BAREKENG: Journal of Mathematics and Its Applications
March 2026 Volume 20 Issue 1 Page 0481–0492
P-ISSN: 1978-7227 E-ISSN: 2615-3017
https://doi.org/10.30598/barekengvol20iss1pp0481-0492

NEW CONJUGATE GRADIENT METHOD FOR ACCELERATED CONVERGENCE AND COMPUTATIONAL EFFICIENCY IN UNCONSTRAINED OPTIMIZATION PROBLEMS

Basim A. Hassan [1*], **Alaa Luqman Ibrahim** [2], **Thaair A. Ameen** [3], **Ibrahim Mohammed Sulaiman** [4]

[1] Department of Mathematics, College of Computer Sciences and Mathematics, University of Mosul Mosul, 41002, Iraq

[2] College of Science, University of Zakho Zakho, Kurdistan Region, 41002, Iraq

[3] Mosul University Presidency, University of Mosul Mosul, 41002, Iraq

[4] Institute of Strategic Industrial Decision Modelling, School of Quantitative Sciences, Universiti Utara Malaysia Kedah, Sintok, 06010, Malaysia

[4] Faculty of Education and Arts, Sohar University Sultanate of Oman, Sohar, 311, Oman

*Corresponding author's e-mail: ** **basimah@uomosul.edu.iq**

Article Info

Article History:

Received: 12th April 2025
Revised: 17th June 2025
Accepted: 4th August 2025
Available online: 24th November 2025

Keywords:

Conjugate gradient;
Global convergence:
Optimization;
Sufficient descent.

ABSTRACT

Conjugate gradient (CG) algorithms play an important role in solving large-scale unconstrained optimization problems due to their low memory requirements and strong convergence properties. However, many classical CG algorithms suffer from inefficiencies when dealing with complex or ill-conditioned objective functions. This paper addresses this challenge by proposing a new conjugate gradient method that combines the descent direction of traditional CG algorithms with Newton-type updates to improve convergence and computational efficiency. The proposed method is constructed to ensure sufficient descent at each iteration and global convergence under standard assumptions. By integrating the modified Newton update mechanism, the method effectively accelerates convergence without incurring the high computational cost typically associated with full Newton methods. To evaluate the performance of the proposed approach, we conducted extensive numerical experiments using a collection of well-known benchmark functions from the CUTEr test suite. The results show that the new method consistently outperforms the classical Hestenes-Stiefel method in terms of CPU time, number of function evaluations, and iteration count. These findings confirm the method's potential as an efficient and robust alternative for solving large-scale unconstrained optimization problems.

How to cite this article:

B. A. Hassan, A. L. Ibrahim, T. A. Ameen, and I. M. Sulaiman, "NEW CONJUGATE GRADIENT METHOD FOR ACCELERATED CONVERGENCE AND COMPUTATIONAL EFFICIENCY IN UNCONSTRAINED OPTIMIZATION PROBLEMS", *BAREKENG: J. Math. & App.*, vol. 20, iss. 1, pp. 0481-0492, Mar, 2026.

Journal homepage: *https://ojs3.unpatti.ac.id/index.php/barekeng/*
Journal e-mail: *barekeng.math@yahoo.com; barekeng.journal@mail.unpatti.ac.id*

Research Article · **Open Access**

1. INTRODUCTION

Optimization plays a crucial role in various scientific and engineering applications by seeking the minimum or maximum of an objective function. Among the many branches of optimization, unconstrained optimization is particularly important as it deals with the minimization of real-valued functions without any imposed constraints. This process typically takes place in a metric space, which provides a mathematical structure for measuring distances and ensuring convergence of iterative algorithms [1], [2]. Over the years, a wide range of numerical methods have been developed to solve such problems efficiently, including the Steepest Descent (SD), Newton's method, Conjugate Gradient (CG) methods, and Quasi-Newton (QN) methods.

Among these techniques, conjugate gradient methods have gained prominence due to their efficiency and low memory requirements, especially for problems of large-scale. These algorithms have been extensively applied in various domains. In engineering, for instance, CG methods are employed to address challenges in structural design, fluid dynamics, and control systems [3]. In data science and machine learning, they are instrumental in optimizing loss functions in regression, classification tasks, and neural network training [4], [5], [6], [7], [8]. Signal and image processing also benefit from CG techniques, which are used for signal recovery, image de-noising, and reconstruction [9], [10], [11]. Moreover, scientific computing relies on CG algorithms to solve large sparse linear systems, which are commonly encountered in computational physics and finite element analysis.

The classical CG formulas, including the pioneering works by Hestenes and Stiefel (HS) [12] and Fletcher and Reeves (FR) [13], laid the foundation for iterative optimization using gradient-based updates. Since then, numerous enhancements and modifications have been proposed in the literature to improve their convergence rate, robustness, and applicability to real-world problems [14], [15], [16]. Despite their advantages, traditional CG algorithms may still face limitations when faster convergence is required or when applied to ill-conditioned problems. Therefore, ongoing research has focused on developing new variants that maintain the numerical efficiency and theoretical strengths of CG methods while addressing practical performance bottlenecks.

This study aims to contribute to this ongoing effort by introducing some novel conjugate gradient methods tailored for large-scale unconstrained optimization. The new algorithms seek to improve computational efficiency and convergence speed by incorporating a Newton-inspired update strategy. The primary objective of this research is to develop algorithms that not only ensure global convergence under standard conditions but also outperform existing CG methods on practical benchmark problems.

The structure of this paper is as follows: Section 2 introduces the new conjugate gradient algorithms and describes their corresponding formulation. Section 3 establishes the sufficient descent condition and analyzes the convergence properties of the proposed methods, presenting further numerical experiments that compare our methods with the Hestenes-Stiefel method based on the number of iterations, function evaluations, and computation time using well-known benchmark problems. Finally, Section 4 provides a brief conclusion.

2. RESEARCH METHODS

The general form of an unconstrained optimization problem can be written as:

$$Min\{f(x): x \in R^n\} \tag{1}$$

where is $f(x)$ a smooth function with an available gradient $\nabla f(x)$ [17], [18]. The fundamental principle of all CG methods is the iterative generation of a sequence $\{x_k\}$ from an initial estimate $x_0 \in R^n$, following the recurrence relation:

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

where α_k is the step size, determined through a one-dimensional search known as a line search [19]. Line searches can be classified as either exact or inexact, with the latter being generally preferred due to the high computational cost of exact line searches in terms of both time and memory [20], [21], [22]. In our research, we employ a strong Wolfe line search, which is defined by the following conditions [23], [24]:

$$\begin{aligned} f(x_k + \alpha_k d_k) - f(x_k) &\leq \delta \alpha_k g_k^T d_k, \\ |g_{k+1}^T d_k| &\leq -\sigma g_k^T d_k. \end{aligned} \tag{3}$$

where $0 < \delta \leq \sigma < 1$. The step size α_k is chosen to minimize the function f, along $x_k + \alpha d_k$, and it is given by:

$$\alpha_k = -\frac{g_k^T d_k}{d_k^T G d_k}. \tag{4}$$

The search direction d_k is updated using the formula:

$$d_{k+1} = -g_{k+1} + \beta_k d_k. \tag{5}$$

where g_k represents the gradient of the function f at the point x_k, and β_k is a scalar known as the CG parameter. The choice of β_k defines the specific CG variant being used. One well-known formulation is the Hestenes-Stiefel CG method [12] with its formula defined as:

$$\beta_k^{HS} = \frac{g_{k+1}^T y_k}{d_k^T y_k}. \tag{6}$$

This HS algorithm is one of the foundational approaches for unconstrained optimization. It updates the search direction by combining the negative gradient with the previous direction, using a parameter designed to maintain conjugacy. However, despite its effectiveness, the HS algorithm has some notable drawbacks. One key limitation is its sensitivity to line search accuracy; poor line search conditions can lead to a loss of descent direction, causing the method to stall or converge slowly. Additionally, in practical implementations, the method may occasionally suffer from numerical instabilities or fail to maintain sufficient descent, especially for ill-conditioned or non-quadratic problems. These limitations have motivated the development of modified CG algorithms that improve robustness and convergence behavior.

2.1 Method Derivation

In this section, we construct the proposed CG method that integrates Newton updates to enhance convergence properties and computational efficiency. The study begins by examining the fundamental principles of Newton's method and then derives key formulations that lead to the improved CG method.

If the current point is sufficiently close to a local minimizer, the pure Newton method relies on the direction of the search, given by:

$$d_{k+1} = -Q_{k+1}^{-1} g_{k+1}.$$

Nazareth [25] reformulated this equation as:

$$-Q_{k+1}^{-1} g_{k+1} = -g_{k+1} + \beta_k s_k \tag{7}$$

where $s_k = \alpha_k d_k$. Building on this principle, this study integrates Newton updates to develop an improved CG method with enhanced convergence properties and computational efficiency. Using Taylor series expansion, we derive all secant equations and establish a unified QN equation [26]:

$$s_k^T Q_{k+1} s_k = s_k^T y_k + \omega_k [2(f_k - f_{k+1}) + (g_k + g_{k+1})^T s_k] \tag{8}$$

where $\omega_k \in [0,1,2,3]$. This equation utilizes a class of modified quasi-Newton updates to improve accuracy in approximating the second-order curvature of the objective function. By manipulating Eq. (8), we obtain:

$$s_k^T Q_{k+1} s_k = \frac{[s_k^T g_k]^2}{s_k^T y_k + \omega_k [2(f_k - f_{k+1}) + (g_k + g_{k+1})^T s_k]} \tag{9}$$

which yields the matrix $Q(x_k)$ as:

$$Q_{k+1} = \frac{\dfrac{[s_k^T g_k]^2}{s_k^T y_k + \omega_k [2(f_k - f_{k+1}) + (g_k + g_{k+1})^T s_k]}}{s_k^T s_k} I_{k*k}. \tag{10}$$

Substituting Eq. (10) in Eq. (7) we get:

$$\beta_k = \left(1 - \frac{s_k^T s_k}{\frac{[s_k^T g_k]^2}{s_k^T y_k + \omega_k[2(f_k - f_{k+1}) + (g_k + g_{k+1})^T s_k]}} \right) \frac{g_{k+1}^T y_k}{s_k^T y_k} \tag{11}$$

where $\omega_k \in [0,1,2,3]$. To ensure sufficient descent condition, we perform additional algebraic manipulations to obtain:

$$\beta_k = \frac{1}{s_k^T y_k} \left(y_k - \rho \frac{\|y_k\|^2}{s_k^T y_k} s_k \right)^T g_{k+1} \tag{12}$$

where:

$$\rho = \frac{(s_k^T y_k)}{\|y_k\|^2} \left[\frac{s_k^T y_k}{s_k^T s_k} * \frac{s_k^T s_k}{\frac{[s_k^T g_k]^2}{s_k^T y_k + \omega_k[2(f_k - f_{k+1}) + (g_k + g_{k+1})^T s_k]}} \right] \tag{13}$$

The proposed method is denoted as **New**, where:

New1 applies when $\omega_k = 0$.
New2 applies when $\omega_k = 1$.
New3 applies when $\omega_k = 2$.
New4 applies when $\omega_k = 3$.

The implementation steps of the proposed method are outlined in the following algorithm:

Algorithm:

Step 1 : (Initialization) Given an initial point $x_0 \in R^n$, parameters $0 < \delta < \sigma < 1$, and $\varepsilon > 0$. Set $d_0 = -g_0$, and $k = 0$. If $\|g_k\| \leq \varepsilon$ then stop. Compute the step size α_k by the weak Wolfe line search
Step 2 : Determine the next iteration by Eq. (2).
Step 3 : Compute d_{k+1} by Eq. (5) and choose an appropriate conjugate parameter β_k by Eqs. (12) and (13).
Step 4 : Set $k \coloneqq k + 1$ and go to Step 1.

3. RESULTS AND DISCUSSION

3.1. Global Convergence

This section presents the main properties of the proposed algorithm, focusing on the condition of sufficient descent and global convergence

Theorem 1. *Let* $\boldsymbol{s_k, y_k, g_{k+1} \in R^n, \beta_k \in R}$ *and* $\boldsymbol{\beta_k = \frac{1}{s_k^T y_k}\left(y_k - \rho \frac{\|y_k\|^2}{s_k^T y_k} s_k\right)^T g_{k+1}}$ *. If* $\boldsymbol{s_k^T y_k \neq 0}$*, then* $\boldsymbol{d_{k+1}^T g_{k+1} \leq -\left[1 - \frac{1}{4\rho}\right] \|g_{k+1}\|^2}$.

Proof. Using induction. Since $d_0 = -g_0$, we have $g_0^T d_0 = -\|g_0\|^2$. Assuming $d_k^T g_k \leq -c\|g_k\|^2$ holds, then by multiplying Eq. (5) by g_{k+1}, we have:

$$d_{k+1}^T g_{k+1} = -\|g_{k+1}\|^2 + \left(\frac{g_{k+1}^T y_k}{s_k^T y_k} - \rho \frac{\|y_k\|^2}{(s_k^T y_k)^2} g_{k+1}^T s_k \right) s_k^T g_{k+1}. \tag{14}$$

which simplifies to:

$$d_{k+1}^T g_{k+1} = \frac{(g_{k+1}^T y_k)(s_k^T g_{k+1})(s_k^T y_k) - \|g_{k+1}\|^2 (s_k^T y_k)^2 - \rho \|y_k\|^2 (g_{k+1}^T s_k)^2}{(s_k^T y_k)^2}. \tag{15}$$

By applying:

$$w = \frac{1}{\sqrt{2\rho}}\left(s_k^T y_k\right) g_{k+1} \quad \text{and} \quad v = \sqrt{2\rho}\left(g_{k+1}^T s_k\right) y_k \tag{16}$$

to the inequality:

$$w^T v \leq \frac{1}{2}(\|w\|^2 + \|v\|^2) \tag{17}$$

we obtain:

$$\left(g_{k+1}^T y_k\right)\left(s_k^T g_{k+1}\right)\left(s_k^T y_k\right) \leq \frac{1}{2}\left[\frac{1}{2\rho}(s_k^T y_k)^2 \|g_{k+1}\|^2 + 2\rho(s_k^T g_{k+1})^2 \|y_k\|^2\right]. \tag{18}$$

Thus, by Eqs. (15) and (18), we get:

$$d_{k+1}^T g_{k+1} \leq \frac{\left[\frac{1}{4\rho} - 1\right](s_k^T y_k)^2 \|g_{k+1}\|^2 + [\rho - \rho](s_k^T g_{k+1})^2 \|y_k\|^2}{(s_k^T y_k)^2} \tag{19}$$

which leads to:

$$d_{k+1}^T g_{k+1} \leq -\left[1 - \frac{1}{4\rho}\right] \|g_{k+1}\|^2. \tag{20}$$

Thus, completes our proof. ■

Now, we study the global convergence properties of the proposed conjugate gradient method by making specific assumptions and presenting the main lemmas.

Assumption 1. The level set $S = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$ is bounded, i.e., there exists a constant $B > 0$ such that

$$\| x \| \leq B, \quad \text{for all } x \in S\,. \tag{21}$$

Assumption 2. The level set $S = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$ is bounded, i.e., there exists a constant $B > 0$ such that

$$\| \nabla f(x) - \nabla f(y) \| \leq L \| x - y \|, \text{for all } x, y \in \mathcal{N}. \tag{22}$$

Under Assumptions 1 and 2 on f, there exists a constant $\Gamma \geq 0$, such that

$$\| \nabla f(x) \| \leq \Gamma. \tag{23}$$

Lemma 1. *The level set $S = \{x \in \mathbb{R}^n : f(x) \leq f(x_0)\}$ is bounded, i.e., there exists a constant $B > 0$ such that*

$$\sum_{k \geq 0} \frac{1}{\|d_{k+1}\|^2} = \infty, \tag{24}$$

Then

$$\lim_{k \to \infty} \inf \|g_{k+1}\| = 0. \tag{25}$$

The proof of this lemma can be found in [25].

Theorem 2. *Suppose all assumptions hold. Let the sequences $\{\boldsymbol{x_k}\}$ and $\{\boldsymbol{d_k}\}$ be generated by the new method. If step size $\boldsymbol{\alpha_k}$ satisfies Wolfe conditions, then the following holds:*

$$\lim_{k \to \infty} \inf \|g_k\| = 0. \tag{26}$$

Proof. From the search direction in Eq. (5) and the definition of β_k in Eq. (12), we obtain the following:

$$\begin{aligned} \|d_{k+1}\| &= \left\|-g_{k+1} + \beta_k^{\text{New}} d_k\right\| \\ &\leq \|g_{k+1}\| + \left|\beta_k^{\text{New}}\right| \|d_k\| \\ &\leq \|g_{k+1}\| + \left\|\left(y_k - \rho \frac{\|y_k\|^2}{s_k^T y_k} s_k\right)\right\| \frac{\|g_{k+1}\|}{\|s_k\| \|y_k\|} \|d_k\| \end{aligned} \tag{27}$$

$$\leq \|g_{k+1}\| + \frac{\|y_k\|\|g_{k+1}\| + \rho\frac{\|g_{k+1}\|\|y_k\|^2\|s_k\|}{\|s_k\|\|y_k\|}}{\alpha_k\|d_k\|\|y_k\|}\|d_k\|$$
$$\leq \|g_{k+1}\| + \frac{\|y_k\|\|g_{k+1}\| + \rho\|g_{k+1}\|\|y_k\|}{\alpha_k\|d_k\|\|y_k\|}\|d_k\|$$
$$\leq \left[1 + \frac{1}{\alpha_k} + \frac{\rho}{\alpha_k}\right]\|g_{k+1}\|$$
$$\leq \left[\frac{\alpha_k + 1 + \omega}{\alpha_k}\right]\|g_{k+1}\|.$$

Therefore, we have:

$$\sum_{k\geq 1}\frac{1}{\|d_k\|^2} \geq \left(\frac{\alpha_k}{\alpha_k+1+\rho}\right)\frac{1}{\Gamma}\sum_{k\geq 1} 1 = \infty. \tag{28}$$

Using Lemma 1, implies that $\lim_{k\to\infty} inf\|g_k\| = 0$. ■

3.2. Numerical Computation

In this section, we present the results of numerical experiments conducted to assess the performance of the proposed algorithm, compared to the well-established HS algorithm, using several benchmark optimization problems. All tests were implemented in MATLAB R2013a and executed on a laptop with the following specifications: Windows 10 operating system, HP computer (Intel(R) Core (TM) i7-6600U CPU @ 2.60GHz, 2.81 GHz) with 8 GB of RAM.

The experiments evaluate the performance of the proposed algorithm under typical conditions. Specifically, the parameters used in the implementation are $\delta = 0.01$ and $\sigma = 0.3$. Instances where algorithms fail to converge are marked with the symbol 'NaN'. The stopping criterion for convergence is set to $\|g_{k+1}\| \leq 10^{-6}$.

The test problems used in these experiments are sourced from the CUTE library [27], as well as additional unconstrained problem sets from [28] and [29]. These problems vary in their dimensionality, ensuring a robust evaluation across different types of optimization challenges.

For each method, the performance is evaluated based on several key metrics, including the number of iterations (NOI), the number of function evaluations (NOF), and the CPU time (CPUT) required to reach convergence. A summary of these results is presented in Table 1. To facilitate a more comprehensive comparison, we also utilize performance profiles, as proposed by Dolan and Moré [30], to visually represent the performance of each algorithm in terms of CPUT, NOI, and NOF. In these profiles, the curve that is highest indicates the superior performance of the corresponding method, [31], [32].

The graphical results, displayed in Figs. 1, 2, and 3 clearly illustrate the effectiveness of the proposed algorithm. These figures highlight the performance improvements achieved by our method compared to the HS algorithm, showcasing its efficiency in both computational time and convergence speed across the various test problems.

Table 1. Summary of Numerical Results Comparing the Proposed Algorithm with the Hestenes-Stiefel (HS) Method. The Table Includes the NOI, NOF and CPUT for Each Method Across the Test Problems

Test Function	N	HS			New1			New2			New3			New4		
		NOI	NOF	CPUT	NOI	NOF	CPUT	NOI	NOF	CPUT	NOI	NOF	CPUT	NOI	NOF	CPUT
'cosine'	500	342	938	0.190	24	89	0.017	24	115	0.020	24	99	0.018	28	110	0.020
'cosine'	1000	NaN	NaN	NaN	31	122	0.034	32	129	0.037	31	93	0.027	33	103	0.028
'cosine'	5000	NaN	NaN	NaN	27	91	0.121	25	89	0.121	29	103	0.139	31	110	0.146
'cosine'	10000	NaN	NaN	NaN	25	99	0.245	24	88	0.214	26	90	0.218	31	112	0.266
'dixmaanc'	3000	31	170	0.772	25	116	0.554	32	115	0.485	27	98	0.407	38	134	0.602
'dixmaanc'	15000	31	169	3.293	26	109	2.066	36	157	3.093	29	117	2.333	35	162	3.009
'dixmaanf'	15000	717	1062	21.177	613	893	17.547	506	769	15.487	553	829	16.694	685	1038	20.375
'dixmaanh'	300	110	211	0.176	110	189	0.106	98	171	0.113	110	188	0.107	107	197	0.082
'dixmaanh'	1500	220	359	1.004	193	291	0.763	228	364	0.986	219	364	0.731	237	382	0.939

Test Function	N	HS			New1			New2			New3			New4		
		NOI	NOF	CPUT	NOI	NOF	CPUT	NOI	NOF	CPUT	NOI	NOF	CPUT	NOI	NOF	CPUT
'dixmaani'	300	1835	2688	1.891	1301	1851	1.303	1164	1704	1.262	1213	1807	1.396	1332	1905	1.508
'dixmaanj'	300	1208	1778	1.521	891	1304	0.793	1135	1620	1.240	1145	1637	1.258	627	922	0.722
'dixmaank'	1500	709	1039	3.846	355	527	1.839	290	435	1.715	354	521	1.953	295	456	1.627
'dixmaank'	30000	523	770	52.471	521	780	72.172	501	738	51.245	515	780	57.181	522	828	54.950
'dixmaanl'	1500	361	542	2.000	279	432	1.405	397	588	2.377	272	396	1.746	292	450	2.141
'dixmaanl'	3000	324	503	4.332	307	471	4.538	286	442	3.681	282	437	3.443	294	476	3.940
'dixon3dq'	4	36	84	0.017	34	85	0.008	34	85	0.009	34	85	0.009	34	85	0.009
'dixon3dq'	10	95	173	0.016	74	146	0.019	74	146	0.019	74	146	0.016	74	146	0.018
'dixon3dq'	50	500	742	0.071	434	612	0.080	466	665	0.082	482	674	0.085	435	612	0.072
'dixon3dq'	150	NaN	NaN	NaN	1984	2810	0.438	984	1408	0.207	1990	2804	0.427	1344	1898	0.314
'dqdrtic'	500	63	215	0.034	82	217	0.032	71	195	0.031	81	204	0.113	66	205	0.037
'dqdrtic'	1000	57	200	0.036	52	158	0.032	65	214	0.042	67	193	0.043	60	167	0.034
'dqdrtic'	5000	87	225	0.167	74	229	0.182	67	226	0.177	74	215	0.177	70	213	0.161
'dqdrtic'	10000	80	276	0.396	71	208	0.315	74	209	0.330	66	197	0.316	61	199	0.388
'edensch'	500	41	120	0.052	35	95	0.053	37	109	0.062	30	89	0.051	40	102	0.059
'edensch'	1000	38	129	0.125	35	96	0.060	37	89	0.048	37	96	0.055	36	105	0.066
'edensch'	5000	59	209	0.663	35	107	0.357	37	108	0.347	37	103	0.325	42	133	0.434
'edensch'	10000	66	366	2.211	66	322	1.936	42	114	0.709	40	113	0.823	34	103	0.615
'eg2'	4	62	269	0.019	33	105	0.005	34	86	0.004	26	92	0.005	29	108	0.005
'eg2'	10	170	527	0.019	42	125	0.007	55	187	0.008	41	135	0.005	48	166	0.008
'eg2'	100	76	340	0.032	153	543	0.054	84	306	0.028	127	431	0.035	168	610	0.063
'freuroth'	4	118	332	0.047	94	277	0.016	92	269	0.016	105	262	0.017	69	220	0.015
'freuroth'	10	143	581	0.032	78	233	0.014	148	319	0.021	115	516	0.026	106	245	0.017
'genrose'	5	136	301	0.046	113	268	0.024	133	279	0.027	141	314	0.034	138	274	0.025
'genrose'	10	231	396	0.028	179	345	0.024	255	432	0.032	152	311	0.032	180	343	0.033
'genrose'	100	1040	1532	0.137	1029	1518	0.197	1025	1497	0.177	959	1436	0.119	921	1351	0.160
'himmelbg'	500	2	9	0.035	2	9	0.002	2	9	0.002	2	9	0.003	2	9	0.002
'himmelbg'	1000	2	9	0.004	2	9	0.004	2	9	0.004	2	9	0.004	2	9	0.004
'himmelbg'	5000	4	24	0.039	3	21	0.019	3	21	0.021	3	21	0.024	3	21	0.020
'himmelbg'	10000	2	11	0.027	2	11	0.025	2	11	0.028	2	11	0.027	2	11	0.025
'tridia'	100	399	601	0.034	316	472	0.030	363	519	0.032	305	453	0.027	352	523	0.034
'woods'	500	143	373	0.038	116	325	0.028	144	418	0.033	129	329	0.026	134	398	0.031
'woods'	1000	145	340	0.041	94	294	0.037	119	293	0.039	127	354	0.045	156	373	0.045
'woods'	5000	185	436	0.224	171	430	0.216	208	463	0.252	160	448	0.227	193	506	0.263
'woods'	10000	180	485	0.451	145	326	0.320	175	506	0.471	168	504	0.476	156	422	0.406
'bdexp'	500	NaN	NaN	NaN	2	7	0.002	2	7	0.002	2	7	0.002	2	7	0.002
'bdexp'	1000	NaN	NaN	NaN	3	18	0.005	3	18	0.005	3	18	0.005	3	18	0.005
'bdexp'	5000	NaN	NaN	NaN	3	19	0.031	3	19	0.029	3	19	0.028	3	19	0.030
'bdexp'	10000	NaN	NaN	NaN	2	9	0.038	2	9	0.041	2	9	0.044	2	9	0.039
'exdenschnf'	500	35	140	0.020	31	147	0.017	35	118	0.014	39	153	0.018	36	151	0.018
'exdenschnf'	1000	33	171	0.032	38	124	0.026	43	152	0.029	40	189	0.033	35	160	0.028
'exdenschnf'	5000	37	156	0.124	37	170	0.146	33	165	0.137	35	128	0.108	30	181	0.148
'exdenschnf'	10000	32	143	0.226	38	172	0.277	52	171	0.277	29	156	0.248	40	155	0.247
'exdenschnb'	500	19	81	0.011	27	124	0.007	23	95	0.006	26	124	0.009	26	82	0.006
'exdenschnb'	1000	30	128	0.013	28	89	0.010	34	124	0.012	24	88	0.009	32	108	0.011
'exdenschnb'	5000	30	144	0.053	27	103	0.040	25	123	0.047	25	110	0.041	27	93	0.038

Test Function	N	HS			New1			New2			New3			New4		
		NOI	NOF	CPUT	NOI	NOF	CPUT	NOI	NOF	CPUT	NOI	NOF	CPUT	NOI	NOF	CPUT
'exdenschnb'	10000	24	116	0.078	27	110	0.075	34	126	0.092	31	151	0.106	44	141	0.099
'genquartic'	500	22	103	0.010	24	85	0.006	33	97	0.007	27	109	0.008	22	74	0.006
'genquartic'	1000	34	106	0.012	22	72	0.008	26	108	0.014	25	96	0.011	23	104	0.012
'genquartic'	5000	83	296	0.128	25	84	0.038	23	114	0.051	26	91	0.041	24	98	0.044
'genquartic'	10000	80	344	0.274	31	125	0.103	30	114	0.097	26	116	0.095	27	126	0.099
'biggsb1'	4	12	40	0.006	22	85	0.002	22	85	0.003	22	85	0.002	22	85	0.002
'biggsb1'	10	40	91	0.002	49	104	0.007	49	104	0.005	49	104	0.004	49	104	0.006
'sine'	500	NaN	NaN	NaN	38	136	0.016	29	113	0.013	39	133	0.016	37	159	0.018
'sine'	1000	NaN	NaN	NaN	32	105	0.021	38	98	0.020	29	100	0.020	28	106	0.021
'sine'	3000	NaN	NaN	NaN	31	124	0.076	36	126	0.075	39	115	0.071	37	148	0.088
'sine'	4000	NaN	NaN	NaN	33	116	0.093	29	105	0.083	28	117	0.095	33	99	0.082
'raydan2'	500	13	104	0.011	9	51	0.004	9	51	0.004	9	51	0.004	9	51	0.005
'raydan2'	1000	14	109	0.012	15	69	0.009	15	69	0.010	15	69	0.009	15	69	0.008
'raydan2'	5000	13	113	0.057	10	71	0.036	10	71	0.035	11	72	0.037	10	71	0.038
'raydan2'	10000	18	119	0.118	13	66	0.070	13	66	0.066	16	83	0.083	14	67	0.069
'diagonal1'	4	28	85	0.007	20	76	0.003	21	65	0.002	18	74	0.002	20	68	0.002
'diagonal1'	10	35	96	0.004	30	76	0.003	34	85	0.004	33	91	0.005	28	75	0.004
'diagonal2'	500	135	249	0.032	133	243	0.031	136	231	0.027	162	282	0.030	141	242	0.026
'diagonal2'	1000	192	331	0.054	176	310	0.053	215	362	0.065	196	333	0.061	199	365	0.072
'diagonal2'	5000	486	930	0.771	410	682	0.670	429	711	0.601	430	704	0.724	493	833	0.841
'diagonal2'	10000	NaN	NaN	NaN	705	1162	2.030	775	1316	2.140	776	1262	2.043	612	1038	1.661
'diagonal3'	4	24	79	0.007	25	71	0.002	30	69	0.003	30	79	0.003	27	78	0.003
'diagonal3'	10	44	100	0.005	35	92	0.004	40	109	0.004	39	79	0.003	42	106	0.005
'diagonal3'	50	64	123	0.007	59	112	0.006	66	140	0.008	66	124	0.007	65	116	0.007
'singx'	10	138	423	0.027	97	282	0.015	159	455	0.028	146	439	0.023	150	394	0.019
'singx'	100	163	482	0.037	77	304	0.030	159	485	0.041	128	450	0.041	81	255	0.020
'singx'	500	176	495	0.847	119	410	0.682	157	547	0.971	157	513	0.898	145	559	1.015
'singx'	1000	327	1034	6.424	162	516	3.348	224	683	4.269	309	992	6.009	326	1016	6.543
'lin'	10	20	100	0.288	13	79	0.214	16	96	0.291	16	96	0.283	13	79	0.247
'lin'	100	22	105	0.687	27	173	1.117	18	66	0.437	18	65	0.449	21	69	0.457
'lin'	500	19	88	1.062	17	101	1.177	19	110	1.340	18	103	1.353	18	91	1.171
'lin'	1000	25	145	####	16	84	61.003	16	80	60.320	21	85	63.368	19	84	57.936
'pen1'	5	NaN	NaN	NaN	113	519	0.035	109	490	0.032	105	540	0.034	114	526	0.036
'pen1'	10	NaN	NaN	NaN	99	409	0.035	125	560	0.047	123	541	0.051	256	2165	0.176
'pen2'	5	25	104	0.029	22	76	0.010	22	116	0.013	22	81	0.015	24	111	0.013
'pen2'	10	150	561	0.069	356	1230	0.127	251	1047	0.076	341	1314	0.441	434	1653	0.128
'pen2'	100	96	271	0.045	144	356	0.071	111	227	0.058	113	295	0.079	127	344	0.068
'rosex'	50	39	187	0.019	36	153	0.009	47	213	0.010	43	188	0.010	45	182	0.008
'rosex'	100	55	271	0.017	41	156	0.012	49	199	0.015	46	140	0.011	39	151	0.010
'rosex'	500	44	195	0.339	46	194	0.369	44	235	0.351	49	174	0.269	40	161	0.279
'rosex'	1000	45	232	1.547	45	207	1.241	51	196	1.386	50	184	1.016	55	198	1.064

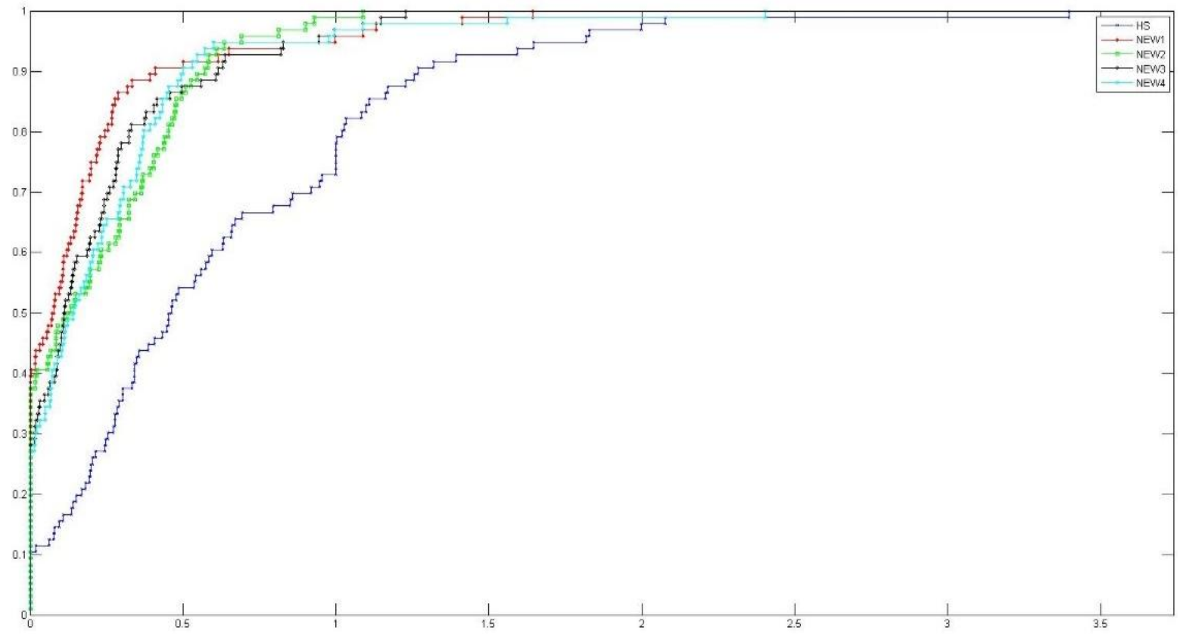

Figure 1. Performance Profile Comparing the Proposed Algorithms and the HS Method in Terms of Number of Iterations

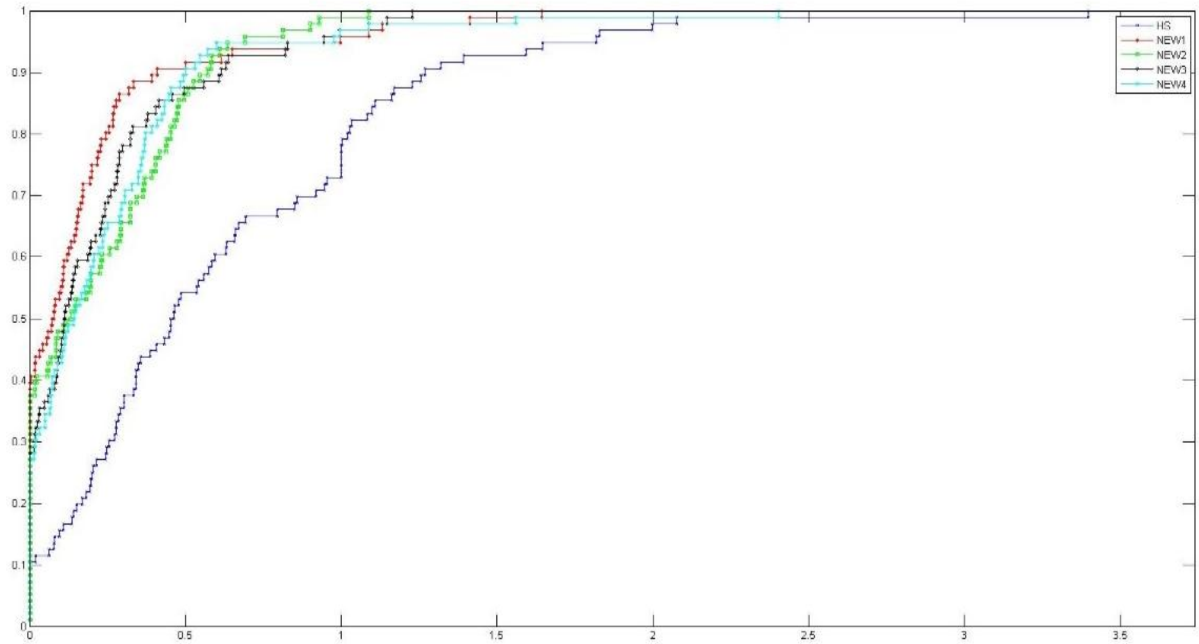

Figure 2. Performance Profile Comparing the Proposed Algorithm and the HS Method in Terms of Number of Function Evaluations

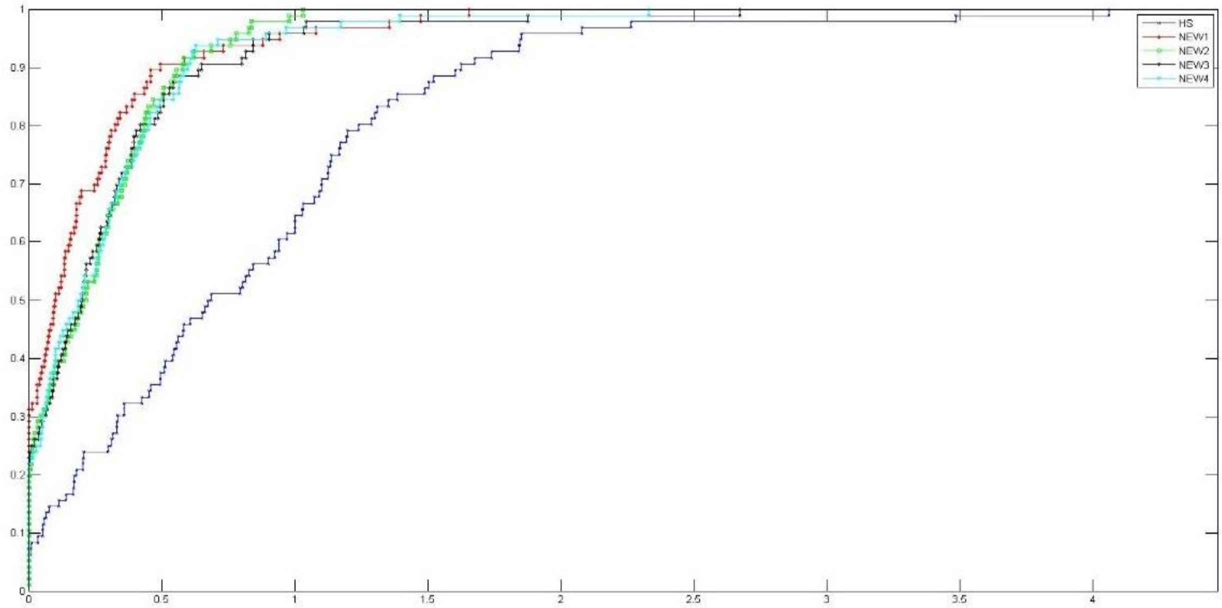

Figure 3. Performance Profile Comparing the Proposed Algorithm and the HS Method in Terms of CPU Time

4. CONCLUSION

In this study, we introduced a new CG method that integrates Newton-type updates to improve convergence properties and computational efficiency. The proposed method satisfies the sufficient descent condition, ensuring that each iteration leads to a reduction in the objective function value, and it demonstrates strong global convergence characteristics. These theoretical guarantees establish the method as a reliable and effective tool for solving unconstrained optimization problems. The numerical results show that the proposed methods (New1, New2, New3, and New4) offer significant improvements over the classical Hestenes-Stiefel (HS) method in terms of key performance metrics: NOI, NOF, and CPUT. Specifically, the proposed methods achieved faster convergence and reduced computational costs, demonstrating their enhanced efficiency. Among the variants, New3 yielded the best performance, with the lowest NOI (26.68), NOF (25.82), and a competitive CPUT (23.51 seconds), highlighting its superior efficiency across a range of test problems. These improvements indicate that the integration of Newton-type updates into the CG framework provides a powerful method for large-scale optimization problems, especially in applications where computational resources are limited and fast convergence is crucial. In conclusion, the newly proposed conjugate gradient method not only satisfies the theoretical convergence conditions but also outperforms existing methods, offering a promising approach for solving complex optimization problems. Future work could focus on further refinement of the method, extending its applicability to constrained optimization problems, and exploring its performance in more specialized domains.

Author Contributions

Basim A. Hassan: Conceptualization, Writing – Review, Formal Analysis, Methodology. Alaa Luqman Ibrahim: Software, Validation, Resources. Thaair A. Ameen: Data Curation, Visualization. Ibrahim Mohammed Sulaiman: Writing - Original Draft, Supervision. All authors discussed the results and contributed to the final manuscript.

Funding Statement

This research received no specific grant from any funding agency in the public, commercial, or not-for-profit sectors.

Acknowledgment

The authors are very grateful to the University of Mosul / College of Computer Sciences and Mathematics for their provided facilities, which helped to improve the quality of this work.

Declarations

The authors declare no competing interests.

REFERENCES

[1] H. Akhadkulov, S. M. Noorani, A. B. Saaban, F. M. Alipiah, and H. Alsamir, "NOTES ON MULTIDIMENSIONAL FIXED-POINT THEOREMS," *Demonstr. Math.*, vol. 50, no. 1, pp. 360–374, Dec. 2017, doi: https://doi.org/10.1515/dema-2017-0033

[2] H. Alsamir *et al.*, "FIXED POINT RESULTS IN METRIC-LIKE SPACES VIA Σ-SIMULATION FUNCTIONS," *Eur. J. Pure Appl. Math.*, vol. 12, pp. 88–100, Jan. 2019. doi: https://doi.org/10.29020/nybg.ejpam.v12i1.3331

[3] C. Luo, L. Wang, Y. Xie, and B. Chen, "A NEW CONJUGATE GRADIENT METHOD FOR MOVING FORCE IDENTIFICATION OF VEHICLE–BRIDGE SYSTEM," *J. Vib. Eng. Technol.*, vol. 12, no. 1, pp. 19–36, Jan. 2024. doi: https://doi.org/10.1007/s42417-022-00824-1

[4] Y. Yuan, D. H. K. Tsang, and V. K. N. Lau, "COMBINING CONJUGATE GRADIENT AND MOMENTUM FOR UNCONSTRAINED STOCHASTIC OPTIMIZATION WITH APPLICATIONS TO MACHINE LEARNING," *IEEE Internet Things J.*, vol. 11, no. 13, pp. 23236–23254, Jul. 2024. doi: https://doi.org/10.1109/JIOT.2024.3376821

[5] S. B. Hanachi, B. Sellami, and M. Belloufi, "A NEW FAMILY OF HYBRID CONJUGATE GRADIENT METHOD FOR UNCONSTRAINED OPTIMIZATION AND ITS APPLICATION TO REGRESSION ANALYSIS," *RAIRO - Oper. Res.*, vol. 58, no. 1, Art. no. 1, Jan. 2024. doi: https://doi.org/10.1051/ro/2023196

[6] B. A. Hassan, I. A. R. Moghrabi, A. L. Ibrahim, and H. N. Jabbar, "IMPROVED CONJUGATE GRADIENT METHODS FOR UNCONSTRAINED MINIMIZATION PROBLEMS AND TRAINING RECURRENT NEURAL NETWORK," *Eng. Rep.*, vol. 7, no. 2, p. e70019, 2025. doi: https://doi.org/10.1002/eng2.70019

[7] D. H. Omar, A. L. Ibrahim, M. M. Hassan, B. G. Fathi, and D. A. Sulaiman, "ENHANCED CONJUGATE GRADIENT METHOD FOR UNCONSTRAINED OPTIMIZATION AND ITS APPLICATION IN NEURAL NETWORKS," *Eur. J. Pure Appl. Math.*, vol. 17, no. 4, pp. 2692–2705, Oct. 2024. doi: https://doi.org/10.29020/nybg.ejpam.v17i4.5354.

[8] B. A. Hassan and H. A. Alashoor, "ON IMAGE RESTORATION PROBLEMS USING NEW CONJUGATE GRADIENT METHODS," Indones. J. Electr. Eng. Comput. Sci., vol. 29, no. 3, p. 1438, Mar. 2023. doi: https://doi.org/10.11591/ijeecs.v29.i3.pp1438-1445

[9] B. A. Hassan and H. Alashoor, "A NEW TYPE COEFFICIENT CONJUGATE ON THE GRADIENT METHODS FOR IMPULSE NOISE REMOVAL IN IMAGES," Eur. J. Pure Appl. Math., vol. 15, no. 4, pp. 2043–2053, Oct. 2022. doi: https://doi.org/10.29020/nybg.ejpam.v15i4.4579

[10] B. A. Hassan and A. A. Saad, "EXPLAINING NEW PARAMETERS CONJUGATE ANALYSIS BASED ON THE QUADRATIC MODEL," *J. Interdiscip. Math.*, vol. 26, no. 6, pp. 1219–1229, 2023. doi: https://doi.org/10.47974/JIM-1620

[11] A. Muhammed Awwal, P. Kumam, K. Sitthithakerngkiet, A. Muhammad, A. Halilu, and S. Ibrahim, "DERIVATIVE-FREE METHOD BASED ON DFP UPDATING FORMULA FOR SOLVING CONVEX CONSTRAINED NONLINEAR MONOTONE EQUATIONS AND APPLICATION," *AIMS Math.*, vol. 6, pp. 8792–8814, Jun. 2021. doi: https://doi.org/10.3934/math.2021510

[12] M. R. Hestenes and E. Stiefel, "METHODS OF CONJUGATE GRADIENTS FOR SOLVING LINEAR SYSTEMS," *J. Res. Natl. Bur. Stand.*, vol. 49, no. 6, p. 409, Dec. 1952. doi: https://doi.org/10.6028/jres.049.044

[13] R. Fletcher and C. M. Reeves, "FUNCTION MINIMIZATION BY CONJUGATE GRADIENTS," *Comput. J.*, vol. 7, no. 2, pp. 149–154, Jan. 1964. doi: https://doi.org/10.1093/comjnl/7.2.149

[14] S. G. Shareef and A. L. Ibrahim, "A NEW CONJUGATE GRADIENT FOR UNCONSTRAINED OPTIMIZATION BASED ON STEP SIZE OF BARZILAI AND BORWEIN," *Sci. J. Univ. Zakho*, vol. 4, no. 1, Art. no. 1, Jun. 2016. doi: https://doi.org/10.25271/2016.4.1.29

[15] B. A. Hassan and A. Ahmed A. Abdullah, "IMPROVEMENT OF CONJUGATE GRADIENT METHODS FOR REMOVING IMPULSE NOISE IMAGES," *Indones. J. Electr. Eng. Comput. Sci.*, vol. 29, no. 1, p. 245, Jan. 2022. doi: https://doi.org/10.11591/ijeecs.v29.i1.pp245-251

[16] J. Sabi'u, S. Ibrahim, P. Kaelo, M. Malik, and S. Kamaruddin, "AN OPTIMAL CHOICE DAI-LIAO CONJUGATE GRADIENT ALGORITHM FOR UNCONSTRAINED OPTIMIZATION AND PORTFOLIO SELECTION," *AIMS Math.*, vol. 9, pp. 642–664, Jan. 2024. doi: https://doi.org/10.3934/math.2024034

[17] J. K. Liu and S. J. Li, "NEW HYBRID CONJUGATE GRADIENT METHOD FOR UNCONSTRAINED OPTIMIZATION," *Appl. Math. Comput.*, vol. 245, pp. 36–43, Oct. 2014. doi: https://doi.org/10.1016/j.amc.2014.07.096

[18] B. A. Hassan, I. A. R. Moghrabi, T. A. Ameen, R. M. Sulaiman, and I. M. Sulaiman, "IMAGE NOISE REDUCTION AND SOLUTION OF UNCONSTRAINED MINIMIZATION PROBLEMS VIA NEW CONJUGATE GRADIENT METHODS," Mathematics, vol. 12, no. 17, Art. no. 17, Jan. 2024. doi: https://doi.org/10.3390/math12172754

[19] Z. Salleh and A. Alhawarat, "AN EFFICIENT MODIFICATION OF THE HESTENES-STIEFEL NONLINEAR CONJUGATE GRADIENT METHOD WITH RESTART PROPERTY," *J. Inequalities Appl.*, vol. 2016, no. 1, p. 110, Apr. 2016. doi: https://doi.org/10.1186/s13660-016-1049-5

[20] G. Yuan, Wei ,Zengxin, and Q. and Zhao, "A MODIFIED POLAK–RIBIÈRE–POLYAK CONJUGATE GRADIENT ALGORITHM FOR LARGE-SCALE OPTIMIZATION PROBLEMS," *IIE Trans.*, vol. 46, no. 4, pp. 397–413, Apr. 2014. doi: https://doi.org/10.1080/0740817X.2012.726757

[21] Z. Wei, G. Li, and L. Qi, "NEW NONLINEAR CONJUGATE GRADIENT FORMULAS FOR LARGE-SCALE UNCONSTRAINED OPTIMIZATION PROBLEMS," *Appl. Math. Comput.*, vol. 179, no. 2, pp. 407–430, Aug. 2006. doi: https://doi.org/10.1016/j.amc.2005.11.150

[22] B. A. Hassan, M. Alnaimy, S. Ibrahim, and A. Abubakar, *AN ALTERNATIVE MODIFIED QUASI-NEWTON EQUATION FOR SOLVING UNCONSTRAINED OPTIMIZATION PROBLEMS*. 2022, p. 205. doi: https://doi.org/10.1109/IT-ELA57378.2022.10107966

[23] P. Wolfe, "CONVERGENCE CONDITIONS FOR ASCENT METHODS. II: SOME CORRECTIONS," *SIAM Rev.*, vol. 13, no. 2, pp. 185–188, Apr. 1971. doi: https://doi.org/10.1137/1013035

[24] P. Wolfe, "CONVERGENCE CONDITIONS FOR ASCENT METHODS," *SIAM Rev.*, Jul. 2006. doi: https://doi.org/10.1137/1011036

[25] L. Nazareth, "A CONJUGATE DIRECTION ALGORITHM WITHOUT LINE SEARCHES," *J. Optim. Theory Appl.*, vol. 23, no. 3, pp. 373–387, Nov. 1977. doi: https://doi.org/10.1007/BF00933447

[26] B. A. Hassan and A. A. Saad, "ELASTIC CONJUGATE GRADIENT METHODS TO SOLVE ITERATION PROBLEMS," *J. Interdiscip. Math.*, vol. 26, no. 6, pp. 1207–1217, 2023. doi: https://doi.org/10.47974/JIM-1619

[27] N. I. M. Gould, D. Orban, and P. L. Toint, "CUTER AND SIFDEC: A CONSTRAINED AND UNCONSTRAINED TESTING ENVIRONMENT, REVISITED," *ACM Trans Math Softw*, vol. 29, no. 4, pp. 373–394, Dec. 2003, doi: https://doi.org/10.1145/962437.962439

[28] N. Andrei, "AN UNCONSTRAINED OPTIMIZATION TEST FUNCTIONS COLLECTION," 2008. Accessed: Feb. 26, 2025. [Online]. Available: https://www.semanticscholar.org/paper/An-Unconstrained-Optimization-Test-Functions-Andrei/3bcfd42ec838188f3366d75d3568c0638a79c8c4

[29] J. J. Moré, B. S. Garbow, and K. E. Hillstrom, "TESTING UNCONSTRAINED OPTIMIZATION SOFTWARE," *ACM Trans Math Softw*, vol. 7, no. 1, pp. 17–41, Mar. 1981. doi: https://doi.org/10.1145/355934.355936

[30] E. D. Dolan and J. J. Moré, "BENCHMARKING OPTIMIZATION SOFTWARE WITH PERFORMANCE PROFILES," *Math. Program.*, vol. 91, no. 2, pp. 201–213, Jan. 2002. doi: https://doi.org/10.1007/s101070100263

[31] B. A. Hassan and H. M. Sadiq, "A NEW FORMULA ON THE CONJUGATE GRADIENT METHOD FOR REMOVING IMPULSE NOISE IMAGES", Bulletin of the South Ural State University. Ser. Mathematical Modelling, Programming & Computer Software (Bulletin SUSU MMCS), 2022, vol. 15, no. 4, pp. 123-130. doi: https://doi.org/10.14529/mmp220412

[32] B. A. Hassan and H. Sadiq, "Efficient new conjugate gradient methods for removing impulse noise images". European Journal of Pure and Applied Mathematics, 15(4), 2011–2021. doi: https://doi.org/10.29020/nybg.ejpam.v15i4.4568