Indonesian Journal of Electrical Engineering and Computer Science
Vol. 33, No. 1, January 2024, pp. 93~100
ISSN: 2502-4752, DOI: 10.11591/ijeecs.v33.i1.pp93-100

A new conjugate gradient for unconstrained optimization problems and its applications in neural networks

Alaa Luqman Ibrahim[1], Mohammed Guhdar Mohammed[2]
[1]Department of Mathematics, Faculty of Science, University of Zakho, Zakho, Iraq
[2]Department of Computer Science, Faculty of Science, University of Zakho, Zakho, Iraq

Article Info

Article history:

Received Oct 31, 2022
Revised Oct 18, 2023
Accepted Oct 25, 2023

Keywords:

Conjugate gradient method
Global convergence
Neural networks
The descent property
Unconstrained optimization

ABSTRACT

We introduce a novel efficient and effective conjugate gradient approach for large-scale unconstrained optimization problems. The primary goal is to improve the conjugate gradient method's search direction in order to propose a new, more active method based on the modified vector v_k^*, which is dependent on the step size of Barzilai and Borwein. The suggested algorithm features the following traits: (i) The ability to achieve global convergence; (ii) numerical results for large-scale functions show that the proposed algorithm is superior to other comparable optimization methods according to the number of iterations (NI) and the number of functions evaluated (NF); and (iii) training neural networks is done to improve their performance.

Corresponding Author:

Alaa Luqman Ibrahim
Department of Mathematics, Faculty of Science, University of Zakho
Zakho, Kurdistan Region, Iraq
Email: alaa.ibrahim@uoz.edu.krd

1. INTRODUCTION

In this section, we will offer a concise yet comprehensive overview of unconstrained optimization and its intersection with neural networks. Unconstrained optimization refers to the mathematical techniques and algorithms used to find the optimal solution of a problem without any constraints on the decision variables. It plays a pivotal role in various machine learning and deep learning applications, especially when training neural networks. Neural networks, on the other hand, are a class of machine learning models inspired by the human brain, consisting of interconnected nodes and layers that learn patterns from data.

– Unconstrained optimization

Take into account the following unconstrained optimization problem;

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

where a function $f: R^n \to R$ be continuously differentiable R^n denotes an n-dimensional euclidean space. Numerous real-world application areas exist for the aforementioned issue, including economics, biology, and engineering. The nonlinear conjugate gradient (CG) method is one of the most well-known and successful methods for (1). Starting with an initial guess of $x_0 \in R^n$, we should create a sequence of $\{x_k\}$ such;

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

where $\alpha_k > 0$ is achieved by line search and the direction d_k are generated as;

$$d_{k+1} = \begin{cases} -g_{k+1}, & k = 0 \\ -g_{k+1} + \beta_k d_k, & k > 0 \end{cases} \quad (3)$$

where $g_k = \nabla f(x_k)$ and β_k is a scalar parameter [1], [2]. In addition, for many years, researchers focused researchers to proposed variety expression of β_k as well as d_{k+1} (see, e.g., [2]–[10]) to increasing more efficient and successful conjugate gradient algorithms. Perry [11] suggested a parameter of conjugate gradient defined as;

$$\beta_k^{Perry} = \frac{g_{k+1}^T(y_k - v_k)}{d_k^T y_k} \quad (4)$$

where $y_k = g_{k+1} - g_k$ and $v_k = x_{k+1} - x_k$. When calculating the step-size, it is argued that α_k satisfies any of the conditions for a line search [12]–[14]. In this study, we employ a strong wolfe line search.

$$f(x_k + \alpha_k d_k) \leq f(x) + \delta \alpha_k g_k^T d_k, 0 \leq \delta \leq \frac{1}{2} \quad (5)$$

$$|d_k^T g(x_k + \alpha_k d_k)| \leq -\sigma g_k^T d_k, \delta \leq \sigma \leq 1 \quad (6)$$

– Summary of artificial neural network history

In order to execute network pattern decision making, ANN is inspired by the biological human brain, which is made up of up to 60 trillion interconnected groups of neurons. Based on this fundamental concept, an artificial neural network's construction starts with a collection of basic, interconnected neurons that function as a single processor. Based on the neuron model developed by McCulloch and Pits, the perceptron notion has been introduced [15]. A single layer of input, process, and output elements forms the basis of an ANN. In order to generate the best outcome for any datasets or issue segments, ANN acts as a complicated mathematical formulation starting from a very fundamental understanding of the information processing cycle. Neurons should be checked using feed-forward and backward algorithms in order to complete a network cycle. Back propagation or backward algorithms are the topics that have received the most attention from researchers in the past and now. The machine learning component of ANN that has gained the most attention in current research and development. Machine learning refers to the capability of a computer to comprehend data structure utilizing mathematical or statistical models. Artificial neural networks, on the other hand, require additional processes in order to deal with complex patterns in vast volumes of data. Deep methodological learning and structured algorithms are needed for this. In its execution process, ANN uses a variety of methodologies, including supervised learning, unsupervised learning, and reinforcement learning. Neural network learning techniques are still being studied and recognized by communities as of this writing. It quickly became well-liked in machine learning [16]–[18] and is frequently considered to be more effective because to its exceptional capacity for self-adaptation and self-learning; $net_j^l = \sum_{i=1}^{N_{l-1}} w_{ij}^{l-1,l} y_i^{l-1} + b_j^l, y_i^l = f\left(net_j^l\right)$ The determines how a FNN works.

Where the sum of its weighted inputs is net_j^l, for the j^{th} node in the l^{th} layer $(j = 1, \ldots, Nl)$, $w_{ij}^{l-1,l}$ are the weights from the i^{th} neuron at the layer to the j^{th} neuron at the l^{th} layer, b_j^l is the bias of the j^{th} neuron at the l^{th} layer, y_i^l is the outputof the j^{th} neuron that belongs to the l^{th} layer, and $f\left(net_j^l\right)$, is the j^{th} neuron activation function.

The core concept of neural network training can be expressed as a nonlinear unconstrained optimization problem. In order to globally reduce the difference between the network's actual output and the planned output for all examples in the training set, a neural network is trained by incrementally changing its weights [19]. The training procedure may therefore be described mathematically as the minimization of the error function $E(w)$, which is defined by the sum of square differences between the actual output of the FNN, i.e.,

$$E(w) = \sum_{p=1}^{p} \sum_{j=1}^{N_l} (y_i^{l-1} - t_{j,p})^2 \quad (7)$$

where $w \in R^n$ is the vector network weights and the number of patterns used in the training set represented by P. [3]-[20].

The remainder of the paper is formatted as follows: we present our strategy for acquiring the new propose CG algorithm acquiring in section 2. The descent and sufficient descent property of our approach and the global convergence property were tested in section 3. In section 4, some numerical experiments to new CG algorithm and the HS algorithm are reported. In section 5, presents the application of the CG algorithm for training neural networks; the paper's conclusion and algorithm's characters were listed in section 5.

2. NEW CONJUGATE GRADIENT ALGORITHM FOR SMOOTH PROBLEMS

The nonlinear conjugate gradient algorithm is a well-known example of an efficient technique for optimization problems due to its low storage requirements and easy structure. It encourages us to conduct additional research and develop a modified conjugate gradient formula for the unrestricted optimization model. In this section, we well submit a new coefficient conjugate gradient algorithm for unconstrained minimization problems depend on the modified vector which is defined as: $v_k^* = \frac{1}{\alpha_k^{BB}} v_k - \frac{\theta}{\alpha_k^{BB}} v_k$, where $\theta \in (0,1)$ and $\alpha_k^{BB} = \frac{v_k^T v_k}{y_k^T v_k}$, see [21].

So, we have $v_k^* = (1-\theta)\frac{y_k^T v_k}{v_k^T v_k} v_k$ (8)

More precisely, the conjugate gradient algorithms are iterative methods of the form given by (1) and (2). The major idea of our new algorithm is to improve the performance of CG algorithm Perry by replacing v_k by v_k^* in (4). Then,

$$\beta_k^{New} = \frac{g_{k+1}^T(y_k - v_k^*)}{d_k^T y_k} = \frac{g_{k+1}^T y_k}{d_k^T y_k} - (1-\theta)\frac{y_k^T v_k}{v_k^T v_k}\frac{g_{k+1}^T v_k}{d_k^T y_k} \quad (9)$$

Algorithm steps: iterative gradient method for unconstrained optimization:
1) Given an initial point $x_0 \in R^n$
2) Set $d_0 = -g_0$, $k = 0$. If $\|g_k\| = 0$ then stop, otherwise go to step 3.
3) Compute the step size α_k by using minimize $f(x_k + \alpha_k d_k)$.
4) Set $x_{k+1} = x_k + \alpha_k d_k$.
5) Determine g_{k+1}, if $\|g_{k+1}\| \leq 10^{-5}$ stop, else go to step 6.
6) Compute d_{k+1} by (2) and β_k^{New} from (9).
7) If $\|g_{k+1}\|^2 \leq \frac{|g_k^T g_{k+1}|}{0.2}$ is satisfied go to step 3, else $k = k+1$ and go to step 3.

3. CHARACTERISTICS OF ALGORITHMS

The properties of the descent and sufficient descent, as well as global convergence of the new algorithm, are stated in this section:

Theorem 3.1: if the search direction d_{k+1} is generated by (2) and β_k^{New} from (9), then,

$$g_{k+1}^T d_{k+1} \leq 0$$

Proof: multiply both sides of (2) by g_{k+1} where β_k^{New} denied in (9), to obtain,

$$g_{k+1}^T d_{k+1} = -g_{k+1}^T g_{k+1} + \frac{g_{k+1}^T y_k}{d_k^T y_k} g_{k+1}^T d_k - (1-\theta)\frac{y_k^T v_k}{v_k^T v_k}\frac{g_{k+1}^T v_k}{d_k^T y_k} g_{k+1}^T d_k \quad (10)$$

if the above search direction is exact, then it is satisfying the descent condition i.e.,

$$g_{k+1}^T d_{k+1} = -\|g_{k+1}\|^2 \leq 0$$

however, if the search direction (10) is inexact (i.e.) $g_{k+1}^T d_k \neq 0$. We get to the conclusion that the first two terms in (10) satisfy the requirement for descent, i.e.,

$$-g_{k+1}^T g_{k+1} + \frac{g_{k+1}^T y_k}{d_k^T y_k} g_{k+1}^T d_k \leq 0 \quad (11)$$

because $g_{k+1}^T d_k \leq d_k^T y_k$ and $g_{k+1}^T y_k = g_{k+1}^T g_{k+1} - g_{k+1}^T g_k$ and $0.2\|g_{k+1}\|^2 \leq |g_k^T g_{k+1}|$ we have since, $g_{k+1}^T v_k = \alpha_k g_{k+1}^T d_k$. Therefore, we can present the (10) as,

$$g_{k+1}^T d_{k+1} \leq -(1-\theta)\alpha_k \frac{y_k^T v_k}{v_k^T v_k \; d_k^T y_k}(g_{k+1}^T d_k)^2 \quad (12)$$

clearly, $\alpha_k, (1-\theta), y_k^T v_k, v_k^T v_k$, $d_k^T y_k$ and $(g_{k+1}^T d_k)^2$ are non-negative. So, we have $g_{k+1}^T d_{k+1} \leq 0$.

Theorem 3.2: the direction d_{k+1} defined by (2) and β_k^{New} from (9); then there exists a positive $\mu > 0$ satisfying.

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

Proof: it is clear from Theorem (3.1) that the first two components of (10) are less than or equal to zero after multiplying the new search direction (2) and (9) by g_{k+1}, and from (12), we obtain,

$$g_{k+1}^T d_{k+1} \leq -(1-\theta)\alpha_k \frac{y_k^T v_k \, \|d_k\|^2}{v_k^T v_k \, d_k^T y_k} \|g_{k+1}\|^2 \quad (13)$$

Let $\mu = -(1-\theta)\alpha_k \frac{y_k^T v_k \, \|d_k\|^2}{v_k^T v_k \, d_k^T y_k}$ which is positive, then, $g_{k+1}^T d_{k+1} \leq -\mu \|g_{k+1}\|^2$, the proof is completed. we propose the following mild assumptions to get aiming at achieving global convergence.

Assumption: [22], [23]

I. The level set $\delta = \{x \,|f(x) \leq f(x_0)\}$ is bounded.

II. In some neighborhood N of δ, f is continuously differentiable, and its gradient is lipschitz continuous with Lipschitz constant $\delta > 0$, i.e.,

$$\|g(x) - g(y)\| \leq \delta\|x - y\| \;\; \forall \, x, y \in \delta$$

from the above assumptions, that there exists a positive constant b such that.

$$\|g(x)\| \leq b \;\forall x \in \delta. \quad (14)$$

following is established for the global convergence of the new algorithm based on the discussion above.

Lemma 3.1: [24] Let assumptions (I)–(II) hold. Consider the methods (1) and (2), where d_{k+1} is a descent direction and α_k satisfies the standard wolfe line search. If,

$\sum_{k\geq 1} \frac{1}{\|d_{k+1}\|^2} = \infty$. Then, $\lim_{k\to\infty} inf \; \|g_{k+1}\| = 0$

based on the above discussion the global convergence of new algorithm is established as follows.

Theorem 2.3: if assumptions (I) and (II) are valid and Algorithm 2.1 generates the corresponding sequences of $\{x_k\}$, $\{d_k\}$, $\{g_k\}$, $\{\alpha_k\}$, then we get to the conclusion that,

$$\lim_{k\to\infty} inf \; \|g_{k+1}\| = 0$$

Proof: from (2) and (9), we have,

$$\|d_{k+1}\| \leq \|g_{k+1}\| + \left|\frac{g_{k+1}^T y_k}{d_k^T y_k} - (1-\theta)\frac{y_k^T v_k \; g_{k+1}^T v_k}{v_k^T v_k \; d_k^T y_k}\right| \| d_k\| \quad (15)$$

$$\|d_{k+1}\| \leq \|g_{k+1}\| + \left(\left|\frac{g_{k+1}^T y_k}{d_k^T y_k}\right| + (1-\theta)\left|\frac{y_k^T v_k \; g_{k+1}^T v_k}{v_k^T v_k \; d_k^T y_k}\right|\right) \| d_k\| \quad (16)$$

since $g_{k+1}^T v_k \leq \alpha_k d_k^T y_k$ and by using (14), also from lipschitz condition $\|y_k\| \leq L\|v_k\|$ and $g_{k+1}^T y_k \leq L g_{k+1}^T d_k$ where $L > 0$, we have,

$$\|d_{k+1}\| \leq M + (L + (1-\theta)\alpha_k \, L)\| \, d_k\| \quad (17)$$

since, $\|v_k\| = \|x - x_k\|$, $D = max\{\|x - x_k\|\}, \forall \, x, x_k \in R\}$. Hence (17) becomes $\|d_{k+1}\| \leq M + (+(1-\theta)\alpha_k \,)DL = \beta$

$$\Rightarrow \sum_{k\geq 1} \frac{1}{\|d_{k+1}\|^2} \geq \sum_{k\geq 1} \frac{1}{\beta^2} = \infty$$

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

by using lemma (3.1), we get $\lim_{k\to\infty} inf\|g_k\| = 0$. Which completes the proof.

4. NUMERICAL RESULTS FOR UNCONSTRAINED OPTIMIZATION

This part focuses on evaluating how well our approach for resolving optimization issues works in practice. We contrast HS method with our novel algorithm. Well-known test functions [25] with varying dimensionality were used in the comparison testing. Fortran 95 is used to write all program lines. The cubic interpolation method utilized in the line search routine utilised function and gradient values. The outcomes shown in Table 1 depend on the number of iterations (NI) and functions (NF). Table 2 experimental findings support the claim that the new method outperforms the HS algorithm in terms of number of iterations, and or number of functions.

Table 1. The results for the HS and New CG methods on the tested problems

Test function	n	HS		New CG		(Continued) Test function	n	HS		New CG	
		NI	NF	NI	NF			NI	NF	NI	NF
G-Cantrel	4	22	159	21	150	Miele	4	28	85	28	85
	10	22	159	21	150		10	31	102	28	85
	100	22	159	22	163		100	33	114	31	102
	500	23	171	22	163		500	40	146	39	139
	1,000	23	171	22	163		1,000	46	176	39	139
	5,000	28	248	23	176		5,000	54	211	39	139
G-Wolfe	4	11	24	11	24	G-Wood	4	30	68	26	61
	10	32	65	32	65		10	30	68	26	61
	100	49	99	49	99		100	30	68	26	61
	500	52	105	52	105		500	30	68	26	61
	1,000	70	141	60	121		1,000	30	68	28	65
	5,000	165	348	165	345		5,000	30	68	28	65
Powell	4	37	102	33	91	Non-Diagonal	4	24	64	24	64
	10	37	102	33	91		10	26	72	26	72
	100	40	117	36	106		100	29	79	29	79
	500	44	136	36	106		500	F	F	29	82
	1,000	44	136	36	106		1,000	29	79	29	79
	5,000	44	136	36	106		5,000	30	81	30	81
Rosen	4	30	83	30	83	OSP	4	8	45	8	44
	10	30	83	30	83		10	13	58	13	60
	100	30	83	30	83		100	49	185	49	173
	500	30	83	30	83		500	112	353	107	341
	1,000	30	83	30	83		1,000	156	475	150	450
	5,000	30	83	30	83		5,000	256	774	256	765
						Total		2,147	6,747	2,004	6,181

Notes:
I. The letter F in the previous table denotes that a technique to determine the minimum was unsuccessful.
II. We assumed that the HS failure result was worth twice as much as the new CG findings.

Table 2. The percentage of improvement between the HS and New CG methods

Tools	HS	New CG
NI	100%	93.3395%
NF	100%	91.6111%

Algorithm 2.1 applications for training neural networks

This section presents the experimental numerical findings used to analyze and compare the effectiveness of the traditional and novel CG methods for training neural networks. We specifically look into how the HS method performed in comparison to the new CG method throughout the course of the program's five times implementation. The conjugate gradient MATLAB neural network toolbox version 8.1 and MATLAB (2013a) are used to implement the methods. To reduce the value of the error's function, the network is trained until the mean squares of the errors are below the error goal. We evaluate all methods using the identical initial weights, which were generated at random from the range (0, 1) where the problems:

– Input P = [-1, -1, 2, 2 0, 5, 0, 5] and the target T = [-1, -1, 1, 1], the target error has been set to $1x10^{-20}$ and the maximum epochs to 1,000 as used.

– Input continuous trigonometric function $f(x) = sin(x) + cos(2x)$ where $x \in [0, \pi]$ and the target error has been set to $1x10^{-20}$ and the maximum epochs to 1,000 as used.

The results of the training methods are present in the Table 3 and Figures 1 to 4.

Table 3. Performance evaluation of new and conventional methods for training neural networks

Methods	No. Running	Epochs	CPU time (s)/Epoch	Gradient
Problem 1				
	1	1,000	0:00:04	0.000376
	2	34	0:00:00	0.000239
Standard	3	41	0:00:00	0.000173
	4	21	0:00:00	0.000209
	5	38	0:00:00	0.000157
	1	8	0:00:01	0.0581
	2	3	0:00:00	0.109
New CG	3	5	0:00:00	0.0700
	4	2	0:00:00	0.971
	5	8	0:00:00	0.00142
Problem 2				
	1	296	0:00:02	0.000244
	2	1,000	0:00:04	0.00334
Standard	3	69	0:00:00	0.000183
	4	1,000	0:00:0	0.00219
	5	1,000	0:00:05	0.000374
	1	20	0:00:00	0.0861
	2	319	0:00:00	0.00291
New CG	3	36	0:00:00	0.00174
	4	4	0:00:00	0.356
	5	5	0:00:05	0.0937

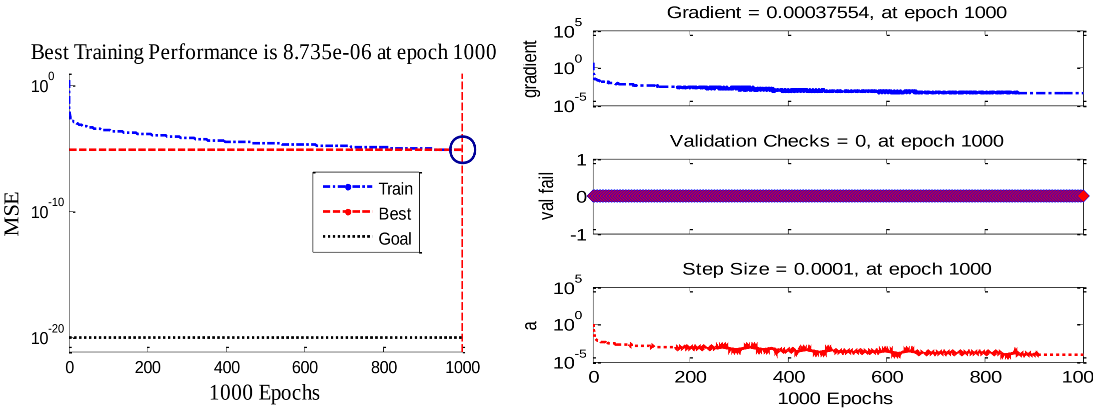

Figure 1. Performance of HS method for training neural networks using problem 1

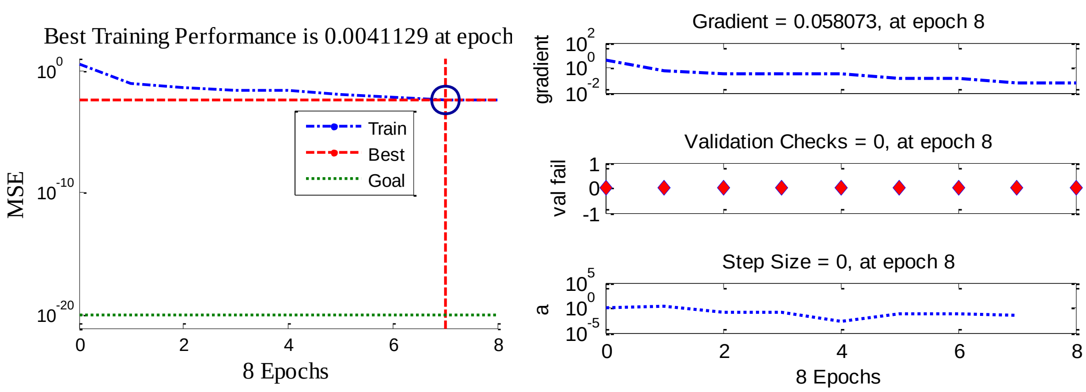

Figure 2. Performance of new method for training neural networks using problem 1

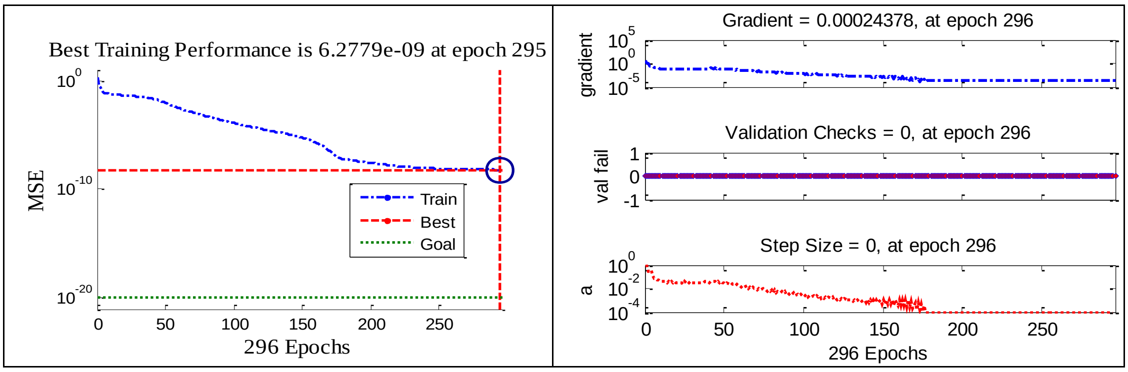

Figure 3. Performance of HS for training neural networks using problem 2

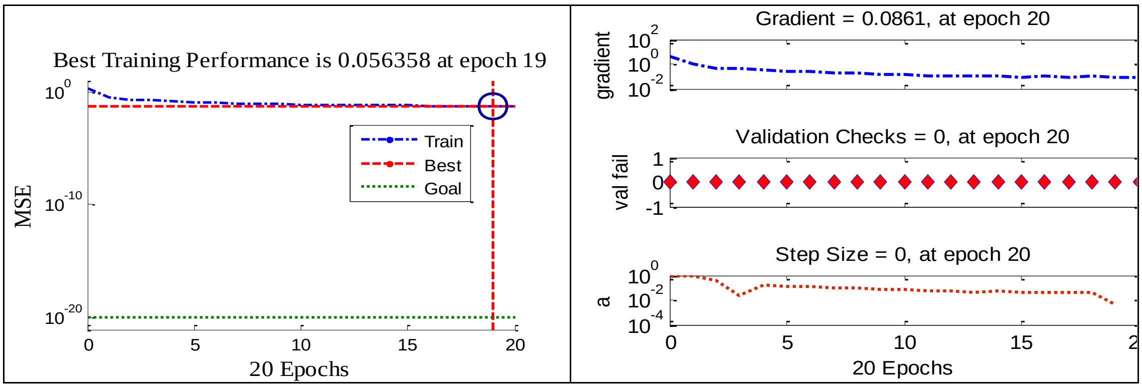

Figure 4. Performance of new method for training neural networks using problem 2

5. CONCLUSION

ANN has employed modified conjugate gradients since the 1980s with a variety of CG algorithms, including single modified and combinations of numerous CG techniques, according to preliminary study. The majority of the research was on increasing the effectiveness of learning algorithms during training. By using appropriate formulations and methods to improve the performance of all test data, this major objective will be accomplished. In addition, the review uses CG as a key component for fixing mistakes in the back-propagation process. In this paper, we presented a new CG method parameter based on the Perry parameter and a modified vector v_k^* based on the Barzilai and Borwein step size. The descent and sufficient descent requirements of the new method are established. Additionally, we investigate the global convergent property under accepted premises. The new method is more effective than the HS algorithm, as demonstrated by the numerical results on problems with low and high dimensionality. Finally, the new method's practical relevance for training neural networks is also investigated. When compared to some well-known algorithms, the strategy is utilized to increase the efficiency of neural networks.

REFERENCES

[1] L. Guanghui, H. Jiye, and Y. Hongxia, "Global convergence of the fletcher-reeves algorithm with inexact linesearch," *Applied Mathematics-A Journal of Chinese Universities*, vol. 10, no. 1, pp. 75–82, Mar. 1995, doi: 10.1007/BF02663897.

[2] Y. Liu and C. Storey, "Efficient generalized conjugate gradient algorithms, part 1: theory," *Journal of Optimization Theory and Applications*, vol. 69, no. 1, pp. 129–137, Apr. 1991, doi: 10.1007/BF00940464.

[3] R. Fletcher, "Function minimization by conjugate gradients," *The Computer Journal*, vol. 7, no. 2, pp. 149–154, Feb. 1964, doi: 10.1093/comjnl/7.2.149.

[4] M. R. Hestenes and E. Stiefel, "Methods of conjugate gradients for solving," *Journal of Research of the National Bureau of Standards*, vol. 49, no. 6, pp. 409–436, 1952.

[5] E. Polak and G. Ribiere, "Note sur la convergence de méthodes de directions conjuguées," *Revue française d'informatique et de recherche opérationnelle. Série rouge*, vol. 3, no. 16, pp. 35–43, May 1969, doi: 10.1051/m2an/196903R100351.

[6] A. Ibrahim and S. Shareef, "Modified conjugate gradient method for training neural networks based on logisting mapping," *The Journal of the University of Duhok*, vol. 22, no. 1, pp. 45–51, Nov. 2019, doi: 10.26682/sjuod.2019.22.1.7.

[7] S. Shareef and A. Ibrahim, "A new conjugate gradient for unconstrained optimization based on step size of Barzilai and Borwein," *Science Journal of University of Zakho*, vol. 4, no. 1, pp. 104–114, Jun. 2016, doi: 10.25271/2016.4.1.29.

[8] A. L. Ibrahim and M. G. Mohammed, "A new three-term conjugate gradient method for training neural networks with global convergence," *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 28, no. 1, p. 551, Oct. 2022, doi: 10.11591/ijeecs.v28.i1.pp551-558.
[9] I. M. Sulaiman, N. A. Bakar, M. Mamat, B. A. Hassan, M. Malik, and A. M. Ahmed, "A new hybrid conjugate gradient algorithm for optimization models and its application to regression analysis," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 23, no. 2, p. 1100, Aug. 2021, doi: 10.11591/ijeecs.v23.i2.pp1100-1109.
[10] A. Y. Al-Bayati and M. M. M. Ali, "New multi-step three-term conjugate gradient algorithms with inexact line searches," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 19, no. 3, p. 1564, Sep. 2020, doi: 10.11591/ijeecs.v19.i3.pp1564-1573.
[11] A. Perry, "A modified conjugate gradient algorithm," *Operations Research*, vol. 26, no. 6, pp. 1073–1078, 1978.
[12] M. Al-Baali, "Descent property and global convergence of the fletcher-reeves method with inexact line search," *IMA Journal of Numerical Analysis*, vol. 5, no. 1, pp. 121–124, 1985, doi: 10.1093/imanum/5.1.121.
[13] G. Zoutendijk, "Nonlinear programming, computational methods," *Integer nonlinear Program*, vol. 5, no. 2, pp. 37–86, 1970.
[14] J. Sun and J. Zhang, "Global convergence of conjugate gradient methods without line search," *Annals of Operations Research*, vol. 103, no. 1–4, pp. 161–173, 2001, doi: 10.1023/A:1012903105391.
[15] M. Negnevitsky, "Artificial intelligence: a guide to intelligent systems.," *Polyhedron*, vol. 123, pp. 334–343, 2005, [Online]. Available: www.pearsoned.co.uk.
[16] C. H. Wu and H. L. Chen, "Gene classification artificial neural system," *International Journal on Artificial Intelligence Tools*, vol. 04, no. 04, pp. 102–107, Dec. 1995, doi: 10.1142/s0218213095000255.
[17] S. C. Kothari and H. Oh, "Neural networks for pattern recognition," in *Advances in Computers*, vol. 37, no. C, 1993, pp. 119–166.
[18] M. G. Mohammed and A. I. Melhum, "Implementation of HOG Feature Extraction with Tuned Parameters for Human Face Detection," *International Journal of Machine Learning and Computing*, vol. 10, no. 5, pp. 654–661, Oct. 2020, doi: 10.18178/ijmlc.2020.10.5.987.
[19] A. K. Goel, S. C. Saxena, S. Bhanot, and A. M. Perceptron, "Modified functional link artificial neural network," *International Journal of Computer and Information Engineering (IJEECS)*, vol. 1, no. 1, pp. 22–30, 2006.
[20] A. M. Qasim, Z. F. Salih, and B. A. Hassan, "A new conjugate gradient algorithms using conjugacy condition for solving unconstrained optimization," *Indonesian Journal of Electrical Engineering and Computer Science (IJEECS)*, vol. 24, no. 3, p. 1647, Dec. 2021, doi: 10.11591/ijeecs.v24.i3.pp1647-1653.
[21] Barzilai, Jonathan, and M. B. Jonathan, "Two-point step size gradient methods," *IMA Journal of Numerical Analysis*, vol. 8, pp. 141–148, 1988.
[22] I. Jusoh, M. Mamat, and M. Rivaie, "A new edition of conjugate gradient methods for large-scale unconstrained optimization," *International Journal of Mathematical Analysis*, vol. 8, no. 45–48, pp. 2277–2291, 2014, doi: 10.12988/ijma.2014.44115.
[23] N. Andrei, "Hybrid conjugate gradient algorithm for unconstrained optimization," *Journal of Optimization Theory and Applications*, vol. 141, no. 2, pp. 249–264, May 2009, doi: 10.1007/s10957-008-9505-0.
[24] K. Sugiki, Y. Narushima, and H. Yabe, "Globally convergent three-term conjugate gradient methods that use secant conditions and generate descent search directions for unconstrained optimization," *Journal of Optimization Theory and Applications*, vol. 153, no. 3, pp. 733–757, Jun. 2012, doi: 10.1007/s10957-011-9960-x.
[25] N. Andrei, "An unconstrained optimization test functions collection," *Advanced Modelling and Optimization*, vol. 10, no. 1, pp. 147–161, 2008.

BIOGRAPHIES OF AUTHORS

Alaa Luqman Ibrahim is a teaching lecturer at the University of Zakho. He received a bachelor's degree in mathematics science from Duhok University and a master's degree in optimization from Zakho University. His current fields of interest area is Applied Mathematics and solve optimization models numerically and analytically in different fields. He can be contacted at email: alaa.ibrahim@uoz.edu.krd.

Mohammed Guhdar Mohammed is a teaching lecturer assistant at the University of Zakho. He received a bachelor's degree in computer science and a master's degree in artificial intelligence from Dohuk University. His current fields of interest are machine learning, algebra, bioinformatics, and computer vision. He can be contacted at email: mohammed.guhdar@uoz.edu.krd.