J. Math. Computer Sci., 36 (2025), 326–332

Online: ISSN 2008-949X

Journal of Mathematics and Computer Science

Journal Homepage: www.isr-publications.com/jmcs

A new spectral conjugate gradient method for unconstrained optimization and its application in neural networks

Asmaa M. Abdulrahman[a,*], Bayda G. Fathi[b], Huda Y. Najm[a]

[a]College of Science, University of Duhok, Iraq.

[b]College of Science, University of Zakho, Iraq.

Abstract

This work introduces a new variation of the Hestenes and Stiefel nonlinear conjugate gradient (HS) method by combining the advantages of the spectral conjugate gradient method and the conjugacy condition of the quasi-Newton method. The proposed method incorporates inexact line searches and categorizing it as a descent method. By employing line searches that satisfy the Wolfe conditions, we establish sufficient descent properties and global convergence condition, assuming that the appropriate conditions are met. Additionally, we perform numerical experiments utilizing benchmark functions frequently used in optimization assignments to evaluate the effectiveness of the proposed method. The results demonstrate that our method outperforms the traditional HS method. Furthermore, we successfully implement the newly developed technique to train neural networks (NNs), demonstrating its practicality for non-traditional optimization tasks.

Keywords: Unconstrained optimization, conjugate gradient method, Wolfe condition, sufficient descent, global convergence, neural networks.

2020 MSC: 42A50, 90C52, 74P10.

1. Introduction

Neural Network (NN) is a computational learning system that employs a complex network of interconnected functions to process and analyze the input data. Each individual neuron in this network is responsible for converting the received input into a corresponding output signal. The neural units that are depicted here, establish intra-neuronal connections with at least one alternative neural unit. The weight coefficient (w_i) that modulates these connections denotes the relative importance of a given connection within the neural network [6]. The training of neural networks (NNs) can be expressed as a nonlinear unconstrained optimization problem. As a result, the training process involves minimizing the error function $E(w)$. The standard performance measure for feedforward networks is typically represented by the

*Corresponding author

Email addresses: asmaa.abdulrahman@uod.ac (Asmaa M. Abdulrahman), bayda.fathi@uoz.edu.krd (Bayda G. Fathi), huda.najm@uod.ac (Huda Y. Najm)

doi: 10.22436/jmcs.036.03.07

Received: 2024-03-27 *Revised:* 2024-05-10 *Accepted:* 2024-07-08

squared error, which is calculated as the average of the squared differences between the output of the network and the desired target that is:

$$E(w) = \frac{1}{2}\sum_{p=1}^{N}\sum_{i=1}^{p}(O_i^p - T_i^p)^2.$$

The above equation pertains to the computation of the squared difference error denoted by $(O_i^p - T_i^p)^2$. This value specifically represents the difference between the output at the k^{th} output layer neuron and the corresponding target output value for a given pattern p, where $w \in R^m$ is the vector network weights [16].

2. The conjugate gradient method

As shown in the following unconstrained optimization problem

$$\min f(x)\ \forall x \in R^m,$$

where f is a continuously differentiable function, conjugate gradient (CG) method generates a sequence $\{x_n\}$ such that the n^{th} iterate is given by

$$x_{n+1} = x_n + \alpha_n d_n, \quad n = 0, 1, 2, \ldots, \tag{2.1}$$

where α_n is a small positive number called a step size, and d_n is the search direction usually given by

$$d_{n+1} = \begin{cases} -g_{n+1}, & \text{if } n = 0, \\ -g_{n+1} + \beta_n d_n, & \text{if } n \geqslant 1, \end{cases}$$

where $g_n = \nabla f(x_n)$ and β_n is the scalar parameter. Certain choices for the parameter β_n correspond to different CG methods. Some well-known formulae for β_n are given below:

$$\beta_n^{HS} = \frac{g_{n+1}^T y_n}{d_n^T y_n}, \quad \text{[11, Hestens-Stiefele (HS)]}, \tag{2.2}$$

$$\beta_n^{FR} = \frac{g_{n+1}^T g_{n+1}}{g_n^T g_n}, \quad \text{[8, Fletcher-Reeves (FR)]},$$

$$\beta_n^{PRP} = \frac{g_{n+1}^T y_n}{g_n^T g_n}, \quad \text{[14, 15, Polak-Ribiere and Polak (PRP)]},$$

$$\beta_n^{DY} = \frac{\|g_{n+1}\|^2}{d_n^T y_n}, \quad \text{[5, Dai-Yuan (DY)]},$$

where $y_n = g_{n+1} - g_n$ and $\|.\|$ is the Euclidean norm. The stepsize α_n is usually obtained by inexact linear search with Wolfe's criterion given by

$$f(x_n + \alpha_n d_n) \leqslant f(x_n) + \sigma_1 \alpha_n g_n^T d_n, \tag{2.3}$$

$$d_n^T g(x_n + \alpha_n d_n) \geqslant \sigma_2 g_n^T d_n, \tag{2.4}$$

with $0 < \sigma_1 < \sigma_2 < 1$, [9, 20]. The sufficient descent condition holds if there exist a constant $c > 0$ such that $g_{n+1}^T d_{n+1} \leqslant -c\|g_{n+1}\|^2$, $\forall n \geqslant 0$.

In 1952, Hestenes and Stiefel [11] proposed the HS conjugate gradient method using β_n as defined in (2.2). In 2017, two new spectral conjugate gradient methods (HS and DY) with better convergence were proposed [18]. Based on Armijo-type line search, a new formula for HS has been developed [13]. To enhance the performance of the direction, a hybrid combination of the HS formula and the PRP formula was suggested in [12, 19]. A hybrid conjugate gradient method also extends to applied mathematics [17, 23]. Numerous approaches for solving problems of this kind have been proposed and are classified into physics and engineering [21, 22].

3. Proposed method and the sufficient descent condition

In this section, a modification of the Hestens-Stiefel method is introduced, where the search direction is given by

$$d_{n+1} = \begin{cases} -g_{n+1}, & \text{if } n = 0, \\ -(1+\theta_n)g_{n+1} + \beta_n^{HS} d_n, & \text{if } n \geqslant 1. \end{cases} \tag{3.1}$$

In order to get the formula for θ_n, we multiply both sides of (3.1) by y_n, and apply the recent update of the conjugacy condition with the condition: $d_{n+1}^T y_n = -t g_{n+1}^T s_n$, where $t \geqslant 0$ and $s_n = x_{n+1} - x_n = \alpha_n d_n$ by Dai and Liao [4]. Doing so, we get:

$$-t g_{n+1}^T s_n = -g_{n+1}^T y_n - \theta_n g_{n+1}^T y_n + \frac{g_{n+1}^T y_n}{d_n^T y_n} d_n^T y_n.$$

This implies that $\theta_n = t\frac{g_{n+1}^T s_n}{g_{n+1}^T y_n}$. Since t is a parameter, suppose t has the following form:

$$t = \frac{g_{n+1}^T y_n}{g_{n+1}^T s_n} - \mu \frac{g_{n+1}^T y_n}{s_n^T y_n},$$

where $\mu \in [0,1]$. Now since $s_n = \alpha_n d_n$, we get

$$\theta_n = \frac{g_{n+1}^T s_n}{g_{n+1}^T y_n}\left[\frac{g_{n+1}^T y_n}{g_{n+1}^T s_n} - \mu \frac{g_{n+1}^T y_n}{s_n^T y_n}\right] = \beta_n^{HS}\frac{g_{n+1}^T d_n}{\|g_{n+1}\|^2} - \mu \frac{g_{n+1}^T d_n}{d_n^T y_n}. \tag{3.2}$$

3.1. Outline of the new proposed method

The new algorithm includes 6 steps given as follows.

Step 1. Select any initial point $x_1 \in R^m$, and accuracy tolerance $\epsilon > 0$. Let $d_1 = -g_1 = -\nabla f(x_1)$, and set $n = 1$.
Step 2. If $\|g_n\| < 0$, terminate, otherwise go to step 3.
Step 3. Compute step length α_n by an inexact line search.
Step 4. Create the next iteration by $x_{n+1} = x_n + \alpha_n d_n$, determine the gradient $g_{n+1} = \nabla f(x_{n+1})$, the spectral parameter θ_n from the equation (3.2), and β^{HS} from (2.2).
Step 5. Compute $d_{n+1} = -(1+\theta_n)g_{n+1} + \beta_n{}^{HS} d_n$.
Step 6. Set $n = n+1$, and return to step 2.

3.2. The sufficient descent condition

The descent property for the suggested conjugate gradient technique (3.1) is demonstrated in the following lemma.

Lemma 3.1. *Let a sequence $\{x_n\}$ and d_n be generated by (3.1), and α_n be obtained by the Wolfe's line search (2.3) and (2.4), if $\mu \in [0,1]$, then we have $g_{n+1}^T d_{n+1} \leqslant -c\|g_{n+1}\|^2$, where $c = \left(1 + \frac{\mu\sigma_2}{1-\sigma_2}\right)$ and $\sigma_2 \in (0,1)$.*

Proof.

Case 1. If $\mu = 0$, from (3.2) and (3.1) we get $g_{n+1}^T d_{n+1} = -\|g_{n+1}\|^2$.

Case 2. If $\mu \neq 0$, the lemma can be proved by induction. For $n = 1$, $g_1^T d_1 = -\|g_1\|^2 \leqslant 0$. Assume $g_i^T d_i \leqslant 0$ for all $i = 1, 2, \ldots, n$. From (3.1) and (2.2) we get:

$$g_{n+1}^T d_{n+1} = \left[-1 + \mu \frac{g_{n+1}^T d_n}{d_n^T y_n}\right]\|g_{n+1}\|^2$$

$$= \left[-(1-\mu) - \mu\left(1 - \frac{g_{n+1}^T d_n}{d_n^T y_n}\right)\right] \|g_{n+1}\|^2 \leqslant \left[-(1-\mu) - \mu\frac{g_n^T d_n}{d_n^T y_n}\right] \|g_{n+1}\|^2.$$

Since $y_n = g_{n+1} - g_n$ and from the Wolfe's condition (2.4), we obtain

$$g_{n+1}^T d_{n+1} \leqslant \left[-(1-\mu) - \mu\frac{g_n^T d_n}{(1-\sigma_2)\, d_n^T g_n}\right] \|g_{n+1}\|^2.$$

Now let $c = \left(1 + \frac{\mu\sigma_2}{1-\sigma_2}\right)$, which is a positive number, then $g_{n+1}^T d_{n+1} \leqslant -c\|g_{n+1}\|^2$. This complete the proof. □

3.3. Convergence analysis of the developed method

The following assumptions are frequently required to prove the global converge analysis of any CG process.

(1) Assume that $f(x)$ is bounded below on the level set R^m and differentiable in a neighborhood N of the level set $S = \{x \in R^m : f(x) \leqslant f(x_0)\}$, then there exists a constant $\gamma > 0$ such that $\|g(x)\| \leqslant \gamma$, $\forall x \in S$.
(2) The gradient $g(x)$ is Lipschitz continuous in N, if $\exists L > 0$, s.t., $\|g(x) - g(x_n)\| \leqslant L\|x - x_n\|$, $\forall x, x_n \in N$.

These assumptions lead to the following lemma.

Lemma 3.2 ([5, Zountendijk theorem]). *Once the sequence $\{x_n\}$ is clearly generated by (2.1), the step size α_n satisfies (2.3) and (2.4) and d_n is the descent direction, then the following holds:*

$$\sum_{n=1}^{\infty} \frac{(g_n^T d_n)^2}{\|d_n\|^2} < \infty.$$

Theorem 3.3. *Assume that assumption (1) is true. If $\mu \in [0,1)$ and the sequence $\{x_n\}$ generated by using the algorithm in subsection (3.1), where α_n satisfies the Wolfe's conditions, then $\liminf_{n\to\infty} \|g_n\| = 0$.*

Proof. The theorem can be proved by using contradiction. That is, there exists a positive constant δ, such that $\|g_n\| \geqslant \delta$ or equivalently $\frac{1}{\|g_n\|^2} \leqslant \frac{1}{\delta^2}$ for all n. Multiplying (3.1) by g_{n+1} and substituting β^{HS} as in (2.2), we obtain

$$d_{n+1}^T g_{n+1} = -\|g_{n+1}\|^2 + \mu\frac{g_{n+1}^T d_n}{d_n^T y_n}\|g_{n+1}\|^2. \tag{3.3}$$

Dividing both sides of (3.3) by $\|g_{n+1}\|^2$ and using the Lipschitz condition, we get

$$\frac{g_{n+1}^T d_{n+1}}{\|g_{n+1}\|^2} + 1 \geqslant \mu\frac{g_{n+1}^T d_n}{L d_n^T s_n}.$$

Using (2.4), we have

$$\frac{d_{n+1}^T g_{n+1}}{\|g_{n+1}\|^2} + 1 \geqslant \frac{\sigma_2 g_n^T d_n}{L\alpha_n\|d_n\|^2}.$$

Therefore,

$$\frac{L\alpha_n}{\mu\sigma_2}\left[\frac{d_{n+1}^T g_{n+1}}{\|g_n + 1\|^2} + 1\right] \geqslant \frac{g_n^T d_n}{\|d_n\|^2}. \tag{3.4}$$

Squaring both sides of (3.4) implies that

$$\left(\frac{L\alpha_n}{\mu\sigma_2}\right)^2 \left(\frac{d_{n+1}^T g_{n+1}}{\|g_n + 1\|^2} + 1\right)^2 \|d_n\|^2 \geqslant \frac{(g_n^T d_n)^2}{\|d_n\|^2}.$$

Since $\cos^2(\theta) = \frac{(g_n^T d_n)}{\|d_n\|\|g_n\|}$, then

$$\sum_{n=1}^{\infty} \left(\frac{L\alpha_n}{\mu\sigma_2}\right)^2 \left(\frac{d_{n+1}^T g_{n+1}}{\|g_{n+1}\|^2} + 1\right)^2 \|d_n\|^2 \geqslant \sum_{n=1}^{\infty} \|g_n\|^2,$$

that is

$$\sum_{n=1}^{\infty} \frac{(g_n^T d_n)^2}{\|d_n\|^2} \geqslant \sum_{n=1}^{\infty} \delta^2 = \infty. \tag{3.5}$$

This contradicts the Zourtendijk theorem. Therefore, from (3.5) it follows that

$$\sum_{n=1}^{\infty} \frac{(g_n^T d_n)^2}{\|d_n\|^2} < \infty.$$

This completes the proof. Hence it suggests that our method has a property of global convergence. □

4. Numerical results and discussion

We have chosen 45 unconstrained optimization problems in the range $[n = 1000, 2000, \ldots, 10,000]$ broadly and they are based on the generalization in [1]. All algorithms used strong Wolfe condition . The codes are adopted with double precision and using the Fortran language. All of these codes are authored by Andrei [2, 3]. The following figures use Dolan and Moré's analysis [7] to show how efficient our algorithm is. Our new method (New) needs fewer iterations (NI) and function evaluations (NF) than the standard HS ([11]), CR ([17]), and HYB ([10]) conjugate gradient parameters. You can see this in Figures 1 and 2. Moreover, Table 1 presents the percentage of enhancement achieved by our new spectral CG approach. In comparison with the HS method, our new method exhibits improvements of 20.03% in NI and 33.77% in NF.

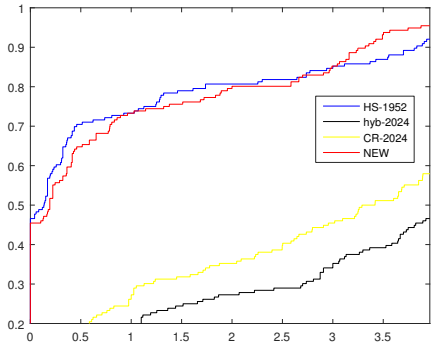

Figure 1: Performance profile based on NI.

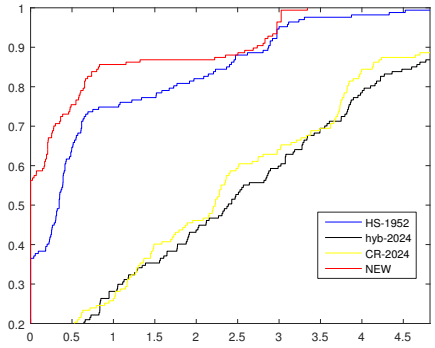

Figure 2: Performance profile based on NF.

Table 1: The percentage of improvement between the standard HS method and the new method.

Tools	HS method	NEW method
NI	100%	79.97%
NF	100%	66.23%

5. Results of training neural network using the new spectral method

In this section, a new spectral technique has been employed to train the neural networks in classical artificial intelligence problems (continuous function approximation). This paper presents a comparative analysis between the conventional HS method and a new method developed in this study. The current study utilized version (8.1) of the MATLAB neural network toolbox to perform the CG method. Consider the approximation of the continuous trigonometric function $f(x) = \sin(x) + \cos(2x)$, where $x \in [0, \pi]$. The outcomes of the training activity are presented in Table 2, while Figures 3 and 4 provide graphical illustrations of the results. The standard HS method and our new method have been compared with the same input and target values. The target error threshold remains at $1 * 10^{-15}$, and the maximum number of epochs is set to 300. The numerical findings reveal that the new method significantly improved the efficiency of the neural networks in comparison to the traditional HS method.

Table 2: Comparison of the performance between the new method and the standard HS approach.

Methods	No. Running	Epochs	CPU time(s)/Epoch	Gradient	Step size
HS	1	266	0.00.01	0.000215	0.00
	2	147	0.00.37	0.000201	0.00
	3	145	0.00.36	0.000221	0.00
	4	119	0.00.35	0.000199	0.00
New	1	40	0.00.01	0.0180	0.00
	2	114	0.00.27	0.000202	0.00
	3	3	0.00.00	0.180	0.00
	4	105	0.00.24	0.000199	0.00

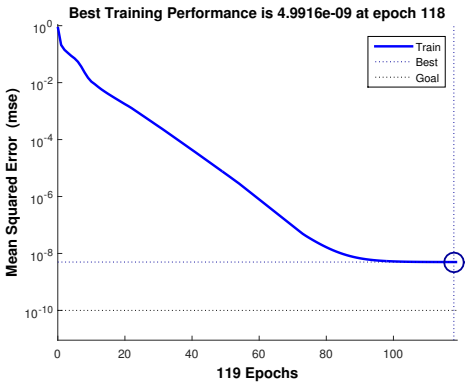

Figure 3: First epoch performance of standard HS method for training neural networks

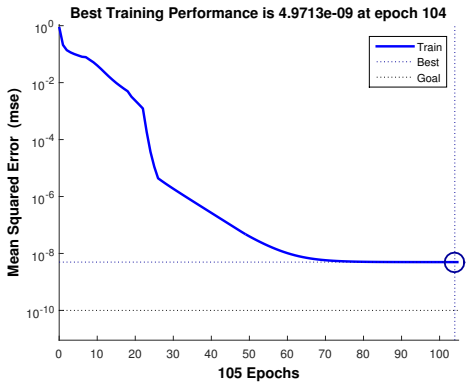

Figure 4: First epoch performance of the new method for training neural networks

6. Conclusion

In this study, a new approach to network training has been developed using the conjugate gradient method. Furthermore, the new method is globally compatible with descent and satisfies adequate conditions. The proposed method has clearly demonstrated its efficiency according to the numerical results.

References

[1] N. Andrei, *An unconstrained pptimization test functions collection*, Adv. Model. Optim., **10** (2008), 147–161. 4
[2] N. Andrei, *New Accelerated Conjugate Gradient Algorithms for Unconstrained Optimization*, ICI Technical Report, (2008), 1–18. 4

[3] N. Andrei, *New accelerated conjugate gradient algorithms as a modification of Dai-Yuan's computational scheme for unconstrained optimization*, J. Comput. Appl. Math., **234** (2010), 3397–3410. 4
[4] Y.-H. Dai, L.-Z. Liao, *New conjugacy conditions and related nonlinear conjugate gradient methods*, Appl. Math. Optim., **43** (2001), 87–101. 3
[5] Y. H. Dai, Y. Yuan, *A nonlinear conjugate gradient method with a strong global convergence property*, SIAM J. Optim., **10** (1999), 177–182. 2, 3.2
[6] H. Demuth, M. Beale, *Neural network toolbox user's guide*, MathWorks, (2000). 1
[7] E. D. Dolan, J. J. Moré, *Benchmarking optimization software with performance profiles*, Math. Program., **91** (2002), 201–213. 4
[8] R. Fletcher, C. M. Reeves, *Function minimization by conjugate gradients*, Comput. J., **7** (1964), 149–154. 2
[9] J. C. Glibart, J. Noceded, *Global convergence properties of conjugate gradient methods for optimization*, SIAM J. Optim., **2** (1992), 21–42. 2
[10] S. B. Hanachi, B. Sellami, M. Belloufi, *A new family of hybrid conjugate gradient method for unconstrained optimization and its application to regression analysis*, RAIRO Oper. Res., **58** (2024), 613–627. 4
[11] M. R. Hestenes, E. Stiefel, *Methods of conjugate gradients for solving linear systems*, J. Research Nat. Bur. Standards, **49** (1952), 409–436. 2, 2, 4
[12] N. A. Japri, S. Basri, M. Mamat, *New modification of the Hestenes-Stiefel with strong wolfe line search*, AIP Conf. Proc., **2355** (2021), 1–7. 2
[13] Y. Li, S. Du, *Modified HS conjugate gradient method for solving generalized absolute value equations*, J. Inequal. Appl., **2019** (2019), 12 pages. 2
[14] B. T. Polyak, *The conjugate gradient method in extremal problems*, USSR Comput. Math. Math. Phys., **9** (1969), 94–112. 2
[15] E. Polak, G. Ribiere, *Note on the convergence of methods of conjugate directions*, Rev. Franç. Inform. Rech. Opér., **3** (1969), 35–43. 2
[16] D. E. Rumelhart, J. L. McClelland, PDP Research Group, *Parallel Distributed Processing: Explorations in the Microstructure of Cognition, Vol. 1: Foundations*, MIT Press, Cambridge, MA, (1986). 1
[17] C. Souli, R. Ziadi, A. Bencherif-Madani, H. M. Khudhur, *A hybrid CG algorithm for nonlinear unconstrained optimization with application in image restoration*, J. Math. Model., **12** (2024), 301–317. 2, 4
[18] G. Wang, R. Shan, W. Huang, W. Liu, J. Zhao, *Two new spectral conjugate gradient algorithms based on Hestenes-Stiefel*, J. Algorithms Comput. Technol., **11** (2017), 345–352. 2
[19] X. Wu, Y. Zhu, J. Yin, *A HS-PRP-type hybrid conjugate gradient method with sufficient descent property*, Comput. Intell. Neurosc., **2021** (2021), 8 pages. 2
[20] H. Yabe, M. Takano, *Global convergence properties of new nonlinear conjugate gradient methods for unconstrained optimization*, Comput. Optim. Appl., **28** (2004), 203–225. 2
[21] R. Ziadi, A. Bencherif-Madani, *A perturbed Quasi-Newton algorithm for bound-constrained global optimization*, J. Comput. Math., **20** (2023), 1–29. 2
[22] R. Ziadi, A. Bencherif-Madani, *A mixed algorithm for smooth global optimization*, J. Math. Model., **11** (2023), 207–228. 2
[23] N. Zullpakkal, N. 'Aini, N. Ghani, N. Mohamed, N. Idalisa, M. Rivaie, *Covid-19 data modelling using hybrid conjugate gradient method*, J. Inform. Optim. Sci., **4** (2022), 837–853. 2