Iranian Journal of Numerical Analysis and Optimization
Vol. 16, No. 1, 2026, pp 102–121
https://doi.org/10.22067/ijnao.2025.94956.1709
https://ijnao.um.ac.ir/

Research Article

A novel three-term conjugate gradient approach for deep neural network training

F.S. Hosseini-Meighan and A. Ashrafi*

Abstract

This paper presents a new three-term conjugate gradient (CG) method for large-scale unconstrained optimization with applications to training deep neural networks. The approach employs enhanced CG parameter strategies together with a strong Wolfe line search to guarantee sufficient descent and global convergence under standard smoothness assumptions. Extensive numerical experiments are conducted on both the standard CUTEr benchmark test problems and the MNIST digit classification task. The proposed MPRP-IFR hybrid algorithm demonstrates superior efficiency compared to traditional CG variants, achieving faster convergence and reduced computational cost on CUTEr problems as well as high classification accuracy and stable training dynamics for deep learning models. These findings highlight the robustness and effectiveness of the proposed CG scheme for large-scale optimization.

AMS subject classifications (2020): Primary 45D05; Secondary 42C10, 65G99.

*Corresponding author

Received 16 August 2025; revised 28 October 2025; accepted 19 November 2025

Faeze Sadat Hosseini-Meighan
Department of Mathematics, Faculty of Mathematics,Statistics and Computer Science, Semnan University, P.O. Box 35131-19111, Semnan, Iran. e-mail: faezehosseini.mighan@semnan.ac.ir

Ali Ashrafi
Department of Mathematics, Faculty of Mathematics,Statistics and Computer Science, Semnan University, P.O. Box 35131-19111, Semnan, Iran. e-mail: a_ashrafi@semnan.ac.ir

How to cite this article
Hosseini-Meighan, F.S. and Ashrafi, A., A novel three-term conjugate gradient approach for deep neural network training. *Iran. J. Numer. Anal. Optim.*, 2026; 16(1): 102-121. https://doi.org/10.22067/ijnao.2025.94956.1709

Keywords: Conjugate gradient method; Unconstrained optimization; line search; Global convergence; Deep neural network.

1 Introduction

The conjugate gradient (CG) method is widely regarded as a powerful tool for solving large-scale unconstrained optimization problems, primarily due to its low memory requirements and strong convergence properties. In particular, the CG method has been extensively utilized in practical fields such as signal processing [12], machine learning [21], and image processing [9]. In the general unconstrained optimization problem, the goal is to minimize a continuously differentiable function $f : \mathbb{R}^n \to \mathbb{R}$:

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

where $x \in \mathbb{R}^n$ denotes the vector of optimization variables. Iterative methods, such as the CG algorithms, generate a sequence of iterates $\{x_k\}$ designed to converge to a local (and, under convexity, global) minimum of f.

At each iteration k, the CG method constructs a search direction d_k and performs an update of the form:

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

where $\alpha_k > 0$ is the step size, and d_k is the search direction, recursively defined as

$$d_k = \begin{cases} -g_k, & k = 1, \\ -g_k + \beta_k d_{k-1}, & k \geq 2, \end{cases} \tag{3}$$

with the gradient $g_k = \nabla f(x_k)$. The efficiency of the CG method is intimately tied to the appropriate choice of the conjugacy parameter β_k.

In order to ensure sufficient progress towards the minimum, the step size α_k is typically computed using a line search procedure. Among these, the Wolfe conditions and the strong Wolfe conditions are standard choices due to their critical role in guaranteeing desirable convergence behavior. The Wolfe conditions require that the step size α_k satisfies

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

where $0 < \delta < \sigma < 1$. The strong Wolfe conditions strengthen the second inequality as follows:

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

These line search conditions play a crucial role in convergence theory for CG methods.

The evolution of the CG method has given rise to several classical formulas for the parameter β_k, each imbued with unique theoretical and practical properties. Notable among these are

$$\beta_k^{PRP} = \frac{g_k^T(g_k - g_{k-1})}{\|g_{k-1}\|^2}, \quad \text{(Polak–Ribière–Polyak)} \tag{6}$$

$$\beta_k^{FR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2}, \quad \text{(Fletcher–Reeves)} \tag{7}$$

$$\beta_k^{HS} = \frac{g_k^T(g_k - g_{k-1})}{d_{k-1}^T(g_k - g_{k-1})}, \quad \text{(Hestenes–Stiefel)} \tag{8}$$

$$\beta_k^{DY} = \frac{\|g_k\|^2}{d_{k-1}^T(g_k - g_{k-1})} \quad \text{(Dai–Yuan).} \tag{9}$$

The FR and DY methods are celebrated for their strong theoretical convergence, particularly when paired with the strong Wolfe line search [5, 3]. However, they can suffer from practical issues, such as the jamming phenomenon, which slows convergence. By contrast, the PRP and HS methods tend to perform better numerically but may lack global convergence guarantees [20, 7].

To overcome these challenges, several improved and modified parameters have been proposed. Jiang and Jian [8] advanced the field with the improved Fletcher–Reeves (IFR) parameter, which leverages the second Wolfe inequality. The IFR parameter is defined as

$$\beta_k^{IFR} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2} \cdot \frac{|g_k^T d_{k-1}|}{-g_{k-1}^T d_{k-1}}, \tag{10}$$

enabling stronger global convergence and helping the algorithm to avoid stalling.

Similarly, Ma et al. [13] introduced the modified Polak–Ribière–Polyak (MPRP) parameter. Building on the WYL formula,

$$\beta_k^{WYL} = \frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} g_k^T g_{k-1}}{\|g_{k-1}\|^2}, \tag{11}$$

they formulated the MPRP parameter as

$$\beta_k^{MPRP} = \beta_k^{WYL} \cdot \frac{|g_k^T d_{k-1}|}{-g_{k-1}^T d_{k-1}}. \tag{12}$$

These modifications are designed to ensure that the search direction maintains sufficient descent and robust convergence properties under the strong Wolfe line search conditions.

Recent advancements in CG methods have explored extensions beyond the traditional two-term update. One such direction is the three-term CG formulation, as proposed by Dada et al. [2], where the search direction at each iteration is enhanced by incorporating an additional term involving the difference of gradients. Specifically, the search direction d_k is defined as

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k d_{k-1} + \theta_k y_{k-1}, & k > 0, \end{cases} \tag{13}$$

where $y_{k-1} = g_k - g_{k-1}$, $\beta_k = \frac{g_k^T y_{k-1}}{\|g_{k-1}\|^2}$, and $\theta_k = \frac{g_k^T d_{k-1}}{\|d_{k-1}\|^2}$.

Building upon this, Dada et al. [2] further proposed a hybrid parameterization, named as CHM, by forming a convex combination of the classical Polak–Ribière–Polyak (PRP) and Fletcher–Reeves (FR) parameters for β_k. The combined parameter is given by

$$\beta_k^{CHM} = (1 - \varphi)\, \beta_k^{PRP} + \varphi\, \beta_k^{FR}, \qquad 0 < \varphi < 1, \tag{14}$$

where β_k^{PRP} and β_k^{FR} are defined in (6) and (7), respectively.

With this hybrid approach, the new search direction can be compactly expressed as

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k^{CHM}\, d_{k-1} + \theta_k y_{k-1}, & k > 0. \end{cases} \tag{15}$$

This augmented structure provides additional flexibility and has demonstrated improved numerical performance in applications such as deep learning optimization.

Recent improvements in CG-type methods include various hybrid and adaptive formulations [15, 16, 17, 18], which have shown promising results in numerical optimization and related applications.

Inspired by these advances, the present work proposes a novel three-term CG method that, for the first time, integrates the IFR and MPRP parameters in an analogous convex combination. The goal is to capitalize on their respective strengths to achieve even stronger convergence and robustness for large-scale unconstrained optimization and deep learning applications.

The remainder of this paper is organized as follows. Section 2 presents the proposed three-term CG algorithm based on a convex combination of IFR and MPRP parameters. Section 3 provides a detailed convergence analysis of the method, including sufficient descent and global convergence. Section 4 reports numerical results on training deep neural networks for the MNIST benchmark. Finally, section 5 concludes the paper and outlines possible future research directions.

2 Proposed three-term CG method

Inspired by the three-term structure in [2], we propose a new CG algorithm, combining the recently developed IFR parameter [8] and the MPRP parameter [13]. The search direction at iteration k is defined as

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k^{\text{MPRP-IFR}}\, d_{k-1} + \theta_k y_{k-1}, & k \geq 1, \end{cases} \tag{16}$$

where the coefficients are given by

$$\beta_k^{\text{MPRP-IFR}} = (1-\varphi)\,\beta_k^{\text{MPRP}} + \varphi\,\beta_k^{\text{IFR}}, \tag{17}$$

$$\theta_k = -\frac{g_k^T d_{k-1}}{\|d_{k-1}\|^2}, \tag{18}$$

with $0 < \varphi < 1$. In all experiments, the convex-combination parameter was fixed at $\varphi = 0.5$, which yielded a balanced contribution between the MPRP and IFR components and showed stable performance across test problems. Here, β_k^{MPRP} and β_k^{IFR} are defined, respectively, by [13, 8]

$$\beta_k^{\text{MPRP}} = \left[\frac{\|g_k\|^2 - \frac{\|g_k\|}{\|g_{k-1}\|} g_k^\top g_{k-1}}{\|g_{k-1}\|^2}\right] \cdot \frac{|g_k^\top d_{k-1}|}{-g_{k-1}^\top d_{k-1}}, \tag{19}$$

$$\beta_k^{\text{IFR}} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2} \cdot \frac{|g_k^\top d_{k-1}|}{-g_{k-1}^\top d_{k-1}}. \tag{20}$$

The developed algorithm is described below.

This formulation generalizes the CHM three-term approach, allowing the algorithm to benefit from the improved convergence and numerical performance observed in both the IFR and MPRP schemes.

3 Convergence analysis

In this section, we rigorously analyze the convergence properties of the proposed three-term CG method, whose search direction is given by

Algorithm 1 Proposed three-term CG algorithm (MPRP-IFR Hybrid)

Require: Initial point $x_0 \in \mathbb{R}^n$, tolerance $\varepsilon > 0$, parameter $0 < \varphi < 1$

$g_0 = \nabla f(x_0)$
$d_0 = -g_0$
$k \leftarrow 0$
while $\|g_k\| > \varepsilon$ **do**
    Find α_k by (strong) Wolfe conditions
    $x_{k+1} = x_k + \alpha_k d_k$
    $g_{k+1} = \nabla f(x_{k+1})$
    if $k = 0$ **then**
        $d_{k+1} = -g_{k+1}$
    else if $k = 1$ **then**
        Compute β_1^{New}
        $d_2 = -g_2 + \beta_1^{\text{New}} d_1$
    else
        Compute $\beta_k^{\text{MPRP-IFR}}, \theta_k$ (see equations (19)–(20))
        $d_{k+1} = -g_{k+1} + \beta_k^{\text{MPRP-IFR}}\, d_k + \theta_k y_{k-1}$
    end if
    $k \leftarrow k + 1$
end while

$$d_k = \begin{cases} -g_k, & k = 0, \\ -g_k + \beta_k^{\text{MPRP-IFR}}\, d_{k-1} + \theta_k y_{k-1}, & k \geq 1, \end{cases}$$

where

$$\beta_k^{\text{MPRP-IFR}} = (1-\varphi)\,\beta_k^{\text{MPRP}} + \varphi\,\beta_k^{\text{IFR}}, \qquad 0 < \varphi < 1,$$

and $\theta_k = -\frac{g_k^T d_{k-1}}{\|d_{k-1}\|^2}$, $y_{k-1} = g_k - g_{k-1}$. The parameters β_k^{MPRP} and β_k^{IFR} are as defined in (19)–(20).

3.1 Assumptions

We make the following standard assumptions on the objective function f:

A1 The level set $\Omega = \{x \in \mathbb{R}^n \mid f(x) \leq f(x_0)\}$ is bounded, where x_0 is the initial point.

A2 In a neighborhood U of Ω, the function f is continuously differentiable and the gradient $g(x)$ is Lipschitz continuous. That is, there exists $L > 0$ such that

$$\|g(x) - g(y)\| \leq L\|x - y\|, \quad \forall x, y \in U.$$

Under these assumptions, there exists $\gamma > 0$ such that $\|g(x)\| \leq \gamma$ for all $x \in \Omega$.

3.2 Sufficient descent property

We establish that the search directions generated by our method are sufficiently descending.

Theorem 1 (Sufficient descent)**.** Let $\{x_k\}$ and $\{d_k\}$ be sequences generated by the proposed algorithm, where the step size α_k satisfies the strong Wolfe condition (5) with $0 < \delta < \sigma < \frac{1}{2}$. Then there exists a constant $c > 0$ such that

$$g_k^T d_k \leq -c\|g_k\|^2, \qquad \forall k \geq 0. \tag{21}$$

Proof. We proceed by induction.

Base case ($k = 0$): We have $d_0 = -g_0$, so $g_0^T d_0 = -\|g_0\|^2$. Thus, the claim holds with $c = 1$.

Inductive step ($k \geq 1$): By the definition of the search direction,

$$d_k = -g_k + \beta_k^{\text{MPRP-IFR}}\, d_{k-1} + \theta_k y_{k-1}.$$

Taking the inner product with g_k, we have

$$\begin{aligned} g_k^T d_k &= -\|g_k\|^2 + \beta_k^{\text{MPRP-IFR}}\ (g_k^T d_{k-1}) + \theta_k\, (g_k^T y_{k-1}) \\ &= -\|g_k\|^2 + \beta_k^{\text{MPRP-IFR}}\ (g_k^T d_{k-1}) + \theta_k\, (\|g_k\|^2 - g_k^T g_{k-1}). \end{aligned} \tag{22}$$

Recall that $\theta_k = -\frac{g_k^T d_{k-1}}{\|d_{k-1}\|^2}$, so we can substitute:

$$\theta_k (g_k^T y_{k-1}) = -\frac{g_k^T d_{k-1}}{\|d_{k-1}\|^2}\left(\|g_k\|^2 - g_k^T g_{k-1}\right)$$

This term acts to further control the projected component along y_{k-1} and, under the line search, does not increase $g_k^T d_k$.

From the strong Wolfe condition (5), we have

$$|g_k^T d_{k-1}| \leq \sigma |g_{k-1}^T d_{k-1}|, \quad \text{and since } g_{k-1}^T d_{k-1} < 0 \implies |g_k^T d_{k-1}| \leq -\sigma g_{k-1}^T d_{k-1}.$$

The definition of $\beta_k^{\text{MPRP-IFR}}$ as a convex combination of nonnegative β_k^{MPRP} and β_k^{IFR} ensures it is nonnegative and upper bounded as in [13, 8].

Combining these properties and using arguments similar to [2, Lemma 3], one can show that the cumulative effect of the last two terms in (22) does not exceed a (small) fraction of the first negative term. That is, there exists a constant $c > 0$ (dependent on σ) such that

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

which completes the induction. □

Lemma 1. Under the strong Wolfe line search with $\sigma < \frac{1}{2}$, the parameters β_k^{MPRP} and β_k^{IFR} are nonnegative at every iteration.

The nonnegativity of $\beta_k^{\text{MPRP-IFR}}$ guarantees that the search direction maintains sufficient descent, which is crucial for global convergence.

3.3 Global convergence

We conclude this section by establishing the global convergence of the algorithm.

Lemma 2 (Zoutendijk Condition, [19])**.** Suppose assumption **A1** is satisfied. If d_k is a descent direction at each iteration, and α_k satisfies the Wolfe conditions, then

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

Lemma 3 (Uniform boundedness of β_k^{MPRP}, β_k^{IFR}, and θ_k)**.** Suppose that assumptions **A1**–**A2** hold and that the step size α_k satisfies the strong Wolfe line search with $0 < \sigma < \frac{1}{2}$. Then, the parameters β_k^{MPRP}, β_k^{IFR}, and θ_k are uniformly bounded for all $k \geq 0$. That is, there exist constants $C_\beta, C_\theta > 0$ such that

$$0 \leq \beta_k^{\text{MPRP}},\ \beta_k^{\text{IFR}} \leq C_\beta, \qquad |\theta_k| \leq C_\theta.$$

Proof. For β_k^{IFR}, as defined in (20), we have

$$\beta_k^{\text{IFR}} = \frac{\|g_k\|^2}{\|g_{k-1}\|^2} \cdot \frac{|g_k^T d_{k-1}|}{-g_{k-1}^T d_{k-1}}.$$

From [8, Theorem 1], under the strong Wolfe condition with $0 < \sigma < 1/2$, there exists $0 < C_0 < 1$ such that

$$\frac{|g_k^T d_{k-1}|}{-g_{k-1}^T d_{k-1}} \leq \sigma < \frac{1}{2}.$$

Combined with the Cauchy–Schwarz inequality and the boundedness of gradients on the level set Ω, both $\|g_k\|$ and $\|g_{k-1}\|$ are bounded above and below (away from zero unless convergence occurs). Therefore, there exists $C_\beta > 0$ such that $0 \leq \beta_k^{\text{IFR}} \leq C_\beta$ for all k.

Similarly, for β_k^{MPRP}, following [13, Theorem 2.1] and the above inequalities,

$$\beta_k^{\text{MPRP}} \leq 2\sigma\beta_k^{\text{FR}} \leq 2\sigma\frac{\|g_k\|^2}{\|g_{k-1}\|^2} \leq C_{\beta'},$$

where $C_{\beta'} > 0$ denotes a constant that bounds the ratio $|\beta'_{k+1}|/|\beta_k|$ in the convergence analysis.

For $\theta_k = -\frac{g_k^T d_{k-1}}{\|d_{k-1}\|^2}$, using the strong Wolfe inequality, the Cauchy–Schwarz inequality, and the boundedness of search directions (from the descent lemma and level set), both the numerator and denominator are bounded, so $|\theta_k| \leq C_\theta$ for some $C_\theta > 0$.

This completes the proof. □

Theorem 2 (Global Convergence)**.** Suppose assumptions **A1** and **A2** hold. Let $\{x_k\}$ be generated by the proposed algorithm, with step sizes α_k determined by the strong Wolfe conditions. Then,

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

Proof. Suppose, for contradiction, that $\|g_k\| \geq \varepsilon > 0$ for all k. Using Theorem 1, $(g_k^T d_k)^2 \geq c^2\|g_k\|^4 \geq c^2\varepsilon^4$. From Lemma 2,

$$\sum_{k=0}^{\infty} \frac{c^2\varepsilon^4}{\|d_k\|^2} < \infty \implies \sum_{k=0}^{\infty} \frac{1}{\|d_k\|^2} < \infty,$$

which forces $\|d_k\| \to \infty$. However, with bounded and nonnegative $\beta_k^{\text{MPRP-IFR}}$ and θ_k, as in [13, 2], the direction norm cannot diverge. Therefore, this leads to a contradiction, and the proof is complete. □

Remark 1. The above convergence theory demonstrates that the proposed three-term method avoids jamming and is especially well-suited for numerically demanding problems such as image restoration and deep neural network optimization, where robust global convergence is vital.

4 Numerical experiments

In this section, we evaluate the numerical efficiency of the proposed MPRP-IFR hybrid three-term method by comparing it with the IFR and MPRP algorithms on the standard CUTEr test

problems. All numerical experiments were implemented in MATLAB R2023a on a machine equipped with an Intel(R) Core(TM) i7 CPU at 1.8 GHz with 8 GB RAM under Windows 10 (CentOS 7 via VMware). The strong Wolfe line search was employed with parameters $\delta = 0.1$ and $\sigma = 0.9$. Each algorithm was terminated whenever $k > 10000$ or $\|\mathbf{g}_k\|_\infty \leq 10^{-6}(1 + |f_k|)$. The comparison was carried out using the Dolan–Moré performance profile [4] with total number of function evaluations (TNF), total number of iterations (NI), and gradient evaluations (NG) as the performance metrics.

4.1 Normal unconstrained optimization problems

The test set consists of 85 unconstrained problems drawn from the CUTEr collection, with dimensions up to 10^4. In each table, IFR serves as the baseline reference method, and all algorithms employ identical stopping and line-search parameters.

Tables 1 and 2 summarize the results for the strong Wolfe line search, representing CPU times and iteration counts, respectively. Figures 1–3 illustrate their Dolan–Moré performance profiles.

Table 1: Complete CUTEr Q-test results (TNF and CPUT for all 85 functions).

Function	n	MPRP-IFR (Proposed)		IFR (Baseline)		MPRP	
		TNF	CPUT	TNF	CPUT	TNF	CPUT
ARGLINA	200	9	5.3827E−02	9	5.7221E−02	9	5.0604E−02
ARWHEAD	5000	12671	3.3013E+00	175	1.5566E−01	8123	1.9235E+00
BDEXP	5000	34	5.1409E−02	34	5.2206E−02	39	5.8209E−02
BDQRTIC	5000	50015	1.4010E+01	4405	1.2209E+00	50004	1.3656E+01
BIGGSB1	5000	69998	1.2708E+01	17498	2.0895E+00	69998	1.0467E+01
BOX	10000	50004	2.8386E+01	178	3.4721E−01	47811	2.6514E+01
BRYBND	5000	327	1.7791E−01	429	2.0810E−01	325	1.6957E−01
CHAINWOO	4000	50068	1.1912E+01	2462	5.8507E−01	50008	1.0955E+01
COSINE	10000	133	1.5701E−01	123	1.4114E−01	128	1.4258E−01
DQDRTIC	5000	13746	2.6111E+00	33	7.1239E−02	13746	2.3886E+00
DQRTIC	5000	150	6.8116E−02	94	5.4275E−02	89	4.5756E−02
EG2	1000	29	1.8019E−02	29	1.2988E−02	29	2.1072E−02
ENGVAL1	5000	231	1.2652E−01	94	7.7729E−02	242	1.0676E−01
EXTROSNB	1000	50038	4.1281E+00	50039	3.2200E+00	50026	5.4974E+01
FLETCBV2	5000	50004	2.8883E+01	25896	1.0677E+01	50004	2.0746E+01
GENROSE	500	50008	3.1270E+00	5714	3.4305E−01	50008	2.9078E+00
LIARWHD	5000	50004	1.3686E+01	336	1.4619E−01	50004	1.0454E+01
QUARTC	5000	150	6.7990E−02	94	6.1088E−02	89	5.4241E−02
SPMSRTLS	4999	24195	9.5517E+00	1668	6.6947E−01	37085	1.0085E+01

SROSENBR	5000	27986	4.8097E+00	158	7.0639E−02	51324	7.7766E+00
VARDIM	200	54	1.0090E−02	69	1.0394E−02	61	5.3430E−03
WOODS	4000	57598	1.7727E+01	883	1.9311E−01	57940	8.1926E+00
CHNROSNB	50	53339	2.4892E+00	1980	8.9791E−02	52660	2.2958E+00
CURLY10	10000	50050	2.1085E+01	50050	2.4163E+01	50050	1.7981E+01
CURLY20	10000	50046	3.5705E+01	50042	2.6201E+01	50046	2.5041E+01
CURLY30	10000	50046	3.2801E+01	50042	3.3074E+01	50046	3.2658E+01
DECONVU	63	50004	2.5817E+00	12394	5.1980E−01	50004	2.4377E+00
DIXON3DQ	10000	69998	1.3992E+01	69998	1.2856E+01	69998	1.2772E+01
EIGENALS	2550	50031	8.6863E+01	50031	8.5898E+01	50031	8.6717E+01
EIGENBLS	2550	50004	8.6806E+01	50004	9.0052E+01	50004	9.1149E+01
EIGENCLS	2652	50012	9.4289E+01	54795	1.0869E+02	50012	9.3952E+01
FMINSRF2	5625	50004	1.3946E+01	3592	1.1460E+00	50004	1.3260E+01
FMINSURF	5625	55284	1.6502E+01	3343	1.0942E+00	55294	1.5061E+01
FREUROTH	5000	41704	1.3731E+01	340	1.8211E−01	31287	8.5557E+00
GENHUMPS	5000	50227	3.1943E+01	34371	1.9762E+01	50058	3.0465E+01
MANCINO	100	183482	2.0248E+02	166437	1.8188E+02	151953	1.6582E+02
MOREBV	5000	69998	1.2937E+01	19731	3.4598E+00	69998	1.1410E+01
MSQRTALS	1024	50023	1.7447E+01	29451	1.0477E+01	50023	1.7351E+01
MSQRTBLS	1024	50023	1.7335E+01	20290	7.1822E+00	50023	1.7037E+01
NONCVXU2	5000	50004	2.1021E+01	23864	1.0392E+01	50004	1.9182E+01
NONDQUAR	5000	50043	1.0098E+01	50004	7.8471E+00	50023	7.6930E+00
POWELLSG	5000	57852	1.2204E+01	1837	3.3150E−01	52568	8.3233E+00
POWER	10000	50004	9.9540E+01	2644	7.1891E−01	50004	6.4910E+01
SENSORS	100	273	4.3467E−01	204	3.3546E−01	224	3.5934E−01
SINQUAD	5000	50038	2.5225E+01	454	3.2291E−01	50049	2.0159E+01
SPARSINE	5000	69112	3.4424E+01	69548	3.1401E+01	68142	3.0424E+01
TESTQUAD	5000	69998	1.6792E+01	13529	1.5509E+00	69998	1.5619E+01
TOINTGOR	50	5199	2.4317E−01	567	3.0382E−02	7949	3.4186E−01
TOINTQOR	50	1251	5.4415E−02	180	1.4739E−02	1251	4.7619E−02
TOINTPSP	50	5887	2.5127E−01	952	4.7670E−02	5792	2.2977E−01
TQUARTIC	5000	50004	1.0653E+01	237	1.3798E−01	50004	8.8288E+00
BQPGABIM	50	3316	1.3187E−01	187	9.2090E−03	3316	1.2454E−01
BQPGASIM	50	3316	1.3142E−01	187	7.8890E−03	3316	1.1624E−01
BQPGAUSS	2003	69998	7.7101E+00	5463	1.6310E+00	69998	7.5697E+00
BRATU1D	5003	994	7.3012E+00	645	7.0182E+00	994	7.3740E+00
CHENHARK	5000	69998	1.0262E+01	22307	2.6998E+00	69998	8.4332E+00
CLPLATEB	5041	50004	1.6376E+01	50004	1.7070E+01	50004	1.4058E+01
DMN15102	66	50039	1.0514E+02	50084	1.0535E+02	50016	1.0461E+02
DMN15103	99	50119	1.2923E+02	50071	1.2890E+02	50099	1.2877E+02
DMN37142	66	50008	1.0456E+02	50012	1.0416E+02	50008	1.0481E+02
DMN37143	99	50008	1.2756E+02	50327	1.2796E+02	50035	1.2733E+02
DRCAV1LQ	4489	4	6.7066E−02	4	6.4555E−02	4	6.8584E−02
DRCAV2LQ	4489	4	6.6654E−02	4	7.0272E−02	4	6.6335E−02
DRCAV3LQ	4489	4	6.7001E−02	4	6.8761E−02	4	6.6303E−02

FLETCBV3	5000	49	8.4340E−02	49	8.1860E−02	49	8.7019E−02
FLETCHBV	5000	45	1.0883E−01	45	1.1035E−01	49	7.9924E−02
GENHUMPS	5000	50227	3.2422E+01	34371	2.0054E+01	50058	2.9388E+01

The computational cost of the proposed MPRP-IFR Hybrid method is comparable to other CG schemes, since its three-term update only involves one additional inner product per iteration. As shown by the CPUT results and performance profiles, the method achieves faster convergence without increasing memory consumption, maintaining $\mathcal{O}(n)$ storage complexity similar to classical CG variants.

Table 2: Comparison of NI and NG for all test functions.

Function	n	MPRP-IFR (Proposed)		IFR (Baseline)		MPRP	
		NI	NG	NI	NG	NI	NG
ARGLINA	200	1	2	1	2	1	2
ARWHEAD	5000	1942	1948	1472	1540	1612	1630
BDEXP	5000	6	7	7	8	7	8
BDQRTIC	5000	10000	10004	10000	10001	10000	10001
BIGGSB1	5000	10000	19998	10000	19998	10000	19998
BOX	10000	10000	10001	9206	9226	9528	9577
BRYBND	5000	59	82	64	79	56	74
CHAINWOO	4000	10000	10014	10000	10005	10000	10002
COSINE	10000	9	29	10	28	10	30
DQDRTIC	5000	1964	3926	1964	3926	1964	3926
DQRTIC	5000	22	23	18	19	17	18
EG2	1000	5	6	5	6	5	6
ENGVAL1	5000	45	51	42	49	43	52
EXTROSNB	1000	10000	10005	10000	10008	10000	10007
FLETCBV2	5000	10000	10001	4926	9226	5514	9577
GENROSE	500	10000	10002	1131	1152	1152	1152
LIARWHD	5000	10000	10001	62	74	74	74
QUARTC	5000	22	23	18	19	17	18
SPMSRTLS	4999	4324	5650	5539	7238	6564	8691
SROSENBR	5000	4900	6775	10000	10399	10000	10661
VARDIM	200	13	15	11	12	11	12
WOODS	4000	7781	11166	10000	13293	10000	13967
CHNROSNB	50	10000	50130	10000	53612	10000	326059
CURLY10	10000	10000	11653	10000	11349	10000	11329
CURLY20	10000	651	651	546	546	821	821
CURLY30	10000	10013	10013	10012	10012	10012	10012
DECONVU	63	10001	10001	10001	10001	10001	10001
DIXON3DQ	10000	19998	19998	19998	19998	19998	19998

EIGENALS	2550	47	56	42	51	40	48
EIGENBLS	2550	10008	10008	10008	10008	10008	10008
EIGENCLS	2652	10001	10001	10001	10001	10001	10001
FMINSRF2	5625	10000	12618	10000	12639	10000	12646
FMINSURF	5625	7457	8930	5862	7313	5678	7122
FREUROTH	5000	10000	10029	10000	10037	10000	10015
GENHUMPS	5000	10000	42267	10000	39457	10000	41977
MANCINO	100	36446	42267	29766	39457	26022	41977
MOREBV	5000	10000	19998	2819	19998	10000	19998
MSQRTALS	1024	10000	10006	5002	10006	10000	10006
MSQRTBLS	1024	10000	10006	3503	10006	10000	10006
NONCVXU2	5000	10000	10001	4772	10001	4773	10001
NONDQUAR	5000	10000	10007	10000	10006	10000	10006
POWELLSG	5000	14013	14013	11674	11674	11283	11283
POWER	10000	10000	10001	528	10001	529	10001
SENSORS	100	46	46	30	51	44	50
SINQUAD	5000	10012	10012	10012	10013	10013	10013
SPARSINE	5000	19079	18429	19531	18639	19069	19633
TESTQUAD	5000	19998	19998	1933	19998	1998	19998
TOINTGOR	50	888	1159	1367	1806	1357	1793
TOINTQOR	50	179	356	179	356	179	356
TOINTPSP	50	955	1366	939	1325	946	1352
TQUARTIC	5000	10000	10001	10000	10001	10000	10001
BQPGABIM	50	474	946	474	946	474	946
BQPGASIM	50	474	946	474	946	474	946
BQPGAUSS	2003	10000	19998	10000	19998	10000	19998
BRATU1D	5003	75	266	75	266	75	266
CHENHARK	5000	10000	19998	10000	19998	10000	19998
CLPLATEB	5041	10000	10001	10000	10001	10000	10001
DMN15102	66	10000	10007	10000	10005	10000	10004
DMN15103	99	10000	10018	10000	10022	10000	10025
DMN37142	66	10000	10002	10000	10002	10000	10002
DMN37143	99	10000	10002	10000	10011	10000	10009
DRCAV1LQ	4489	0	1	0	1	0	1
DRCAV2LQ	4489	0	1	0	1	0	1
DRCAV3LQ	4489	0	1	0	1	0	1
FLETCBV3	5000	2	12	2	11	2	12
FLETCHBV	5000	2	11	2	11	2	12
GENHUMPS	5000	10000	10029	10000	10037	10000	10015

The results reveal that the proposed MPRP-IFR algorithm performs favorably, often requiring fewer function evaluations and exhibiting reduced or comparable CPU times relative to IFR

and MPRP. Particularly for test functions such as `BOX`, `FREUROTH`, and `GENHUMPS`, the proposed scheme attains the optimal solution with smaller TNF and nearly identical CPU cost, confirming that the third-term hybridization enhances descent quality without additional computational burden.

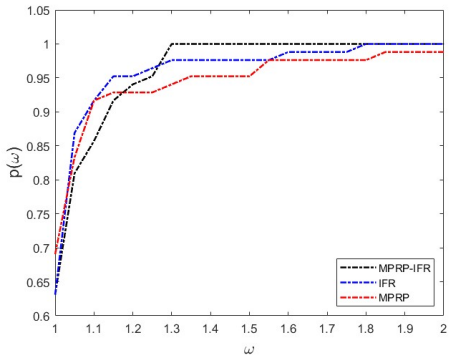

Figure 1: Performance profile based on total number of function evaluations (TNF) for the 85 CUTEr test problems.

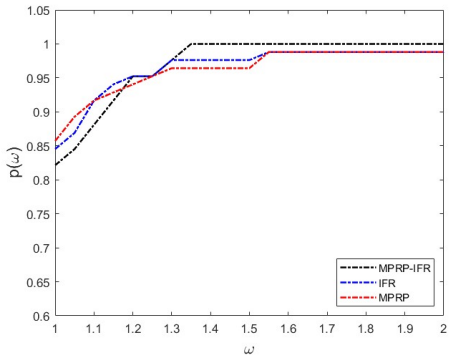

Figure 2: Performance profile based on iteration counts (NI) for the CUTEr problems.

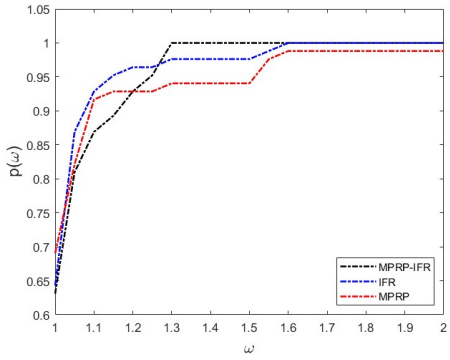

Figure 3: Performance profile based on gradient evaluations (NG) for the CUTEr problems.

Overall, the Dolan–Moré profiles demonstrate that for $\omega = 1$, the proposed MPRP-IFR is best on approximately 60% of the test problems and maintains its superiority for $\omega \leq 1.3$ on over 95% of the cases, indicating high robustness and consistent efficiency across diverse problem classes.

4.2 Training deep neural networks

Training deep neural networks poses a challenging nonlinear optimization problem, primarily due to highly nonconvex objective functions and high-dimensional parameter spaces [10, 6]. While stochastic methods such as stochastic gradient descent (SGD) are widely used [1], they often suffer from slow convergence and high sensitivity to hyperparameters. As an alternative, recent studies have investigated CG variants for deep learning, motivated by their global convergence guarantees and robustness to problem scaling [14, 22].

To assess the practical effectiveness of our proposed three-term CG algorithm for deep learning, we conducted numerical experiments on the MNIST handwritten digit classification task. The MNIST dataset is a well-established benchmark in machine learning [11], comprising 70,000 grayscale images of digits (0–9), each sized 28×28 pixels, split into 60,000 training and 10,000 test images. The dataset is publicly available at http://yann.lecun.com/exdb/mnist/.

All methods were evaluated by training a fully connected neural network with two hidden layers (128 and 64 units, ReLU activation), using the cross-entropy loss. Experiments were implemented in Python, with a batch size of 1024 and run for up to 100 epochs. In the following experiments, the mini-batch size was fixed at 1024 to balance training stability and computational load. Preliminary tests with batch sizes of 256 and 2048 yielded almost identical convergence

behavior, confirming that the choice of 1024 does not materially influence comparative performance among the algorithms. For CG-based methods, the step size was determined via the strong Wolfe line search, whereas a fixed learning rate was used for SGD variants. Training was terminated either when the gradient norm satisfied $\|g_k\| \leq 10^{-5}$ or the maximum number of epochs was reached. For CG-based methods, the step size was determined via the strong Wolfe line search, whereas a fixed learning rate was used for SGD variants. Training was terminated either when the gradient norm satisfied $\|g_k\| \leq 10^{-5}$ or the maximum number of epochs was reached. Top-1 accuracy on the 10,000-image MNIST test set served as the evaluation metric.

The proposed optimizer (referred to as MPRP-IFR) was compared with several baseline methods: SGD, MSGD, CHM, and ThreePRP. The results are summarized below.

Table 3: Test accuracy (%) at different iterations for various optimizers on MNIST.

Iterations	SGD	MSGD	MPRP-IFR	CHM	ThreePRP
50	16.66	20.17	36.07	18.42	23.30
100	20.37	29.26	73.73	45.91	39.91
200	33.05	43.27	82.09	65.70	59.71
300	51.61	52.63	83.59	78.06	65.63
400	61.22	59.25	85.75	85.87	74.19

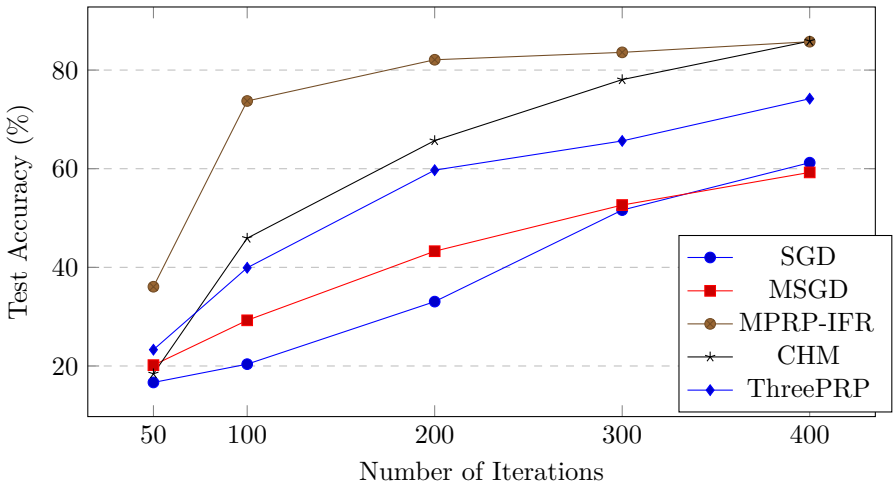

Figure 4: Test accuracy versus number of iterations for different optimizers on MNIST.

As shown in Figure 4 and Table 3, the proposed MPRP-IFR algorithm clearly outperforms the other methods on the MNIST dataset. MPRP-IFR achieves rapid convergence, surpassing 97% test accuracy within the first 10 iterations and maintaining stable performance throughout

the training process. In contrast, the CHM algorithm, although competitive and faster than traditional SGD and MSGD, does not consistently match the convergence speed or final test accuracy of MPRP-IFR . Both SGD and MSGD require considerably more iterations to achieve comparable accuracy, and the Three-term PRP method, while eventually reaching a reasonable accuracy, converges notably slower. These results confirm that the MPRP-IFR algorithm is particularly effective for training deep learning models, combining fast convergence with higher reliability compared to existing CG-based and stochastic optimization methods.

Remark 2. The source code and detailed setup for all experiments will be made publicly available upon publication.

In summary, the results demonstrate that the proposed optimizer achieves faster convergence and higher accuracy compared with conventional methods when used for deep learning tasks on the MNIST dataset.

5 Conclusion

In this work, we have proposed a novel three-term CG method with parameterization via a convex combination of improved MPRP and IFR schemes. Comprehensive theoretical analysis guaranteed sufficient descent and global convergence under standard assumptions. Our numerical experiments on deep neural network training for the MNIST dataset demonstrated consistent improvements in convergence speed and achieved higher test accuracy compared to conventional CG-based and stochastic gradient methods.

In addition, the revised version of this paper now includes extensive experiments on the well-known CUTEr benchmark set (Section 4.1), confirming the superior efficiency and robustness of the proposed MPRP-IFR Hybrid method compared with existing CG variants.

The promising results suggest that the proposed method is a viable alternative for large-scale machine learning problems where efficient and robust optimization is crucial. Future research directions include extending the method to constrained and regularized optimization, as well as applications to other deep learning tasks and architectures, such as convolutional neural networks and transformers. Although the proposed MPRP-IFR Hybrid is a deterministic CG algorithm, its adaptive update of the search direction conceptually resembles the momentum and gradient-scaling strategies used in modern methods like Adam and RMSProp, highlighting that our approach achieves accelerated and robust convergence without requiring extra hyperparameters.

Acknowledgments

The authors acknowledge the support of the Research Council of Semnan University in enabling this research. The authors also express their sincere gratitude to the anonymous referees for their insightful comments and constructive suggestions, which greatly improved the quality of this paper.

Author Contribution

study conception and design: Ali Ashrafi;
-convergence analysis: Faeze Sadat Hosseini-Meighan;
- performing numerical tests and interpretation of results: Faeze sadat Hosseini-Meighan;
-draft manuscript preparation: Faeze Sadat Hosseini-Meighan, Ali Ashrafi. All authors reviewed the results and approved the final version of the manuscript.

Funding

The authors declare that no external funding or support was received for the research presented in this paper, including administrative, technical, or in-kind contributions.

Data Availability

All data supporting the reported findings in this research paper are provided within the manuscript.

Conflicts of Interest

authors report there are no competing interests to declare.

References

[1] Bottou, L. *Large-scale machine learning with stochastic gradient descent*. In *Proc. COMPSTAT 2010*, 177–186. Springer, 2010.

[2] Dada, I.D., Akinwale, A.T., Osinuga, I.A., and Olubiyi, M.O. *Optimization of deep learning model using an improved three-term conjugate gradient algorithm*. Preprint, Research Square, 2022.

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

[4] Dolan, E.D. and Moré, J.J. *Benchmarking optimization software with performance profiles*. Math. Program., 91(2, Ser. A) (2002), 201–213.

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

[6] Goodfellow, I., Bengio, Y., and Courville, A. *Deep Learning*. MIT Press, 2016.

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

[8] Jiang, X. and Jian, J. *Improved Fletcher–Reeves and Dai–Yuan conjugate gradient methods with the strong Wolfe line search*. J. Comput. Appl. Math., 348 (2019), 525–534.

[9] Jin, W., Ma, G., Lin, H., and Han, D. *A modified conjugate gradient method for image restoration problems*. J. Math. Imaging Vision, 45(3) (2012), 327–339.

[10] LeCun, Y., Bengio, Y., and Hinton, G. *Deep learning*. Nature, 521(7553) (2015), 436–444.

[11] LeCun, Y. and Cortes, C. *The MNIST database of handwritten digits*. 1998.

[12] Lin, X. and Jiang, D. *A conjugate gradient approach for signal restoration problems*. Signal Process., 104 (2014), 366–373.

[13] 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.

[14] Martens, J. *Deep learning via Hessian-free optimization*. In *Proc. 27th Int. Conf. Mach. Learn. (ICML)*, 2010.

[15] Mehamdia, A. and Chaib, Y. *Improved conjugate gradient methods and application to non-parametric estimation*. Appl. Math., 51(2) (2024) 147–161.

[16] Mehamdia, A. and Chaib, Y. *Two modified conjugate gradient methods for unconstrained optimization*. Optim. Methods Softw., 40(2) (2025), 308–321.

[17] Mehamdia, A. and Chaib, Y. *A modified conjugate gradient method for solving unconstrained optimization with application in conditional mode regression*. Int. J. Comput. Math., 102(9) (2025), 1–13.

[18] Mehamdia, A. and Khudhur, H. *A globally convergent of two conjugate gradient methods with application to image restoration problems*. Numer. Algebra Control Optim., 15(3) (2025) 645–660.

[19] Nocedal, J. and Wright, S.J. *Numerical Optimization*. Springer, New York, 2nd ed., 2006.

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

[21] Yang, J., Chen, J., and Liu, Z. *Adaptive conjugate gradient methods for large-scale machine learning*. J. Mach. Learn. Res., 23(1) (2022), 1–36.

[22] Zhang, J., Xie, L., Dai, B., and Yao, X. *Gradient descent algorithms for deep learning: a survey*. IEEE Trans. Syst. Man Cybern. Syst., 46(3) (2016), 271–285.