Iranian Journal of Numerical Analysis and Optimization
Vol. 16, No. 1, 2026, pp 264–282
https://doi.org/10.22067/ijnao.2025.95643.1739
https://ijnao.um.ac.ir/

Research Article

An efficient hybrid three-term conjugate gradient method with practical applications

Y.E. Hemici*, S. Khelladi and D. Benterki

Abstract

The conjugate gradient method is widely used for large-scale unconstrained optimization due to its efficiency in iteration number and computing time. In this paper, we present a new parameter β_k by hybridizing the Liu–Storey parameter β_k^{LS} and its modification β_k^{DLS}. We also use a hybrid three-term technique ensuring sufficient descent based on strong Wolfe conditions. We prove that the new direction possesses the descent property and that the corresponding algorithm is globally convergent. Numerical experiments on test problems and some applications, such as image restoration and sparse signal recovery, further show that our approach is efficient, consistently outperforming other approaches in terms of convergence speed and solution quality.

*Corresponding author

Received 26 September 2025; revised 24 November 2025; accepted 5 December 2025

Youcef Elhamam Hemici
Laboratory of Fundamental and Numerical Mathematics, Department of Mathematics, Faculty of Sciences, Setif1 University-Ferhat Abbas, 19000, Algeria. e-mail: youcefelhamam.hemici@univ-setif.dz

Samia Khelladi
Laboratory of Fundamental and Numerical Mathematics, Department of Mathematics, Faculty of Sciences, Setif1 University-Ferhat Abbas, 19000, Algeria. e-mail: samia.boukaroura@univ-setif.dz

Djamel Benterki
Laboratory of Fundamental and Numerical Mathematics, Department of Mathematics, Faculty of Sciences, Setif1 University-Ferhat Abbas, 19000, Algeria. e-mail: djbenterki@univ-setif.dz

How to cite this article
Hemici, Y.E., Khelladi, S. and Benterki, D., An efficient hybrid three-term conjugate gradient method with practical applications. *Iran. J. Numer. Anal. Optim.*, 2026; 16(1): 264-282. https://doi.org/10.22067/ijnao.2025.95643.1739

AMS subject classifications (2020): 90C26, 90C30, 65K05

Keywords: Unconstrained optimization, Conjugate gradient method, Descent direction, Image restoration, Signal recovery.

1 Introduction

Optimization techniques are widely used in science, engineering, economics, management, industry, and other areas. The most common optimization algorithms are the descent methods, namely gradient, Newton's, and conjugate gradient methods.

Let $f : \mathbb{R}^n \to \mathbb{R}$ be a continuously differentiable function. Our goal is to solve the unconstrained optimization problem $\{\min f(x),\ x \in \mathbb{R}^n\}$. The principle of the conjugate gradient method is to construct a sequence

$$x_1 \in \mathbb{R}^n,\ x_{k+1} = x_k + \alpha_k d_k, \qquad \text{for } k \geq 1, \tag{1}$$

where x_1 is an initial point and x_k denotes the current iterate. The step size $\alpha_k > 0$ is computed using an exact or inexact line search, while the search direction d_k is defined as

$$d_1 = -g_1,\ d_{k+1} = -g_{k+1} + \beta_k d_k, \qquad \text{for } k \geq 1,$$

where $g_k = \nabla f(x_k)$ is the gradient of f at x_k and β_k is a scalar conjugacy coefficient.

The conjugate gradient method was first introduced by Hestenes and Stiefel (HS) in 1952 [19]. Later, different formulas of the parameter β_k were introduced, leading to various conjugate gradient directions d_k. Some of the most well-known choices of β_k include those proposed by Fletcher and Reeves (FR) [16], Polak–Ribière–Polyak (PRP) [29, 28], conjugate descent (CD) [6], Liu–Storey (LS) [22], and Dai–Yuan (DY) [15, 14]. For more details, see [18].

These methods are identical for a strongly convex quadratic function and the exact line search, but they behave differently for general nonlinear functions and the inexact line search. The FR, DY, and CD conjugate gradient methods are known for their strong convergence but poor practical behavior, as they are prone to taking short steps, which can lead to jamming. On the other hand, the LS, HS, and PRP conjugate gradient methods are well known for their good practical behavior due to their ability to switch to the steepest descent direction in order to avoid short steps that result in jamming [20], but they may not converge in general.

Based on the latter, many researchers have proposed modifications of the original schemes. Dai and Wen [8] proposed two modified conjugate gradient methods, denoted DPRP and DHS, given by

$$\beta_k^{DPRP} = \frac{\|g_{k+1}\|^2 - \frac{\|g_{k+1}\|}{\|g_k\|}\left|g_{k+1}^T g_k\right|}{\mu\left|g_{k+1}^T d_k\right| + \|g_{k+1}\|^2}, \qquad \mu > 1, \tag{2}$$

$$\beta_k^{DHS} = \frac{\|g_{k+1}\|^2 - \frac{\|g_{k+1}\|}{\|g_k\|}\left|g_{k+1}^T g_k\right|}{\mu\left|g_{k+1}^T d_k\right| + d_k^T y_k}, \qquad \mu > 1. \tag{3}$$

Additionally, inspired by (2) and (3), Zhang and Zheng proposed in [36] a modified LS method denoted as DLS, given by

$$\beta_k^{DLS} = \frac{\|g_{k+1}\|^2 - \frac{\|g_{k+1}\|}{\|g_k\|}\left|g_{k+1}^T g_k\right|}{\mu\left|g_{k+1}^T d_k\right| - d_k^T y_k}, \qquad \mu > 1. \tag{4}$$

In recent years, many researchers have introduced new hybrid conjugate gradient methods. The main concept behind hybridization is to combine classical conjugate gradient strategies in order to take advantage of their positive properties while reducing undesirable behaviors such as the jamming phenomenon. Depending on how the combination is formed, hybrid schemes can be either convex or nonconvex. Several examples of such approaches have been reported in the literature, including those proposed by Dalladji, Belloufi, and Sellami [9], Mtagulwa and Kaelo [26], Djordjević [11, 10, 12], Ben Hanachi, Sellami, and Belloufi [3], Yang, Luo, and Dai [35], Rivaie, Mustafa, and Abdelrhaman [30], Sellami and Chaib [31, 32], Sulaiman et al. [34], and Ouaoua, Khelladi, and Benterki [27].

In 2022, Jiang et al. [21], based on the hybrid conjugate gradient method and the convex combination technique, proposed a new family of hybrid three-term conjugate gradient methods:

$$d_{k+1} = \begin{cases} -g_{k+1} & \text{if} \quad k = 0, \\ -g_{k+1} + (1-\lambda_k)\beta_k^* d_k + \lambda_k \theta_k g_k & \text{if} \quad k > 0, \end{cases} \tag{5}$$

where $\beta_k^* = \max\{0, \min\{\beta_k, \beta_k^{DY}\}\}$, β_k is any conjugate parameter, $\lambda_k = \frac{\left|g_{k+1}^T d_k\right|}{\|g_{k+1}\|\|d_k\|}$ with $\lambda_k \in [0,1]$, and $\theta_k = -\eta \frac{g_{k+1}^T g_k}{\|g_k\|^2}$, $0 < \eta < 1$.

The objective of this work is to propose a new hybridization between LS and DLS using the formula (5) to calculate a new descent search direction.

The paper is structured as follows. In Section 2, we introduce the proposed hybrid conjugate gradient parameter together with the corresponding algorithm. Section 3 provides a thorough analysis of the descent property of the obtained direction and establishes the global convergence of the algorithm under the strong Wolfe line search conditions. In Section 4, we assess the efficiency of the proposed method through numerical experiments on standard benchmark functions, as well as its application to image restoration and signal recovery. Finally, Section 5 presents the conclusion of this paper.

2 New hybrid conjugate gradient algorithm

Inspired by the works of Zhang and Zheng [36] and Jiang et al. [21], we propose a new hybrid parameter between LS and DLS (4), denoted by β_k^{MDLS}, defined by

$$\beta_k^{MDLS} = \begin{cases} \beta_k^{LS} = -\frac{g_{k+1}^T y_k}{g_k^T d_k} & \text{if} \quad \|g_{k+1}\|^2 \geq \left|g_{k+1}^T g_k\right|, \\ \beta_k^{DLS} = \frac{\|g_{k+1}\|^2 - \frac{\|g_{k+1}\|}{\|g_k\|}\left|g_{k+1}^T g_k\right|}{\mu\left|g_{k+1}^T d_k\right| - d_k^T y_k} & \text{otherwise,} \end{cases} \tag{6}$$

where $\mu > 1$.

The search direction is given as follows:

$$d_{k+1} = \begin{cases} -g_{k+1} & \text{if } k = 0 \text{or } \left|g_{k+1}^T g_k\right| \geq 0.5\|g_{k+1}\|^2, \\ -g_{k+1} + (1-\lambda_k)\beta_k^{MDLS} d_k + \lambda_k \theta_k g_k & \text{if } \ k > 0, \end{cases} \tag{7}$$

where $\lambda_k = \frac{\left|g_{k+1}^T d_k\right|}{\|g_{k+1}^T\|\|d_k\|}$ and $\theta_k = -\eta\frac{g_{k+1}^T g_k}{\|g_k\|^2}$. Note that $\lambda_k \in [0,1]$ and $0 < \eta < 1$.

We evaluate the behavior of our method, we use the strong Wolfe line search,that is, we find the stepsize α_k using the following conditions:

$$f(x_k + \alpha_k d_k) \leq f(x_k) + \delta \alpha_k g_k^T d_k, \tag{8}$$

$$\left|g_{k+1}^T d_k\right| \leq -\sigma g_k^T d_k, \tag{9}$$

where $0 < \delta < \sigma < 1$.

The corresponding algorithm is given bellow.

3 Convergence analysis

We introduce the following theorem to establish the sufficient descent condition.

Algorithm 1 MDLS

Step 0: Start by the initial point x_1 and a tolerance $\varepsilon > 0$.
Step 1: Set $k = 1$ and compute $d_1 = -g_1$.
Step 2: If $\|g_k\| \leq \varepsilon$, then **Stop**; else go to **Step 3**.
Step 3: Compute the stepsize $\alpha_k \in]0, 1]$, which satisfies strong Wolfe conditions (8), (9).
Step 4: Compute $x_{k+1} = x_k + \alpha_k d_k$.
Step 5: Compute $g_{k+1} = \nabla f(x_{k+1})$, $y_k = g_{k+1} - g_k$.
Step 6: Take $\beta_k = \beta_k^{MDLS}$ (6).
Step 7: Compute d_{k+1} using (7).
Step 8: Set $k = k + 1$ and return to **Step 2.**

Theorem 1. Let g_k and d_k denote the sequences produced by the $MDLS$ algorithm. Then, the corresponding search direction satisfies the sufficient descent condition given by

$$g_{k+1}^T d_{k+1} \leq -c\|g_{k+1}\|^2, \quad \text{for all } k \geq 0, \tag{10}$$

where c is a positive constant.

Proof. Multiplying (7) by g_{k+1}^T from the left gives

$$g_{k+1}^T d_{k+1}^{MDLS} = -\|g_{k+1}\|^2 + (1 - \lambda_k)\beta_k^{MDLS}\, g_{k+1}^T d_k + \lambda_k \theta_k\, g_{k+1}^T g_k. \tag{11}$$

We now analyze two cases.

Case 1. If $\|g_{k+1}\|^2 \geq \left|g_{k+1}^T g_k\right|$, then $\beta_k^{MDLS} = \beta_k^{LS}$ with

$$\beta_k^{LS} = \frac{g_{k+1}^T y_k}{-g_k^T d_k}, \qquad \theta_k = -\eta \frac{g_{k+1}^T g_k}{\|g_k\|^2}.$$

We have

$$\frac{(g_{k+1}^T g_k)^2}{\|g_k\|^2} = \|g_{k+1}\|^2 \cos^2 \omega. \tag{12}$$

Substituting these into (11) and using (12), we get

$$\begin{aligned} g_{k+1}^T d_{k+1}^{MDLS} &= -\|g_{k+1}\|^2 + (1 - \lambda_k)\frac{g_{k+1}^T y_k}{-g_k^T d_k} g_{k+1}^T d_k - \eta \lambda_k \frac{(g_{k+1}^T g_k)^2}{\|g_k\|^2} \\ &= -(1 + \eta \lambda_k \cos^2 \omega)\|g_{k+1}\|^2 + (1 - \lambda_k)\frac{g_{k+1}^T y_k}{-g_k^T d_k} g_{k+1}^T d_k, \end{aligned} \tag{13}$$

where ω is the angle between g_k and g_{k+1}.

Now, using the second condition of strong Wolfe (9), we get

$$\frac{g_{k+1}^T y_k}{-g_k^T d_k} g_{k+1}^T d_k \leq \sigma g_{k+1}^T y_k.$$

Using this in (13) gives

$$\begin{aligned} g_{k+1}^T d_{k+1}^{MDLS} &\leq -(1 + \eta\lambda_k \cos^2 \omega)\|g_{k+1}\|^2 + (1 - \lambda_k)\sigma g_{k+1}^T y_k \\ &= -(1 + \eta\lambda_k \cos^2 \omega)\|g_{k+1}\|^2 + (1 - \lambda_k)\sigma \|g_{k+1}\|^2 \\ &\quad -(1 - \lambda_k)\sigma g_{k+1}^T g_k \\ &\leq -(1 + \eta\lambda_k \cos^2 \omega)\|g_{k+1}\|^2 + 2(1 - \lambda_k)\sigma \|g_{k+1}\|^2 \\ &= -(1 - 2\sigma + (2\sigma + \cos^2 \omega)\lambda_k)\|g_{k+1}\|^2. \end{aligned}$$

Thus

$$g_{k+1}^T d_{k+1}^{MDLS} \leq -c\|g_{k+1}\|^2,$$

where

$$c = 1 - 2\sigma + (2\sigma + \cos^2 \omega)\lambda_k.$$

Since $0 < \sigma < \frac{1}{2}$ implies $1 - 2\sigma > 0$, and $\lambda_k \geq 0$, $\cos^2 \omega \geq 0$, we conclude $c > 0$.

Case 2. If $\|g_{k+1}\|^2 < \left|g_{k+1}^T g_k\right|$, then $\beta_k^{MDLS} = \beta_k^{DLS}$. By Stanimirović [33], we know that

$$\beta_k^{DLS} \leq -\frac{\|g_{k+1}\|^2}{g_k^T d_k}.$$

Proceeding as in Case 1, we obtain

$$\begin{aligned} g_{k+1}^T d_{k+1}^{MDLS} &\leq -(1 + \eta\lambda_k \cos^2 \omega)\|g_{k+1}\|^2 + (1 - \lambda_k)\sigma \|g_{k+1}\|^2 \\ &= -(1 - \sigma + (\sigma + \cos^2 \omega)\lambda_k)\|g_{k+1}\|^2. \end{aligned}$$

Hence

$$g_{k+1}^T d_{k+1}^{MDLS} \leq -c\|g_{k+1}\|^2,$$

where

$$c = 1 - \sigma + (\sigma + \cos^2 \omega)\lambda_k > 0.$$

In all cases, we have shown that

$$g_{k+1}^T d_{k+1}^{MDLS} \leq -c\|g_{k+1}\|^2, \qquad c > 0,$$

which proves the sufficient descent property (10). □

To prove the global convergence of the algorithm, we make the following assumptions:

- *(i)* The level set $S = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$ is bounded; that is, there exists a constant $B > 0$ such that
$$\|x\| \leq B, \quad \text{for all } x \in S.$$
- *(ii)* In a neighborhood $\mathcal{N}$ of S, the gradient $\nabla f(x)$ is Lipschitz continuous; that is, there exists a constant $L > 0$ such that
$$\|\nabla f(x) - \nabla f(y)\| \leq L\|x - y\|, \quad \text{for all } x, y \in \mathcal{N}. \tag{14}$$

These assumptions imply the existence of a constant $\kappa \geq 0$, where
$$\|\nabla f(x)\| \leq \kappa, \qquad \text{for all } x \in S. \tag{15}$$

Lemma 1. [37] Assume that conditions (i) and (ii) are satisfied. Consider iterate (1), where d_k is the descent direction in (7), and α_k is chosen according to the strong Wolfe line search conditions (8) and (9). Then, the Zoutendijk condition
$$\sum_{k\geq 0} \frac{(g_k^T d_k)^2}{\|d_k\|^2} < \infty, \tag{16}$$
is satisfied.

Lemma 2. [12] Assume that assumptions (i) and (ii) hold and that $\alpha_k > 0$ is computed by the strong Wolfe line search (8) and (9). Then there exists $\alpha_* > 0$ such that
$$\alpha_k \geq \alpha_*, \qquad \text{for all } k \geq 1.$$

Theorem 2. Suppose that assumptions (i), (ii) and (10) hold. Then the $MDLS$ algorithm converges and satisfies
$$\lim_{k\to\infty} \inf \|g_k\| = 0. \tag{17}$$

Proof. Assume that $g_k \neq 0$, for all k. We need to prove (17).

Suppose that (17) does not hold. Then there exists a constant $r > 0$ such that
$$\|g_k\| \geq r, \qquad \text{for all } k. \tag{18}$$

Let D denote the diameter of the set S.

We have

$$|\beta_k^{DLS}| \leq -\frac{\|g_{k+1}\|^2}{g_k^T d_k} \leq \frac{\kappa^2}{cr^2},$$

$$|\beta_k^{LS}| = \frac{g_{k+1}^T y_k}{-g_k^T d_k} \leq \frac{\kappa LD}{cr^2}.$$

On the one hand, we have

$$\begin{aligned}\|d_{k+1}\| &= \|g_{k+1}\| + (1-\lambda_k)|\beta_k^{MDLS}|\|d_k\| + \lambda_k|\theta_k|\|g_k\| \\ &= \|g_{k+1}\| + (1-\lambda_k)|\beta_k^{MDLS}|\frac{\|s_k\|}{\alpha_k} + \lambda_k|\theta_k|\|g_k\|. \end{aligned} \tag{19}$$

From Lemma 2 and by using (14) and (15), we get two conditions:

1. If $\|g_{k+1}\|^2 \geq \left|g_{k+1}^T g_k\right|$, then (19) becomes

$$\|d_{k+1}\| \leq \kappa + \frac{\kappa LD}{\alpha^* cr^2} + \eta\kappa = C,$$

where C is a constant.

2. If $\|g_{k+1}\|^2 < \left|g_{k+1}^T g_k\right|$, but in this case we have $\left|g_{k+1}^T g_k\right| \geq \|g_{k+1}\|^2$, then (19) becomes

$$\|d_{k+1}\| \leq \kappa + \frac{\kappa^2}{\alpha^* cr^2} + 2\eta\kappa = C,$$

where C is a constant.

Now, we obtain

$$\sum_{k\geq 1} \frac{1}{\|d_{k+1}\|^2} = \infty. \tag{20}$$

Furthermore, combining inequalities (10) and (18) with the Zoutendijk condition (16), we obtain

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

which contradicts condition (20). Therefore, assumption (18) must be not true.

Consequently, we conclude that

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

□

4 Numerical experiments

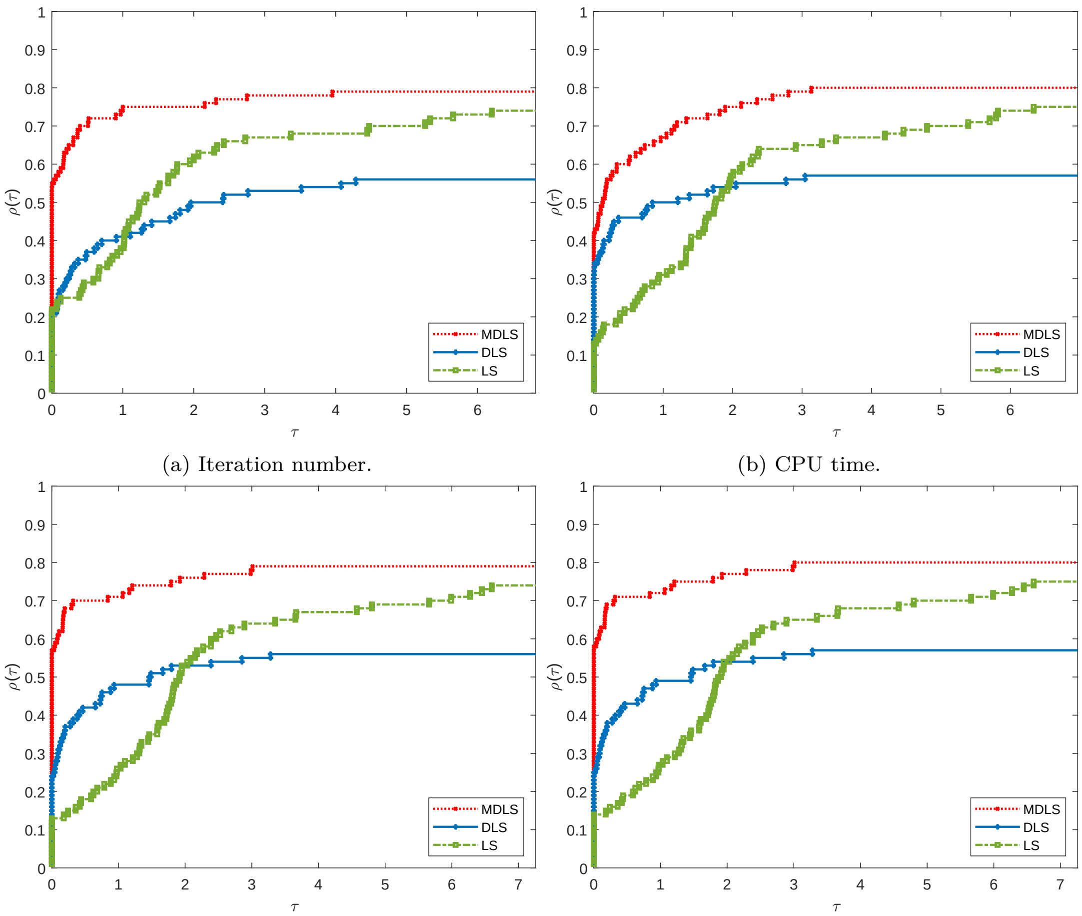
In this section, we present the performance of our new MDLS algorithm via some theoretical standard tests problems and practical applications.

4.1 Standard theoretical tests problems

We have selected various test functions given in Table 1, taken from the CUTE library of [1, 2, 17, 25] for different dimensions, from $n = 10$ to $n = 800000$. At the same time, we present a numerical comparison with other conjugate gradient algorithms (LS, DLS). We set the parameter $\eta = 0.4$, $\varepsilon = 10^{-6}$, and the stopping criterion is $\|g_k\| \leq \varepsilon$ or the iteration number > 4000. The stepsize α_k is calculated using the strong Wolfe line search with $\sigma = 0.01$, $\delta = 0.1$. In order to evaluate the data number of iterations, CPU time, gradient evaluation, and objective function evaluation, we use the performance profile of Dolan and Moré [13] presented in Figure 1.

Table 1: List of test functions and their dimensions.

Functions	**Dimensions**	**Functions**	**Dimensions**
Genrose	10000	Pen1	200, 1000
Osb2	10	Rosex	500, 1000
Eg2	100	Diagonal3	500, 2000
Freuroth	460	Singx	1000, 2000
Pen2	160	Diagonal1	800, 2000
Dixon3dq	150	Trid	500, 8000
Fletcbv3	100	Ie	500, 1500
Power1	150	Raydan1	500, 5000
Biggsb1	300	Lin	100, 1300
Dixmaani	360	Nonscomp	5000, 80000
Bv	2000, 20000	Dqrtic	5000, 150000
Liarwhd	6000, 30000	Dqdrtic	9000, 90000
Woods	150000, 200000	Dixmaana	6000, 90000
Quartc	80000, 50000	Dixmaanb	24000, 48000
Dixmaanc	2700, 27000	Dixmaand	12000, 90000
Dixmaane	2400, 48000	Dixmaanf	15000, 60000
Dixmaang	12000, 90000	Edensch	7000, 40000, 50000
Diagonal2	8000, 50000	Cosine	6000, 100000, 800000
Dixmaank	12000, 12000	Fletchcr	1000, 50000, 200000
Dixmaanl	2400, 24000	Exdenschnb	6000, 24000, 300000
Himmelbg	70000, 240000	Exdenschnf	90000, 280000, 600000
Dixmaanh	6000, 150000	Raydan2	2000, 20000, 500000
Dixmaanj	3000, 15000	Sine	100000, 250000, 500000



(a) Iteration number. (b) CPU time.

(c) Objective function evaluation. (d) Gradient evaluation.

Figure 1: Performance profile of MDLS, DLS and LS.

4.2 Practical applications

4.2.1 Image restoration

In this part, we evaluate the performance of the MDLS algorithm in comparison with the other conjugate gradient methods (LS and DLS) for image restoration problems affected by impulse noise. In this processing, we use the two-phase restoration scheme proposed by Chan, Ho, and Nikolova [7], which has proven effective in handling high-density of salt-and-pepper noise. This method was also used in [4, 18, 21, 23, 24], where it was shown that conjugate gradient-based algorithms can successfully recover images even when the corruption ratio is extremely high, reaching up to 90%.

Let $X \in \mathbb{R}^{M\times N}$ represent the corrupted image, and let $\mathcal{A} = \{1,\dots,M\} \times \{1,\dots,N\}$ be the index set of all pixels. The restoration process is formulated as the following minimization problem:

$$\min_u F_\alpha(u) = \sum_{(i,j)\in\mathcal{B}} \left(\sum_{(m,n)\in\mathcal{V}_{i,j}\setminus\mathcal{B}} 2\varphi_\alpha(u_{i,j} - y_{m,n}) + \sum_{(m,n)\in\mathcal{V}_{i,j}\cap\mathcal{B}} \varphi_\alpha(u_{i,j} - u_{m,n}) \right),$$

where $\mathcal{B} \subset \mathcal{A}$ denotes the set of pixels identified as corrupted during the first phase, and the 4-neighborhood of a pixel located at position (i,j) is defined as

$$\mathcal{V}_{i,j} = \{(i,j-1),\ (i,j+1),\ (i-1,j),\ (i+1,j)\}.$$

The observed image is represented by the set $y = \{y_{i,j}\}$, where $y_{i,j}$ denotes the intensity value at pixel (i,j). The edge-preserving potential function is given by $\varphi_\alpha(t) = \sqrt{t^2+\alpha}$, where $\alpha = 100$ is used in our experiments.

To evaluate the effectiveness of the restoration process, we use the Peak Signal-to-Noise Ratio (PSNR), which is defined as

$$PSNR = 10\log_{10}\left[\frac{255^2}{\frac{1}{MN}\sum_{i,j}(x^r_{i,j} - x^*_{i,j})^2}\right],$$

where $x^r_{i,j}$ and $x^*_{i,j}$ represent the pixel values of the reconstructed and reference images, respectively.

The experiments are conducted on three gray-scale images of size 512×512 *Lady*, *Baboon*, and *Bridge*. The stopping criterion is defined as either the number of iterations exceeding 300 or the relative change in the objective function satisfying

$$\frac{|\Delta F_\alpha(u_k)|}{F_\alpha(u_k)} \le 10^{-4}.$$

We test the performance of the algorithms under various levels of salt-and-pepper noise: 30%, 50%, 70%, and 90%.

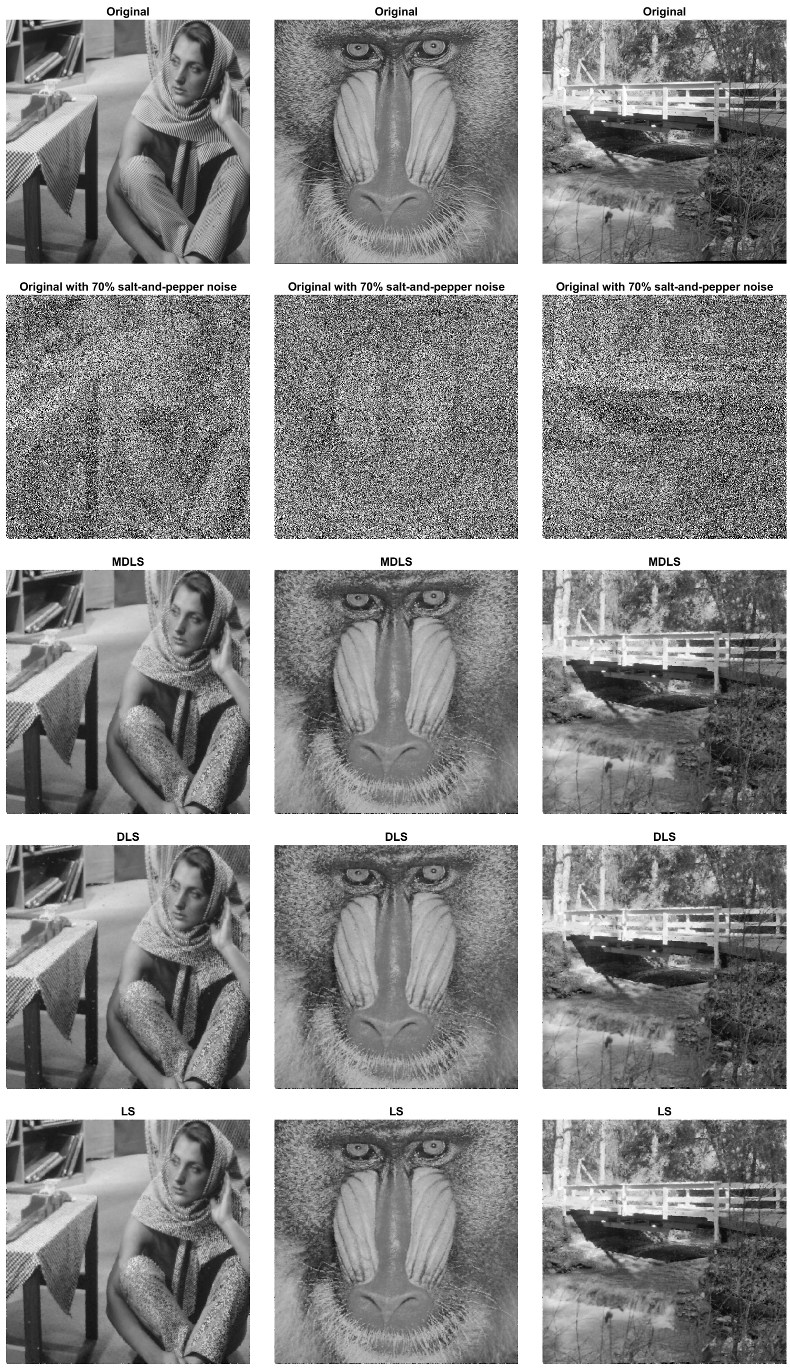


Figure 2: The noisy images with 70% salt-and-pepper noise and the restored images by MDLS, DLS, and LS.

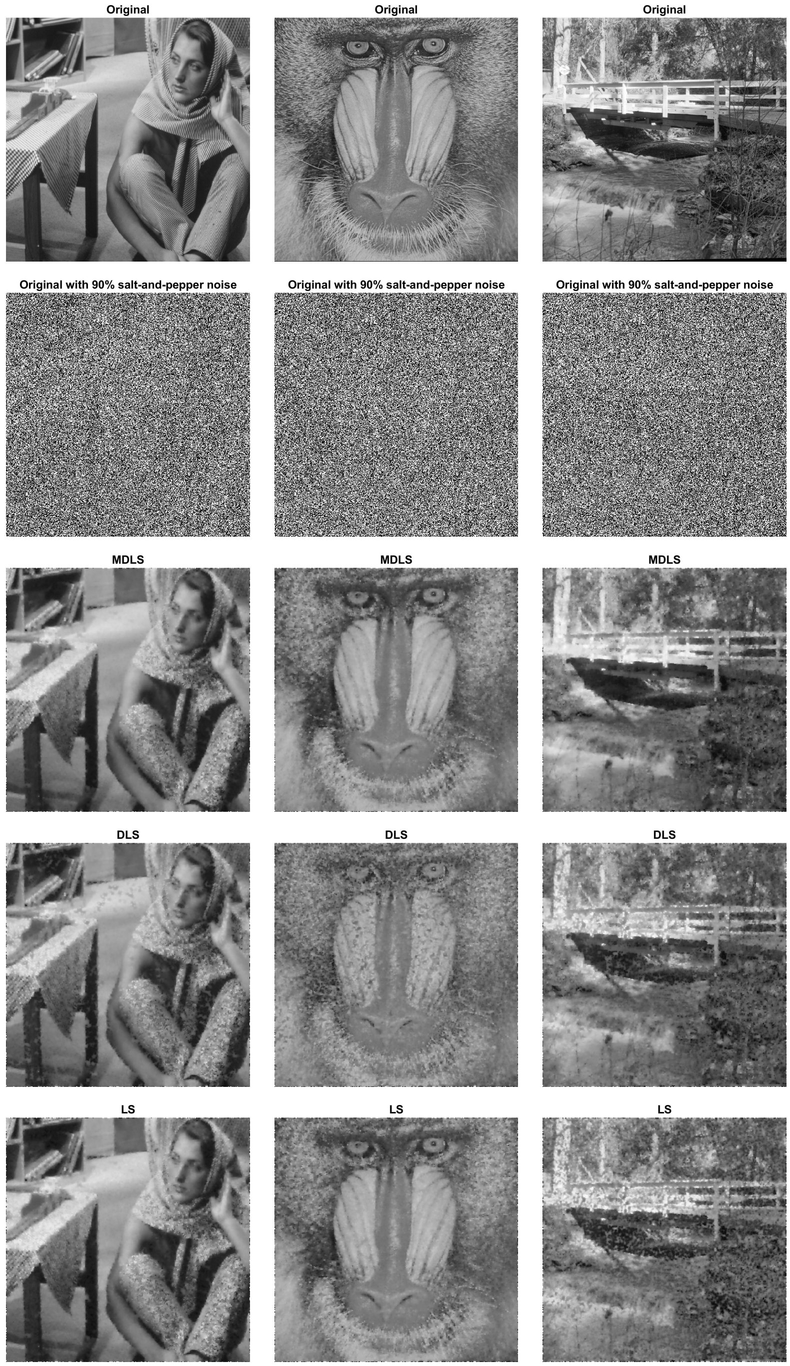


Figure 3: The noisy images with 90% salt-and-pepper noise and the restored images by MDLS, DLS, and LS.

Table 2: Performance results for different images and compression levels

Image/Noise	MDLS			DLS			LS		
	Iter	Time	PSNR	Iter	Time	PSNR	Iter	Time	PSNR
Barbara/0.30	16	8.73	28.39	55	13.71	28.40	15	14.40	28.27
Barbara/0.50	19	16.26	26.37	34	23.29	26.39	16	22.55	26.13
Barbara/0.70	16	21.43	24.36	56	37.22	24.34	24	43.75	24.37
Barbara/0.90	28	36.05	22.11	105	81.23	21.64	30	61.19	22.09
Baboon/0.30	18	7.52	27.91	39	11.95	27.89	15	9.11	27.84
Baboon/0.50	13	12.37	25.96	27	19.12	25.93	16	23.75	25.60
Baboon/0.70	18	28.81	23.80	75	38.74	23.75	16	25.47	23.75
Baboon/0.90	27	37.59	21.38	63	46.15	20.39	31	63.04	21.37
Bridge/0.30	15	8.54	28.54	31	11.35	28.39	14	8.23	28.46
Bridge/0.50	20	21.05	26.61	41	27.19	26.58	22	26.86	26.30
Bridge/0.70	24	23.80	24.45	88	55.35	24.36	26	46.82	24.42
Bridge/0.90	37	41.61	21.42	75	76.54	20.20	18	91.73	19.43

4.2.2 Sparse signal recovery

This part is dedicated to making a comparison of the three algorithms and analyzing the effectiveness of our proposed MDLS algorithm for solving sparse signal recovery problems. Sparse signal recovery is a critical problem in signal processing, where the objective is to recover a sparse signal from noisy or incomplete measurements. The optimization problem can be formulated as

$$\min_{x\in\mathbb{R}^n} f(x) = \frac{1}{2}\|Ax-b\|^2 + \theta\|x\|_1,$$

where $A \in \mathbb{R}^{m\times n}$ (with $m < n$) is the measurement matrix, $b \in \mathbb{R}^m$ represents the observed data, $\|x\|_1 = \mathbf{e}_n^T x$ with $\mathbf{e}_n = (1, 1, \ldots, 1)^T \in \mathbb{R}^n$, and $\theta > 0$ is a regularization parameter controlling the trade-off between data fidelity and sparsity [5]. The effectiveness of signal recovery is typically evaluated using the mean squared error (MSE) given by

$$\text{MSE} = \frac{1}{n}\|x^* - x\|^2,$$

where x^* is the recovered signal and x is the original sparse signal. Note that a low MSE means that the estimated signal is very close to the true signal, indicating better performance. In our tests, N represents the sparsity level, which equals 32. The size of the problems is $n = 4096$ and $m = 1024$. The regularization parameter is $\theta = 2.9 \times 10^{-3}$.

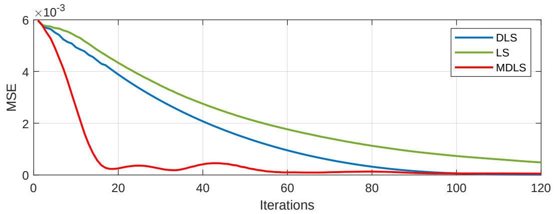


Figure 4: MSE in terms of iterations.

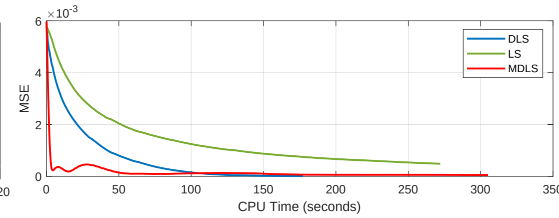


Figure 5: MSE in terms of CPU time.

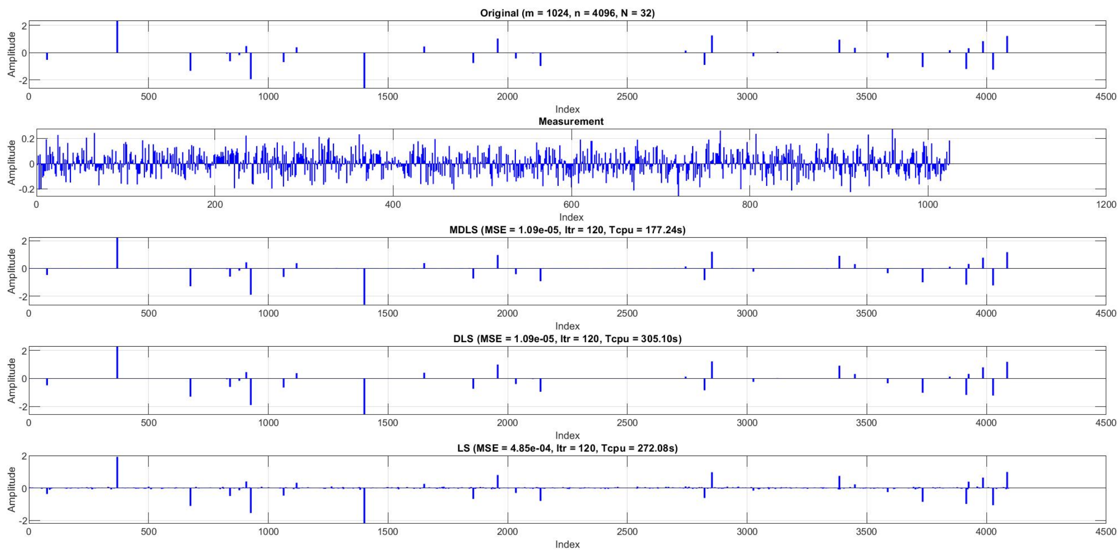


Figure 6: The original sparse signal compared to the restored signals.

Commentaries

The performance profile results, as depicted in Figures 1a, 1b, 1c, and 1d, show that our proposed conjugate gradient method MDLS, using β_k^{MDLS}, consistently outperforms methods based on β_k^{DLS} and β_k^{LS} for unconstrained optimization problems.

In the context of image restoration from Figures 2 and 3 and the results of Table 2, the MDLS method strikes an optimal balance between computational speed and image quality, particularly in scenarios with high levels of noise. It achieves superior PSNR values compared to other methods while requiring fewer iterations than the DLS and LS approaches.

For sparse signal restoration, Figures 4, 5, and 6 highlight the robustness and efficiency of the MDLS algorithm in addressing sparse signal recovery problems. It shows faster convergence and reduced computational costs, offering a significant improvement over the other considered algorithms.

5 Conclusion

We proposed a new hybrid conjugate gradient method for solving unconstrained optimization problems. Our method, called MDLS, uses a new switching rule between the LS and DLS formulas. This rule automatically makes sure the search direction is a descent direction, without needing other extra safeguards that are often added to hybrid conjugate gradient methods.

We also used a three-term search direction to improve the algorithm's performance. This helps keep the value of β_k stable, avoids poor search directions, and makes the method more reliable. We proved that the MDLS method satisfies the sufficient descent condition and globally converges under the strong Wolfe line search.

Numerical tests showed that the MDLS method is more efficient compared to the standard LS and DLS methods. It consistently reduced the number of iterations and computing time while performing well in practical problems like image restoration and sparse signal recovery. This shows its potential for solving various optimization problems.

Funding

This research received no external funding.

Conflicts of Interest

The authors declare no conflicts of interest.

References

[1] Andrei, N., *An unconstrained optimization test functions collection*, Adv. Model. Optim., 10 (2008), 147–161.

[2] Andrei, N., *Nonlinear conjugate gradient methods for unconstrained optimization*, Springer Optim. Appl., 158 (2020), 455–466.

[3] Ben Hanachi, S., Sellami, B. and Belloufi, M., *New iterative conjugate gradient method for nonlinear unconstrained optimization*, RAIRO Oper. Res., 56 (2022), 2315–2327.

[4] Cai, J., Chan, R. and Morini, B., *Minimization of an edge-preserving regularization functional by conjugate gradient type methods*, in *Image Processing Based on Partial Differential Equations*, Math. Vis., Springer, Berlin–Heidelberg, 2007.

[5] Candès, E., Romberg, J. and Tao, T., *Robust uncertainty principles: exact signal reconstruction from highly incomplete frequency information*, IEEE Trans. Inf. Theory, 52(2) (2006), 489–509.

[6] Fletcher, R., *Practical methods of optimization: unconstrained optimization*, Vol. 1, Wiley, New York, 1987.

[7] Chan, R., Ho, C. and Nikolova, M., *Salt-and-pepper noise removal by median-type noise detectors and detail-preserving regularization*, IEEE Trans. Image Process., 14(10) (2005), 1479–1485.

[8] Dai, Z. and Wen, F., *Another improved Wei–Yao–Liu nonlinear conjugate gradient method with sufficient descent property*, Appl. Math. Comput., 218(13) (2012), 7421–7430.

[9] Delladji, S., Belloufi, M. and Sellami, B., *New hybrid conjugate gradient method as a convex combination of FR and BA methods*, J. Inf. Optim. Sci., 42(3) (2021), 591–602.

[10] Djordjević, S., *New hybrid conjugate gradient method as a convex combination of HS and FR methods*, J. Appl. Math. Comput., 2(9) (2018), 366–378.

[11] Djordjević, S., *New hybrid conjugate gradient method as a convex combination of LS and CD methods*, Filomat, 31(6) (2017), 1813–1825.

[12] Djordjević, S., *New hybrid conjugate gradient method as a convex combination of LS and FR methods*, Acta Math. Sci. Ser. B, 39(1) (2019), 214–228.

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

[14] Dai, Y. and Yuan, Y., *An efficient hybrid conjugate gradient method for unconstrained optimization*, Ann. Oper. Res., 103 (2001), 33–47.

[15] Dai, Y. and Yuan, Y., *A nonlinear conjugate gradient method with a strong global convergence property*, SIAM J. Optim., 10(1) (1999), 177–182.

[16] Fletcher, R. and Reeves, C., *Function minimization by conjugate gradients*, Comput. J., 7(2) (1964), 149–154.

[17] Gould, N., Orban, D. and Toint, P., *CUTEr and SifDec: a constrained and unconstrained testing environment, revisited*, ACM Trans. Math. Softw., 29(4) (2003), 373–394.

[18] Hemici, Y. E., Khelladi, S. and Benterki, D., *New hybrid conjugate gradient method for nonlinear optimization with application to image restoration problems*, Kybernetika, 60(4) (2024), 535–552.

[19] Hestenes, M. and Stiefel, E., *Methods of conjugate gradients for solving linear systems*, J. Res. Natl. Bur. Stand., 49(6) (1952), 409–436.

[20] Hager, W. and Zhang, H., *A survey of nonlinear conjugate gradient methods*, Pac. J. Optim., 2 (2006), 35–58.

[21] Jiang, X., Liao, W., Yin, J. and Jian, J., *A new family of hybrid three-term conjugate gradient methods with applications in image restoration*, Numer. Algorithms, 91(1) (2022), 161–191.

[22] Liu, Y. and Storey, C., *Efficient generalized conjugate gradient algorithm. Part I: theory*, J. Optim. Theory Appl., 69(1) (1991), 129–137.

[23] Ma, G., Lin, H., Jin, W. and Han, D., *Two modified conjugate gradient methods for unconstrained optimization with applications in image restoration problems*, J. Appl. Math. Comput., 68 (2022), 4733–4758.

[24] Malik, M., Sulaiman, I., Abubakar, A., Ardaneswari, G. and Sukono, A., *A new family of hybrid three-term conjugate gradient method for unconstrained optimization with application to image restoration and portfolio selection*, AIMS Math., 8(1) (2023), 1–28.

[25] Moré, J., Garbow, B. and Hillstrom, K., *Testing unconstrained optimization software*, ACM Trans. Math. Softw., 7(1) (1981), 17–41.

[26] Mtagulwa, P. and Kaelo, P., *A convergent modified HS–DY hybrid conjugate gradient method for unconstrained optimization problems*, J. Inf. Optim. Sci., 40(1) (2019), 97–113.

[27] Ouaoua, M. L., Khelladi, S. and Benterki, D., *New parameter of conjugate gradient method for unconstrained nonlinear optimization*, Stat. Optim. Inf. Comput., 13(6) (2025), 2382–2390.

[28] Polyak, B., *The conjugate gradient method in extreme problems*, USSR Comput. Math. Math. Phys., 9 (1969), 94–112.

[29] Polak, E. and Ribière, G., *Note sur la convergence des méthodes de directions conjuguées*, Rev. Française Inform. Rech. Opér., 16 (1969), 35–43.

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

[31] Sellami, B. and Chaib, Y., *A new family of globally convergent conjugate gradient methods*, Ann. Oper. Res., 241 (2016), 497–513.

[32] Sellami, B. and Chaib, Y., *New conjugate gradient method for unconstrained optimization*, RAIRO Oper. Res., 50 (2016), 1013–1026.

[33] Stanimirović, P., Ivanov, B. and Mosić, D., *A survey of gradient methods for solving nonlinear optimization*, Electron. Res. Arch., 28(4) (2020), 1573–1624.

[34] Sulaiman, I. M., Kaelo, P., Khalid, R. and Nawawi, M. K. M., *A descent generalized RMIL spectral gradient algorithm for optimization problems*, Int. J. Appl. Math. Comput. Sci., 34(2) (2024), 225–233.

[35] Yang, X., Luo, Z. and Dai, X., *A global convergence of LS–CD hybrid conjugate gradient method*, Adv. Numer. Anal., 2013 (2013), Article ID 5.

[36] Zhang, Y. and Zheng, B., *Two new Dai–Liao-type conjugate gradient methods for unconstrained optimization problems*, J. Optim. Theory Appl., 175 (2017), 502–509.

[37] Zoutendijk, G., *Nonlinear programming, computational methods*, in *Integer and Nonlinear Programming*, North-Holland, Amsterdam, 1970, pp. 37–86.