}\right|}(\sigma)\left|g_{k-1}^T d_{k-1}\right| + \frac{E^2}{\alpha_k^2}\frac{(\sigma g_{k-1}^T d_{k-1})^2\,\|s_{k-1}\|^2}{(1-\sigma)^2\left|g_{k-1}^T d_{k-1}\right|^2} \\
&\leq \|g_k\|^2 + \frac{2E}{\alpha_k}\frac{\|g_k\|\,\|s_{k-1}\|}{(1-\sigma)}\sigma + \frac{E^2}{\alpha_k^2}\frac{\sigma^2\,\|s_{k-1}\|^2}{(1-\sigma)^2}.
\end{aligned}$$

Further calculation gives

$$\begin{aligned}\frac{\|d_k\|^2}{\|g_k\|^4} &\le \frac{\|g_k\|^2}{\|g_k\|^4} + \frac{2E}{\alpha_k}\frac{\|g_k\|\,\|s_{k-1}\|}{(1-\sigma)\|g_k\|^4}\sigma + \frac{E}{\alpha_k^2}\frac{\sigma^2\,\|s_{k-1}\|^2}{(1-\sigma)^2\|g_k\|^4}\\ &\le \frac{1}{\|g_k\|^2} + \frac{2E}{\alpha_k}\frac{\|s_{k-1}\|}{(1-\sigma)\|g_k\|^3}\sigma + \frac{E^2}{\alpha_k^2}\frac{\sigma^2\,\|s_{k-1}\|^2}{(1-\sigma)^2\|g_k\|^4}\\ &\le \frac{1}{\|g_k\|^2} + \frac{2E}{\alpha_k}\frac{\|s_{k-1}\|}{(1-\sigma)\|g_k\|^3}\sigma + \frac{E^2}{\alpha_k^2}\frac{\sigma^2\,\|s_{k-1}\|^2}{(1-\sigma)^2\|g_k\|^4}.\end{aligned}$$

If

$$\|g_k\|^m = \min\left\{\|g_k\|^2, \|g_k\|^3, \|g_k\|^4\right\}, \quad m \in \mathbb{N},$$

then it follows that

$$\frac{\|d_k\|^2}{\|g_k\|^4} \le \frac{1}{\|g_k\|^m}\left(1 + \frac{2E}{\alpha_k}\frac{\lambda}{(1-\sigma)}\sigma + \frac{E^2}{\alpha_k^2}\frac{\sigma^2\lambda^2}{(1-\sigma)^2}\right).$$

Also,

$$R = \left(1 + \frac{2E}{\alpha_k}\lambda\sigma + \frac{E^2}{\alpha_k^2}\frac{\sigma^2\lambda^2}{(1-\sigma)^2}\right)$$

initiates

$$\frac{\|d_k\|^2}{\|g_k\|^4} \le \frac{R}{\|g_k\|^m} \le R\sum_{i=1}^{k}\frac{1}{\|g_i\|^m} \quad \text{and} \quad \frac{\|g_k\|^4}{\|d_k\|^2} \ge \frac{\epsilon^m}{kR}.$$

Therefore,

$$\sum_{k=0}^{\infty}\frac{\|g_k\|^4}{\|d_k\|^2} = \infty.$$

This result contradicts (15). Therefore, $\liminf_{k\to\infty}\|g_k\| = 0$, completing the proof. □

4 Numerical Results and Discussion

To analyze the efficiency of the proposed method, we use more than 200 standard test functions presented in Table 1. These test functions are available from the CUTEst library [28] with the CUTEr/st test functions and SIF extension available on the website

`http://www.cuter.rl.ac.uk/Problems/mastsif.shtml`

The numerical results of CG_Descent 5.3 were obtained by running the code provided by Hager and Zhang [29] with memory set to 0. The numerical results of AZHS are obtained using a modified CG_Descent code with the SWP line search, employing $\sigma = 0.1$ and $\delta = 0.01$. If $\mu_k > 1$, then we conclude that $L < 1$ and $\frac{\|g_k\|^2}{|g_k^T g_{k-1}|} > 1$. Thus, it is reasonable to modify Eq.(11) as follows:

$$\beta_k^{AZHS} = \begin{cases} \frac{\|g_k\|^2 - |g_k^T g_{k-1}|}{d_{k-1}^T y_{k-1}}, & \text{if } \|g_k\|^2 > |g_k^T g_{k-1}|,\\ \frac{\|g_k\|^2 - \mu_k|g_k^T g_{k-1}|}{d_{k-1}^T y_{k-1}} - \frac{1}{\alpha_k}\mu_k\frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}, & \text{if } \|g_k\|^2 > \mu_k|g_k^T g_{k-1}|,\\ -\frac{1}{\alpha_k}\mu_k\frac{g_k^T s_{k-1}}{d_{k-1}^T y_{k-1}}, & \text{otherwise.}\end{cases}$$

Note that if $\beta_k^{AZHS} = \frac{\|g_k\|^2 - |g_k^T g_{k-1}|}{d_{k-1}^T y_{k-1}}$, then $\beta_k^{AZHS} \leq \beta_k^{HS}$, thus the proof will be similar to that presented in [7].

The host computer used was an AMD A4-7210 APU with AMD Radeon R3 Graphics, 4 GB RAM, and a 64-bit operating system. The graphs on the following results were obtained using SigmaPlot, a performance measure introduced by Dolan and Moré [30].

This performance measure compares the performance of a set of solvers S on a set of problems ρ. For n_s solvers and n_p problems in S and ρ, respectively, the measure $t_{p,s}$ is the computation time (e.g., the number of iterations or CPU time) required for solver s to solve problem p.

To establish a baseline for comparison, the performance of solver s on problem p is scaled relative to the best performance of any solver in S on that problem, yielding the ratio

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

A parameter $r_M \geq r_{p,s}$ for all p, s is selected such that $r_{p,s} = r_M$ if and only if solver s cannot solve problem p. To obtain an overall assessment of the performance of each solver, we define the measure

$$P_s(t) = \frac{1}{n_p} \text{size}\{p \in \rho : r_{p,s} \leq t\}.$$

$P_s(t)$ is the probability for solver $s \in S$ that the performance ratio $r_{p,s}$ will be within a factor $t \in \mathbb{R}$ of the best possible ratio. If we denote the cumulative distribution function of the performance ratio as p_s, then the performance measure $p_s : \mathbb{R} \to [0, 1]$ for a given solver is non-decreasing and piecewise continuous from the right. The value of $p_s(1)$ is the probability that the solver will achieve the best performance among all solvers. In general, a solver with higher values of $P_s(t)$, which will lie closer to the upper right corner of the figure, is preferable.

The numerical results are shown in Figures 1, 2, 3 and 4. Figure 1 depicts the number of iterations, showing that the new modification significantly outperforms CG_Descent 5.3. Figure 2 illustrates that the new modification, AZHS, outperforms CG_Descent 5.3 in the number of function evaluations. Figures 3 and 4 show the performance based on the number of gradient evaluations and CPU time, respectively. It is observed that AZHS outperforms CG_Descent 5.3 in CPU time and is significantly competitive with CG_Descent 5.3 in the number of function evaluations and gradient evaluations as the latter used an approximate Wolfe line search with $\sigma = 0.9$ and $\delta = 0.1$. Thus, we can conclude that β_k^{AZHS} outperforms CG_Descent 5.3 in all figures.

5 Application to Heat Conduction Problem [32]

Suppose a rectangular flat plate with dimensions of 5×4 units generates heat [33]. Suppose the thermal conductivity k is fixed, and the heat production per unit area f is a nonlinear function of the temperature M. Our objective is to define the temperature of the slab such that the temperature outside the perimeter of the slab is zero. Poisson's equation classifies the temperature distribution within this region as follows:

$$k\left[\frac{\partial^2 M}{\partial x^2} + \frac{\partial^2 M}{\partial y^2}\right] + f(M) = 0.$$

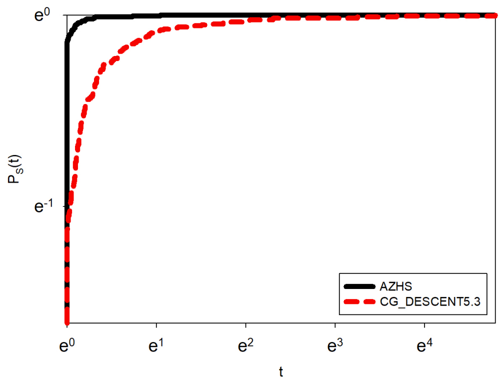


Figure 1: Performance measure based on the number of iterations.

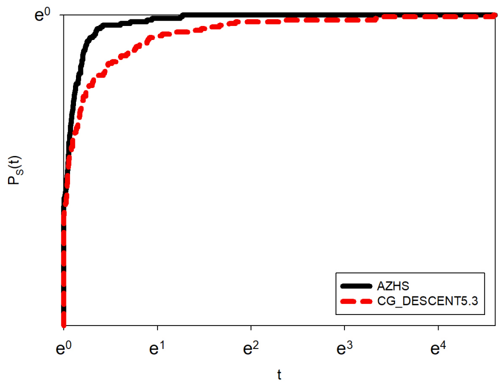


Figure 2: Performance measure based on the function evaluation.

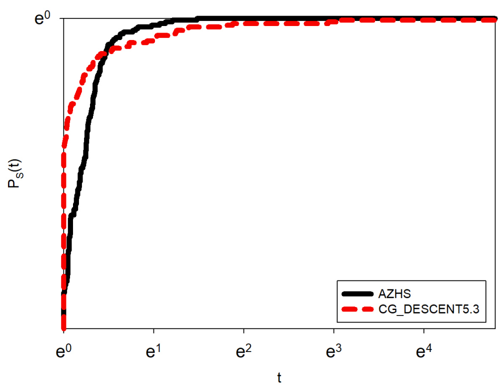


Figure 3: Performance measure based on the gradient evaluation.

Function	Dim	Function	Dim	Function	Dim
AKIVA	2	FBRAIN2LS	4	OSCIPATH	10
ALLINITU	4	FLETCBV2	5000	PALMER1C	8
ARGLINB	200	FLETCHCR	1000	PALMER1D	7
ARGLINC	200	FMINSRF2	5625	PALMER2C	8
ARWHEAD	5000	FMINSURF	5625	PALMER3C	8
BARD	3	GENHUMPS	5000	PALMER4C	8
BDEXP	5000	GROWTHLS	3	PALMER5C	6
BDQRTIC	5000	GULF	3	PALMER6C	8
BEALE	2	HAHN1LS	7	PALMER7C	8
BIGGS3	6	HAIRY	2	PALMER8C	8
BIGGS5	6	HATFLDD	3	PARKCH	15
BIGGS6	6	HATFLDE	3	PENALTY1	1000
BIGGSB1	5000	HATFLDFL	3	PENALTY2	200
BOX2	3	HATFLDFLS	3	PENALTY3	200
BOX3	3	HEART6LS	6	PENALTY3	200
BOX	10000	HEART8LS	8	POWELLBSLS	2
BRKMCC	2	HELIX	3	POWELLSG	5000
BROYDNBDLS	10	HIELOW	3	POWER	10000
BROWNAL	200	HILBERTA	2	POWERSUM	4
BROWNBS	2	HILBERTB	10	PRICE3	2
BROWNDEN	4	HIMMELBB	2	PRICE4	2
BROYDN7D	5000	HIMMELBF	4	QING	100
BRYBND	5000	HIMMELBG	2	QUARTC	5000
CAMEL6	2	HIMMELBH	2	RAT43LS	4
CHNROSNB	50	HUMPS	2	RECIPELS	3
CLIFF	2	HYDCAR6LS	29	ROSENBR	2
COSINE	10000	INDEF	5000	ROSENBRTU	2
CUBE	2	INDEFM	100000	S308	2
CURLY10	10000	INTEQNELS	12	SCHMVETT	5000
CURLY20	10000	JENSMP	2	SENSORS	100
CURLY30	10000	JIMACK	3549	SINEVAL	2
DENSCHNA	2	JUDGE	2	SINQUAD	5000
DENSCHNB	2	KOWOSB	4	SISSER	2
DENSCHNC	2	KSSLS	1000	SNAIL	2
DENSCHND	3	LANCZOS1LS	6	SPMSRTLS	4999
DENSCHNE	3	LANCZOS2LS	6	SROSENBR	5000
DENSCHNF	2	LANCZOS3LS	6	SSCOSINE	5000
DIXMAANA	3000	LIARWHD	5000	SSI	3
DIXMAANB	3000	LOGHAIRY	2	STREG	4
DIXMAANC	3000	LSC1LS	3	STRATEC	10
DIXMAAND	3000	LSC2LS	3	STRTCHDV	10
DIXMAANE	3000	LUKSAN11LS	100	TESTQUAD	5000
DIXMAANF	3000	LUKSAN12LS	98	THURBERLS	7
DIXMAANG	3000	LUKSAN13LS	98	TOINTGOR	50
DIXMAANH	3000	LUKSAN14LS	98	TOINTGSS	5000
DIXMAANI	3000	LUKSAN15LS	100	TOINTPSP	50
DIXMAANJ	3000	LUKSAN16LS	100	TOINTQOR	50
DIXMAANK	3000	MANCINO	100	TQUARTIC	5000
DIXMAANL	3000	MARATOSB	2	TRIDIA	5000
DIXMAANP	3000	MEXHAT	2	TRIGON1	10
DIXON3DQ	10000	MEYER3	3	TRIGON2	10
DJTL	2	MGH09LS	4	VANDANMSLS	22
DMN15332LS	66	MGH10LS	3	VARDIM	200
DQDRTIC	5000	MGH10SLS	3	VAREIGVL	50
ECKERLE4LS	3	MGH17LS	5	VESUVIALS	8
EDENSCH	2000	MISRA1BLS	2	VESUVIOULS	8
EGGCRATE	2	MISRA1CLS	2	VIBRBEAM	8
EG2	1000	MISRA1DLS	2	WAYSEA1	2
EIGENALS	2550	MODBEALE	20000	WAYSEA2	2
EIGENBLS	2550	MOREBV	5000	WOODS	4000
EIGENCLS	2652	MSQRTALS	1024	YATP1CLS	123200
ELATVIDU	2	MSQRTBLS	1024	YATP2CLS	123200
ENGVAL1	5000	NCB20	5010	YFITU	3
ENGVAL2	3	NELSONLS	3	ZANGWIL2	2
ENSOLS	9	NONCVXU2	5000		
EXPFIT	2	NONDIA	5000		

Table 1: Test functions.

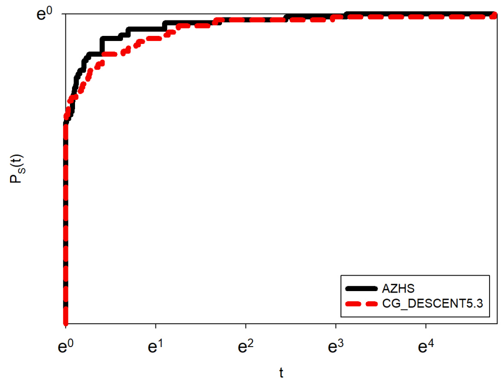


Figure 4: Performance measure based on the CPU time.

If $k = 2$ and $f(M) = 20 - \frac{3}{2}M + \frac{1}{20}M^2$, there are 12 mesh points in total. Symmetry reduces the problem to only four distinct temperatures.

$$2(M_2 + M_3 - 4M_1) = -20 + \frac{3}{2}M_1 - \frac{1}{20}M_1^2,$$

$$2(M_3 + M_1 + M_4 - 4M_3) = -20 + \frac{3}{2}M_3 - \frac{1}{20}M_3^2,$$

$$2(M_1 + M_4 + 4M_2) = -20 + \frac{3}{2}M_2 - \frac{1}{20}M_2^2,$$

$$2(2M_3 + M_2 - 3M_4) = -20 + \frac{3}{2}M_4 - \frac{1}{20}M_4^2.$$

These equations, expressed in powers of M_1, are as follows:

$$\left(M_1^2 - 190M_1\right) + 40\left(M_2 + M_3 + 10\right) = 0,$$

$$M_1 + \frac{M_3^2 - 150M_3 + 400}{40} + M_4 = 0,$$

$$2M_1 + \frac{M_2^2 - 190M_2 + 400}{40} + M_4 = 0,$$

$$\left(M_4^2 - 150M_4\right) + 40M_2 + 80M_3 + 400 = 0.$$

The objective function f is constructed by summing the squares of the functions connected with each nonlinear equation as follows:

$$f(M_1, M_2, M_3, M_4, H_1, H_2, H_3, H_4, H_5, H_6) = Q_1 + Q_2 + Q_3 + Q_4,$$

where

$$Q_1 = Q_5^2, \quad Q_2 = Q_6^2, \quad Q_3 = Q_7^2, \quad Q_4 = Q_8^2,$$

$$Q_5 = \frac{1}{20}\left[M_1^2 + H_1M_1 + H_2\left(M_2 + M_3 + H_3\right)\right],$$

$$Q_6 = 2\left[M_1 + \frac{M_3^2 + H_4M_3}{H_2} + H_5 + M_4\right],$$

$$Q_7 = 2\left[H_6M_1 + \frac{M_2^2 + H_1M_2}{H_2} + H_5 + M_4\right],$$

$$Q_8 = \frac{1}{20}\left[M_4^2 + H_4M_4 + H_2M_2 + H_2H_6M_3 + H_2H_5\right].$$

If

$$H_1 = -190, \quad H_2 = 40, \quad H_3 = 10, \quad H_4 = -150, \quad H_5 = 10, \quad H_6 = 2,$$

let

$$M_1 = x_1, \quad M_2 = x_2, \quad M_3 = x_3, \quad M_4 = x_4.$$

Then, we obtain the following function:

$$\begin{aligned} f(x_1, x_2, x_3, x_4) = &\left(2(x_2 + x_3 - 4x_1) + 20 - 1.5x_1 + \frac{x_1^2}{20}\right)^2 \\ &+ \left(2(x_1 - 3x_3 + x_4) + 20 - 1.5x_3 + \frac{x_3^2}{20}\right)^2 \\ + \left(2(2x_1 + x_4 - 4x_2) + 20 - 1.5x_2 + \frac{x_2^2}{20}\right)^2 &+ \left(2(x_2 + 2x_3 - 3x_4) + 20 - 1.5x_4 + \frac{x_4^2}{20}\right)^2. \end{aligned}$$

We say that $f(x_1, x_2, x_3, x_4)$ is the Heat Conduction Problem function. By using Algorithm 1, we can find the values of x_1, x_2, x_3, x_4 as follows:

$$x_1 = 4.8521, \quad x_2 = 6.0545, \quad x_3 = 6.4042, \quad x_4 = 8.1383.$$

The function value is 1.9631×10^{-7}.

6 Application to Image Restoration

Restoring damaged images is one of the most important applications of the CG method. In this study, we applied Gaussian noise with a standard deviation of 25% to the original images in Table 3. After that, we used Algorithm 1 to restore these images. To express the efficiency of the proposed method, we made a comparison between Algorithm 1, CG-Descent5.3, and DL+ in terms of the number of iterations, CPU time, and root-mean-square error (RMSE).

We utilized the RMSE between the restored image and the original true image to calculate the quality of the restored image:

$$\text{RMSE} = \frac{\|\nu - \nu_k\|_2}{\|\nu\|}.$$

The restored image is denoted by ν_k and the true image by ν. The RMSE determines the quality of the restored image, in which lower values correspond to higher quality. The results in Table 2 show that the new search direction outperforms CG-Descent5.3 and DL+ in terms of the number of iterations, CPU time, and the RMSE value. The criteria for stopping is

$$\frac{\|x_{k+1} - x_k\|_2}{\|x_k\|_2} < \varepsilon.$$

In this context, $\epsilon = 10^{-3}$. Note that if $\epsilon = 10^{-4}$ or $\epsilon = 10^{-6}$, then the RMSE remains fixed, meaning that a fixed RMSE can have a variation in the number of iterations.

Table 3 below shows the outcomes of restoring the destroyed images using Algorithm 1, indicating that it can be regarded as an efficient approach.

Image	Algorithm	Number of Iteration	CPU Time (s)	RMSE
Mandi 128 pixels	DL+	127	1.724e+000	0.1003
	AZHS	126	1.663e+000	0.1002
	CG-Descent5.3	134	1.825e-001	0.1004
Coins 128 pixels	DL+	135	1.542e+000	0.0832
	AZHS	130	1.491e+000	0.0824
	CG-Descent5.3	133	1.491e+000	0.0831
Mandi 256 pixels	DL+	120	1.856e+001	0.0519
	AZHS	111	1.545e+001	0.0510
	CG-Descent5.3	119	1.656e+001	0.0991
Coins 256 pixels	DL+	134	1.447e+001	0.0506
	AZHS	120	1.164e+001	0.0501
	CG-Descent5.3	130	1.564e+001	0.0508
Mandi 512 pixels	DL+	114	7.981e+001	0.0371
	AZHS	105	6.755e+001	0.0360
	CG-Descent5.3	116	7.314e+001	0.0472
Kids 512 pixels	DL+	57	6.955e+001	0.0377
	AZHS	56	5.325e+001	0.0384
	CG-Descent5.3	55	5.634e+001	0.0395
Coins 512 pixels	DL+	129	7.323e+001	0.0326
	AZHS	128	5.248e+001	0.0324
	CG-Descent5.3	127	6.323e+001	0.0503
Coins 1024 pixels	DL+	128	3.441e+002	0.0326
	AZHS	110	2.549e+002	0.0172
	CG-Descent5.3	124	2.897e+002	0.0289

Table 2: Numerical outcomes from the images with Gaussian noise with a 25% standard deviation added to the original images using the Dai-Liao CG method, AZHS, as well as CG-Descent5.3.

7 Conclusion

In this study, we investigate a modified Hestenes–Stiefel (HS) conjugate gradient (CG) method based on the Dai–Liao conjugacy parameter, with the restart property depending on L. The newly modified CG method inherits global convergence properties and a sufficient descent condition through the SWP line search. Moreover, the numerical results are efficient and competitive with CG Descent5.3. Applications to solving the Heat Conduction Problem and image restoration are presented. In future studies, we will focus on the Lipschitz constant because it plays an essential role in the efficiency and robustness of the CG method.

Acknowledgment

We would like to thank Prof. William W. Hager for publishing his code in the implementation of the CG method. Additionally, we extend our gratitude to the editor and the reviewers for their valuable feedback and suggestions.

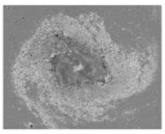
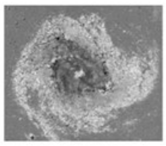
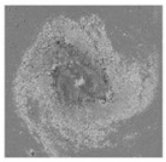
Image	Original Image	Image with Gaussian Noise	Restored Image
Mandi (128 pixels)			
Mandi (256 pixels)			
Coins (256 pixels)			
Kids (512 pixels)			
M.83 (1024 pixels)			

Table 3: Restoration of the destroyed images of Mandi, Coins, Kids, as well as M.83 by reducing z via Algorithm 1.

References

[1] P. Wolfe. Convergence conditions for ascent methods. II: Some corrections. *SIAM Rev.* **13** (2) (1971) 185–188.

[2] S. Yao, L. Ning, H. Tu and J. Xu. A one-parameter class of three-term conjugate gradient methods with an adaptive parameter choice. *Optim. Methods Softw.* **35** (6) (2020) 1051–1064.

[3] M. R. Hestenes and E. Stiefel. Methods of conjugate gradients for solving linear systems. *J. Res. Nat. Bur. Stand.* **49** (1952) 409–436.

[4] R. Fletcher and C. M. Reeves. Function minimization by conjugate gradients. *Comput. J.* **7** (1964) 149–154.

[5] P. Elijah and G. Ribiere. Note sur la convergence de méthodes de directions conjuguées. *ESAIM: Math. Model. Numer. Anal.* **3** R1 (1969) 35–43.

[6] M. J. D. Powell. Nonconvex minimization calculations and the conjugate gradient method. In: *Numerical Analysis (Dundee, 1983)*, Lecture Notes in Math. **1066**, Springer, Berlin, 1984, 122–141.

[7] J. C. Gilbert and J. Nocedal. Global convergence properties of conjugate gradient methods for optimization. *SIAM J. Optim.* **2** (1992) 21–42.

[8] M. Al-Baali. Descent property and global convergence of the Fletcher–Reeves method with inexact line search. *IMA J. Numer. Anal.* **5** (1985) 121–124.

[9] Y. H. Dai and L. Z. Liao. New conjugacy conditions and related nonlinear conjugate gradient methods. *Appl. Math. Optim.* **43** (1) (2014) 87–101.

[10] S. Babaie-Kafaki and R. Ghanbari. The Dai–Liao nonlinear conjugate gradient method with optimal parameter choices. *Eur. J. Oper. Res.* **234** (3) (2014) 625–630.

[11] S. Babaie-Kafaki and R. Ghanbari. A descent family of Dai–Liao conjugate gradient methods. *Optim. Methods Softw.* **29** (3) (2014) 583–591.

[12] A. Perry. A modified conjugate gradient algorithm. *Oper. Res.* **26** (1976) 1073–1078.

[13] N. Andrei. A Dai–Liao conjugate gradient algorithm with clustering of eigenvalues. *Numer. Algorithms* **77** (4) (2018) 1273–1282.

[14] W. W. Hager and H. Zhang. A new conjugate gradient method with guaranteed descent and an efficient line search. *SIAM J. Optim.* **16** (2005) 170–192.

[15] W. W. Hager and H. Zhang. The limited memory conjugate gradient method. *SIAM J. Optim.* **23** (2013) 2150–2168.

[16] K. Zhang, H. Liu and Z. Liu. A new Dai–Liao conjugate gradient method with optimal parameter choice. *Numer. Funct. Anal. Optim.* **40** (2) (2019) 194–215.

[17] M. Al-Baali, Y. Narushima and H. Yabe. A family of three-term conjugate gradient methods with sufficient descent property for unconstrained optimization. *Comput. Optim. Appl.* **60** (2015) 89–110.

[18] Z. F. Dai, H. Y. Zhu, and X. Zhang. Dynamic spillover effects and portfolio strategies between crude oil, gold and Chinese stock markets related to new energy vehicle. *Energy Economics* **109** (2022) 105959.

[19] Z. F. Dai, H. Zhuo, J. Kang, and F. Wen. The skewness of oil price returns and equity premium predictability. *Energy Economics* **94** (2021) 105069.

[20] Z. F. Dai and H. Zhu, Forecasting stock market returns by combining sum-of-the-parts and ensemble empirical mode decomposition. *Applied Economics* **52** (2020) 2309–2323.

[21] Z. Dai and F. Wen, Another improved Wei–Yao–Liu nonlinear conjugate gradient method with sufficient descent property. *Applied Mathematics and Computation* **218** (14) (2012) 7421–7430.

[22] Z. Salleh, A. Almarashi and A. Alhawarat. Two efficient modifications of AZPRP conjugate gradient method with sufficient descent property. *Journal of Inequalities and Applications* (1) (2022) 1–21.

[23] A. Alhawarat, Z. Salleh, M. Mamat and M. Rivaie. An efficient modified Polak–Ribière–Polyak conjugate gradient method with global convergence properties. *Optim. Methods Softw.* **32** (6) (2017) 1299–1312.

[24] S. Yao, L. Ning, H. Tu and J. Xu. A one-parameter class of three-term conjugate gradient methods with an adaptive parameter choice. *Optimization Methods and Software* **35** (6) (2020) 1051–1064.

[25] G. Zoutendijk. Nonlinear programming, computational methods, Integer Nonlinear Program. **143** (1970) 37–86.

[26] Y. H. Dai, J. Y. Han, G. H. Liu, D. F. Sun, H. X. Yin and Y. Yuan. Convergence properties of nonlinear conjugate gradient methods. *SIAM J. Optim.* **10** (2) (1999) 348–358.

[27] G. Yuan and Z. Wei. Non-Monotone Backtracking Inexact BFGS Method for Regression Analysis. *Communications in Statistics-Theory and Methods* **42** (2) (2013) 214–238.

[28] I. Bongartz, A.R. Conn, N. Gould and P.L. Toint, CUTE: Constrained and unconstrained testing environment, *ACM Trans. Math. Softw.* **21** (1995) 123–160.

[29] `http://users.clas.ufl.edu/hager/papers/Software/` (2024).

[30] E. D. Dolan and J. J. Moré, Benchmarking optimization software with performance profiles, *Math. Program.* **91** (2002) 201–213.

[31] S. K. Mishra and B. Ram. Introduction to unconstrained optimization with R. Springer Nature, 2019.

[32] A. J. Surkan and C. L. Solution of reaction and heat flow problems by nonlinear estimation. The Canadian Journal of Chemical Engineering **46** (4) (1968) 229–232.