A new three-term conjugate gradient method for training neural networks with global convergence

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 Mar 28, 2022
Revised Jul 28, 2022
Accepted Aug 4, 2022

Keywords:

Artificial neural networks
Descent condition
Global convergent property
Sufficient descent condition
Three-term conjugate gradient
Unconstrained optimization

ABSTRACT

Conjugate gradient methods (CG) constitute excellent neural network training methods that are simplicity, flexibility, numerical efficiency, and low memory requirements. In this paper, we introduce a new three-term conjugate gradient method, for solving optimization problems and it has been tested on artificial neural networks (ANN) for training a feed-forward neural network. The new method satisfied the descent condition and sufficient descent condition. Global convergence of the new (NTTCG) method has been tested. The results of numerical experiences on some well-known test function shown that our new modified method is very effective, by relying on the number of functions evaluation and number of iterations, also included the numerical results for training feed-forward neural networks with other well-known method in this field.

Corresponding Author:

Moahmmed Guhdar Mohammed
Department of Computer Science, Faculty of Science, University of Zakho
Zakho, Kurdistan Region, Iraq
Email: mohammed.guhdar@uoz.edu.krd

1. INTRODUCTION

Artificial neural networks (ANNs) have been used for decades with major success in many applications related to machine learning [1]-[3] due to their outstanding ability to self-adapting and self-learning. They have been used in areas such as robotics, security, and self-driving cars very intensely. They are often more robust and accurate than other classification techniques due to their resilience in problem solving and parallel processing support [4], [5]. Although several different methods for training have been suggested one of which is feed forward neural networks (FNNs), this training pattern is one of the most known and widely used in many various areas and applications. Multi-layer (FNNs) are parallel computational models composed of densely interconnected, adaptive processing units, characterized by an inherent propensity for learning from experience and discovering new knowledge. Due to its excellent self-adaptation and self-learning ability, it gained early popularity in machine learning [1], [2], [6] and are often found it to be more efficient. The process of a FNN is depend on the below formula:

$$net_j^l = \sum_{i=1}^{N_{l-1}} w_{ij}^{l-1,l} y_i^{l-1} + b_j^l, y_i^l = f\left(net_j^l\right) \tag{1}$$

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 main conception of training a neural networks (NN) can be defined as a nonlinear optimization problem. Training a (NN) is to recursively modify its weights, in order to reduce the scale of the difference between the desired and actual output of all examples of the training set [7]. Therefore, the training process can be write mathematically as the reduction of the error function $E(w)$, which is determined by:

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

where w is a vector weights in R^n and the number of patterns used in the training set represented by P. [7]

CG method are probably one of the well-known iterative methods for efficiently training NN due to their simplicity. This method create a sequence of weights $\{w_k\}$, which is defined by:

$$w_{k+1} = w_k + \lambda_k p_k \tag{3}$$

where the iteration number k called epoch, the learning rate is $\lambda_k > 0$ and p_k is the search direction calculated by:

$$p_k = \begin{cases} -g_0 & if \quad k = 0 \\ -g_k + \beta_k p_{k-1}, & otherwise \end{cases} \tag{4}$$

where g_k is the gradient of E at w_k and β_k is a coefficient of (CG). In the literature, there have been presented several choices for β_k which give rise to distinct CG methods. Some of classical formula methods are fletcher-reeves (FR) method [8], the hestenes-stiefel (HS) method [9], and the polak-rivière (PR) method [10], which are determined respectively as follows:

$$p_k = -g_k + \left(\beta_k^{FR} = \frac{g_k^T g_k}{g_{k-1}^T g_{k-1}}\right) p_{k-1}, \text{ Fletcher and Reeves (FR)} \tag{5}$$

$$p_k = -g_k + \left(\beta_k^{HS} = \frac{g_k^T y_{k-1}}{p_{k-1}^T y_{k-1}}\right) p_{k-1}, \text{ Hestenes and Steifel (HS)} \tag{6}$$

$$p_k = -g_k + \left(\beta_k^{PR} = \frac{g_k^T y_{k-1}}{g_{k-1}^T g_{k-1}}\right) p_{k-1}, \text{ Polak and Ribiere (PR)} \tag{7}$$

Also, there are many studies to improve the parameter of conjugate gradient method such as

$$p_k = -g_k + \left(\beta_k^{New} = \frac{g_k^T g_k}{p_{k-1}^T y_{k-1}} - \mu \left(\frac{g_k^T g_k}{g_{k-1}^T g_{k-1}}\right)^2\right) p_{k-1}, \text{ [11]} \tag{8}$$

$$p_k = -g_k + \left(\beta_k^{New} = \beta_k^{HS} - \frac{\lambda_{k-1} \|p_{k-1}\|^2 g_k^T p_{k-1}}{\left(p_{k-1}^T y_{k-1}\right)^2}\right) p_{k-1}, \text{ [12]} \tag{9}$$

Furthermore, the design of CG-techniques had been studied by many researchers; for more details see, [13]-[15].

The three-term CG method is other important class of the CG methods [16], [17] presented a proposal of the three-term CG method by considering a descent modified (PRP and HS) CG method as (10).

$$p_k^{ZPRP} = \begin{cases} -g_k if k = 0 \\ -g_k + \frac{g_k^T y_{k-1}}{g_{k-1}^T g_{k-1}} p_{k-1} - \frac{g_k^T p_{k-1}}{g_{k-1}^T g_{k-1}} y_{k-1} if k \geq 1 \end{cases} \tag{10}$$

and,

$$p_k^{ZHS} = \begin{cases} -g_k if k = 0 \\ -g_k + \frac{g_k^T y_{k-1}}{p_{k-1}^T y_{k-1}} p_{k-1} - \frac{g_k^T p_{k-1}}{p_{k-1}^T y_{k-1}} y_{k-1} if k \geq 1 \end{cases} \tag{11}$$

In the same way, Ibrahim and Shareef in (2019) [18], propose a new computationally effective new class of the three-term CG method defined in the following:

$$p_k^{NTT-CG} = \begin{cases} -g_k if k = 0 \\ -g_k + \beta_k p_{k-1} - t_{k-1} \left(\frac{g_k^T p_{k-1}}{p_{k-1}^T y_{k-1}}\right) y_{k-1} if k \geq 1 \end{cases} \tag{12}$$

where $t_{k-1} = \gamma \frac{\|y_{k-1}\|}{\|s_{k-1}\|} + (1-\gamma) \frac{s_{k-1}^T y_{k-1}}{\|s_{k-1}\|^2}$, and the parameter β_k is given from normal CG (HS, PRP, and FR) method. Also, nowadays there are many researchers intensely working on developing the three term CG direction and its applications [19], [20].

This paper, offer a new three-term CG method (NTTCG) and apply it for training neural networks in section 2. Section three include the proof of the (descent and sufficient descent) conditions of the (NTTCG) method. Also, we demonstrate that the (NTTCG) method is globally convergent in section 3. Finally, presents our concluding and remarks in section 4.

2. DERIVATION (NEWTT) METHOD

In this section, proposed a new three-term CG method solve optimization problems and for training neural networks by using a modified vector y_{k-1}^* depend on the step size of Barzilai and Borwein [21]. Suppose $y_{k-1}^* = \frac{1}{\alpha_{k-1}^{BB}} y_{k-1} - \frac{\theta}{\alpha_{k-1}^{BB}} y_{k-1}$, where $\theta \in (0,1)$and $\alpha_{k-1}^{BB} = \frac{y_{k-1}^T v_{k-1}}{y_{k-1}^T y_{k-1}}$. So, we have:

$$y_{k-1}^* = (1-\theta) \frac{y_{k-1}^T y_{k-1}}{y_{k-1}^T v_{k-1}} y_{k-1}, \tag{13}$$

Now, by replacing $y_{k-1}$by y_{k-1}^* in numerator the third term of (11) we obtained a new search direction named as (NEWTT):

$$p_k^{NEWTT} = \begin{cases} -g_k & if \quad k=0 \\ -g_k + \frac{g_k^T y_{k-1}}{p_{k-1}^T y_{k-1}} p_{k-1} - (1-\theta) \frac{y_{k-1}^T y_{k-1} g_k^T p_{k-1}}{y_{k-1}^T v_{k-1} p_{k-1}^T y_{k-1}} y_{k-1}, & Otherwise \end{cases} \tag{14}$$

REMARK: Note that when the search direction is exact $(g_k^T p_{k-1} = 0)$ or if $\theta = 1$, the search direction in (14), reduces to the HS method.

2.1. Outlines of ($NEWTT$) method for solving unconstrained optimization

In this section, the outlines of the new method for solving unconstrained optimization problems is stated.

Step 1. Given $x_0 \in R^n$ and set$d_0 = -g_0$, $k = 0$.

Step 2. If $\|g_k\| = 0$ then stop, else continue to Step 3.

Step 3. Set the λ_k by cubic line search method to minimize$f(x_{k+1})$.

Step 4. Set $x_{k+1} = x_k + v_k$.

Step 5. Computeg_{k+1}, if $\|g_{k+1}\| \leq 10^{-5}$ stop.
else go to Step 6.

Step 6. Determine p_{k+1} by using [(14)].

Step 7. If $\|g_{k+1}\|^2 \leq \frac{|g_k^T g_{k+1}|}{0.2}$ go to step 2,
else set $k = k + 1$ and go to step 3.

2.2. Outlines of ($NEWTT$) method for training neural networks

This section, shows the outlines of (NEWTT) method for training neural networks to update the weights $w_k$by using the new search direction p_k from (14).

Step 1. Initiatew_0, $gol = E_G$ and k_{max}, set $k = 0$.

Step 2. Evaluate E_k and $g_k = \nabla E(w_k)$.

Step 3. If $E_k < E_G$, or $\|g_k\| \leq \varepsilon$, return $w^* = w_k$ and $E^* = E_k$, then stop.
else Compute $v_k = w_{k+1} - w_k$ and $y_k = g_{k+1} - g_k$.

Step 4. Compute p_k using equation [(14)].

Step 5. Determine λ_k to minimize $E(w_{k+1})$.

Step 6. Determine $w_{k+1} = w_k + \lambda_k p_k$ and $k = k + 1$

Step 7. If $k > k_{max}$ return "Error Goal not met"
else continue to step 2.

3. DESCENT AND SUFFICIENT DESCENT CONDITIONS AND GLOBAL CONVERGENCE OF ($NEWTT$) METHOD

This section, shows that the new (NEWTT) three-term method sati.sfies the (descent and the sufficient descent) properties. In addition to that, we show the new (NEWTT) method is globally convergent as mentioned in the following theorems:

Theorem 1. Presume that $\{w_k\}$ is generated by (3), then the p_k in (14) satisfies the condition, $g_k^T p_k \leq 0$.
Proof: From (14), we have if $k = 0$, we have $p_0^T g_0 = -\|g_0\|^2 \leq 0$.
suppose that $p_{k-1}^T g_{k-1} \leq 0, \forall k = 1,2, \dots, i$.
Now, we must prove in iteration $(k+1)$ is a descent direction. By multiplying (14) byg_k^T, we have;

$$g_k^T p_k = -g_k^T g_k + \frac{g_k^T y_{k-1}}{p_{k-1}^T y_{k-1}} g_k^T p_{k-1} - (1-\theta) \frac{y_{k-1}^T y_{k-1} g_k^T p_{k-1}}{y_{k-1}^T v_{k-1} p_{k-1}^T y_{k-1}} g_k^T y_{k-1} \tag{15}$$

If the search direction is exact, then the equation above equation is satisfied the descent condition.

i.e. $g_k^T p_k = -g_k^T g_k \leq 0$.

However, If the direction p_k is not exact. (i.e.) $g_k^T p_{k-1} \neq 0$. We conclude:

$$-g_k^T g_k + \frac{g_k^T y_{k-1}}{p_{k-1}^T y_{k-1}} g_k^T p_{k-1} \leq 0, \tag{16}$$

because (16) is a search direction of HS method achieve the descent condition.
Since from Lipschize condition we have:

$$g_k^T y_{k-1} \leq L g_k^T p_{k-1} \text{ where } L > 0. \tag{17}$$

It is clearly $1-\theta > 0$ and $y_{k-1}^T y_{k-1}$, $(g_k^T p_{k-1})^2$, $y_{k-1}^T v_{k-1} \wedge p_{k-1}^T y_{k-1}$ are non-negative.
So, we get to:

$$g_k^T p_k \leq -g_k^T g_k + \frac{g_k^T y_{k-1}}{p_{k-1}^T y_{k-1}} g_k^T p_{k-1} - (1-\theta) \frac{y_{k-1}^T y_{k-1} (g_k^T p_{k-1})^2}{y_{k-1}^T v_{k-1} p_{k-1}^T y_{k-1}} \leq 0 \tag{18}$$

The proof is completed.
Theorem 2. Presume that $\{w_k\}$ is generated sequence by (3) where p_k is define in (14) and λ_k is obtained from strong Wolfe conditions, then the p_k satisfies:

$$g_k^T p_k \leq -c\|g_k\|^2$$

Proof: From (15). we have:

$$g_k^T p_k = -g_k^T g_k + \frac{g_k^T y_{k-1}}{p_{k-1}^T y_{k-1}} g_k^T p_{k-1} - (1-\theta) \frac{y_{k-1}^T y_{k-1} g_k^T p_{k-1}}{y_{k-1}^T v_{k-1} p_{k-1}^T y_{k-1}} g_k^T y_{k-1} \tag{19}$$

It is clearly from (16), the first two terms of the above equation are non- positive and by using (17), we obtained:

$$g_k^T p_k \leq -(1-\theta) \frac{y_{k-1}^T y_{k-1} (g_k^T p_{k-1})^2}{y_{k-1}^T v_{k-1} p_{k-1}^T y_{k-1}} \tag{20}$$

Therefore, (19) can be written as:

$$g_k^T p_k \leq -\|g_k\|^2$$

So, we have:

$$g_k^T p_k \leq -c\|g_k\|^2 \text{ , where } c = (1-\theta) \frac{y_{k-1}^T y_{k-1} (g_k^T p_{k-1})^2}{y_{k-1}^T v_{k-1} p_{k-1}^T y_{k-1} \|g_k\|^2}.$$

Therefore, our new method is sufficient descent.
Now, we need the following assumptions [22], [23] to show the global convergence of (NTTCG) method.
Assumptions:

1. The level set $S = \{w: w \in R^n, E(w) \leq E(w_0)\}$ is bounded. i.e. $\exists B > 0$, such that

$$\|w\| \leq B, \forall w \in S \tag{21}$$

2. In a neighborhood $\Omega \in S$, E is differentiable and its gradient g is Lipschitz continuous, i.e. $\exists$ $L > 0$, such that

$$\|g(w) - g(w_k)\| \leq L\|w - w_k\|, \forall w, w_k \in \Omega \tag{22}$$

From Assumptions 1 and 2, $\exists M > 0$, such that:

$$\|g(w)\| \leq M, \forall w \in S. \tag{23}$$

We can rewrite (21) in the following manner:

$$y_{k-1}^T v_{k-1} \geq L\|v_{k-1}\|, \tag{24}$$

Lemma 1 [24]. Suppose that the Assumptions 1 and 2 holds and the $\{w_k\}$ generated by (3) and (14), where p_k, satisfy the descent condition and λ_k set by strong Wolfe conditions.

$$\sum_{k\geq 1} \frac{1}{\|p_k\|^2} = \infty. \tag{25}$$

Then,

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

Theorem 3. Suppose that Assumptions 1 and 2 holds. If any iteration of the (3) and (14), and λ_k satisfies the strong Wolfe conditions, then:

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

Proof: From (14), we have, we have:

$$\|p_k^{NEWTT}\| \leq \|g_k\| + \left|\frac{g_k^T y_{k-1}}{p_{k-1}^T y_{k-1}}\right| \|p_{k-1}\| + (1-\theta)\left|\frac{y_{k-1}^T y_{k-1} g_k^T p_{k-1}}{y_{k-1}^T v_{k-1} p_{k-1}^T y_{k-1}}\right| \|y_{k-1}\|, \tag{27}$$

since $g_k^T p_{k-1} \leq p_{k-1}^T y_{k-1}$ and by using (17), we get to:

$$\|p_k^{NEWTT}\| \leq \|g_k\| + |L|\|p_{k-1}\| + (1-\theta)\left|\frac{y_{k-1}^T y_{k-1}}{y_{k-1}^T v_{k-1}}\right| \|y_{k-1}\|, \tag{28}$$

Also, Now, from Lipschitz Condition $\|y_{k-1}\| \leq L\|v_{k-1}\|$ and by using equation (24), we have:

$$\|p_k^{NEWTT}\| \leq M + |L|\|p_{k-1}\| + (1-\theta)\frac{L^2}{\|v_{k-1}\|}, \tag{29}$$

Since, $\|v_k\| = \|w - w_k\|$, $D = max\{\|w - w_k\|\}, \forall w, w_k \in R\}$.
Hence (29) becomes:

$$\|p_k^{NEWTT}\| \leq \left[M + \frac{LD}{\lambda_{k-1}} + (1-\theta)\frac{L^2}{D}\right] = \beta$$

$\Rightarrow \sum_{k\geq 1}^{\infty} \frac{1}{\|p_k^{NEWTT}\|^2} \geq \sum_{k\geq 1}^{\infty} \frac{1}{\beta^2} = \infty \quad \Rightarrow \sum_{k\geq 1}^{\infty} \frac{1}{\|p_k^{NEWTT}\|^2} = \infty$ By using lemma (1), we get $\lim_{k\to\infty} inf \|g_k\| = 0$. which completes the proof.

4. EXPERIMENTAL RESULTS

4.1. Numerical results of ($NEWTT$) method for optimization problems

This section is present the implementation of the (NEWTT) method for solving unconstrained optimization problems. We compare our new method against ZHS method. The comparative includes some well-known nonlinear test problems with several dimensional and the test problems are selected from [25]. All program lines are written in FORTRAN 95. The line search method was a cubic interpolation. The numerical results given in Table 1 specifically depend on the number of functions (NOF) and the number iterations (NOI) while Table 2 present the experimental results to confirm that the (NTTCG) method is superior to ZHS method.

Table 2, show the efficiency of the (NEWTT) compare with the (ZHS) method. We also note that the NOI and NOF of the standard method are about 100%. This means that the (NEWTT) method improved compared to the ZHS method by about 21.7715% in NOI and 17.2744% in NOF. Finally, the overall rate of improvement is 26.351% in (NEWTT).

Table 1. The numerical results for (NEWTT) and ZHS methods on the test functions

Test Function	n	$ZHS-CG$	$NEWTT$		
		NOI	NOF	NOI	NOF
Powell	10	38	100	33	92
	50	41	117	33	92
	100	41	117	33	92
	500	43	121	33	92
	1000	43	121	33	92
	5000	43	121	33	92
Mile	10	46	157	32	105
	50	53	190	36	123
	100	53	190	36	123
	500	53	190	39	140
	1000	57	209	44	160
	5000	60	225	44	160
Shallon	10	8	21	8	21
	50	8	21	8	21
	100	8	21	8	21
	500	8	21	8	21
	1000	9	24	9	24
	5000	9	24	9	24
Cubic	10	15	44	13	37
	50	15	44	13	37
	100	15	44	13	37
	500	F	F	13	37
	1000	16	46	13	37
	5000	F	F	13	37
Central	10	22	155	22	159
	50	26	204	22	159
	100	26	204	22	159
	500	26	204	23	171
	1000	28	231	23	171
	5000	32	292	28	248
Wood	10	29	67	29	67
	50	29	67	29	67
	100	29	67	29	67
	500	29	67	29	67
	1000	29	67	29	67
	5000	29	67	29	67
Sum	10	6	34	6	34
	50	11	65	11	60
	100	14	85	14	81
	500	19	106	21	122
	1000	25	139	23	126
	5000	139	174	31	144
Rosen	10	32	86	30	83
	50	33	88	30	83
	100	33	88	30	83
	500	33	88	30	83
	1000	33	88	30	83
	5000	33	88	30	83
Total	1479	5137	1157	4251	

NOTES:

- The letter F in above table refers to that a method is failed to find the minimum
- We considered the failure result in ZHS is a twice value of (NEWTT) results

Table 2. The percentage of improvement between the ($NEWTT$) and (ZHS) method

Tools	ZHS	$NEWTT$
NOI	100%	78.2285
NOF	100%	82.7526

4.2. Numerical results of ($NEWTT$) method for training neural networks

This section tests the implementation of the NEWTT method for training neural networks in classical artificial intelligence problems (continuous function approximation). We search the execution of the (NEWTT) method by comparing it against to the ZHS method. The program steps are implemented five times by using MATLAB (2013a), neural network toolbox (version 8.1) for conjugate gradient.

Problem: suppose the approximation of the function as:

$$f(x) = sin(x) + cos(2x) \text{ , where } x \in [0, \pi].$$

The network is trained to approximate the function and trained until the mean squares errors becomes less than the goal error1e-15 within the limit of 5000 epochs. Table 3 shows the performance comparison of (NEWTT) method with ZHS. The new method displays excellent likelihood (100%) of successful training for network by using the same initial weights. Thus, computational cost is possibly the most appropriate indicator for measuring the efficiency of the methods. The NEWTT method performance is better than the ZHS method in terms of the time, number of epochs, gradient, and step size as shown in Figures 1 and 2.

Table 3. Comparing the performance of new method with ZHS method for training neural network

Methods	No. Running	Epochs	CPU time(s)/Epoch	Gradient	Step size
	1	214	0:00:01	0.000175	0.00
	2	120	0:00:39	0.000201	0.00
ZHS	3	169	0:00:55	0.000223	0.00
	4	114	0:00:34	0.000199	0.00
	5	300	0:01:32	0.00347	0.00101
	1	168	0:00:01	0.000133	0:00
	2	121	0:00:36	0.000206	0.00
NEWTT	3	70	0:0021	0.000170	0.00
	4	66	0:00:19	0.000163	0.00
	5	300	0:01:29	0.00346	0.00102

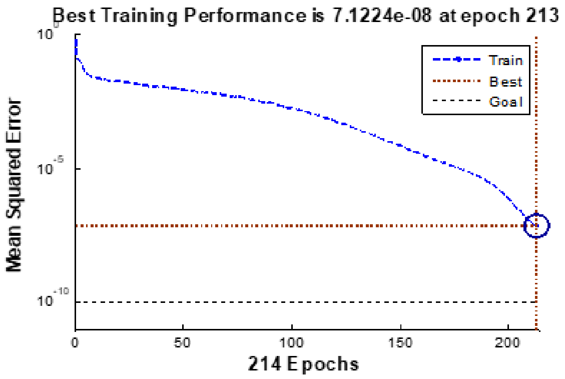

Figure 1. Performance of ZHS method for training neural networks

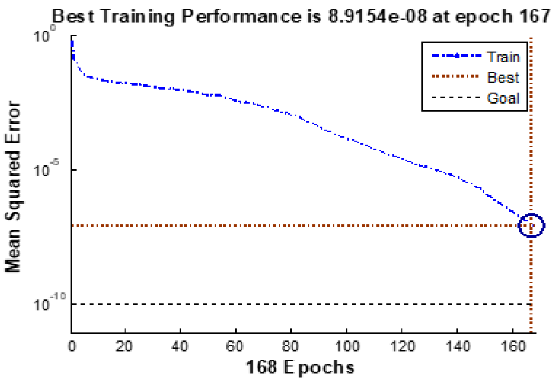

Figure 2. Performance of NEWTT method for training neural networks

5. CONCLUSION

This paper offers a new three-term CG (NTTCG) method and apply it for training neural networks. Also, we proved that the NTTCG method is globally convergent in addition to the decent condition and sufficient decent condition. Depend on the numerical results, it can be clearly observed that the (NTTCG) method is more active in terms of a time, number of epochs, step size and gradient than other methods such as ZHS, which is providing a faster training curve. Finally, the practical application of the (NEWTT) method in risk optimization is also explored. It's efficiency in solving portfolio selection problem was outstanding as it solves the problem with less function evaluations, iteration and CPU time compared with other methods.

REFERENCES

[1] C. H. Wu, H.-L. Chen, and S.-C. Chen, "Gene classification artificial neural system," *International Journal on Artificial Intelligence Tools*, vol. 4, no. 04, pp. 501–510, 1995, doi: 10.1142/S0218213095000255.

[2] C. M. Bishop, "*Neural networks for pattern recognition*," Oxford university press, 1995, doi: 10.1201/9781420050646.ptb6.

[3] A. Hmich, A. Badri, and A. Sahel, "Automatic speaker identification by using the neural network," in 2011 *International Conference on Multimedia Computing and Systems*, 2011, pp. 1–5, doi: 10.1109/ICMCS.2011.5945601.

[4] B. Lerner, H. Guterman, M. Aladjem, and I. hak Dinstein, "A comparative study of neural network-based feature extraction paradigms," *Pattern Recognition Letters*, vol. 20, no. 1, pp. 7–14, 1999, doi: 10.1016/S0167-8655(98)00120-2.

[5] C. J. Spoerer, T. C. Kietzmann, J. Mehrer, I. Charest, and N. Kriegeskorte, "Recurrent neural networks can explain flexible trading of speed and accuracy in biological vision," *PLoS computational biology*, vol. 16, no. 10, p. e1008215, 2020, doi: 10.1371/journal.pcbi.1008215.

[6] 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, 2020, doi: 10.18178/ijmlc.2020.10.5.987.

[7] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, “*Learning internal representations by error propagation*,” California Univ San Diego La Jolla Inst for Cognitive Science, 1985, doi: 10.21236/ADA164453.

[8] R. Fletcher and C. M. Reeves, “Function minimization by conjugate gradients,” *The computer journal*, vol. 7, no. 2, pp. 149–154, 1964, doi: 10.1093/comjnl/7.2.149.

[9] 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, p. 409, 1952, doi: 10.6028/jres.049.044.

[10] E. Polak and G. Ribiere, “Note sur la convergence de méthodes de directions conjuguées,” ESAIM: *Mathematical Modelling and Numerical Analysis-Modélisation Mathématique et Analyse Numérique*, vol. 3, no. R1, pp. 35–43, 1969, doi: 10.1051/m2an/196903R100351.

[11] A. L. Ibrahim and S. G. Shareef, “Modified conjugate gradient method for training neural networks based on logistic mapping,” *Journal of Duhok University*, vol. 22, no. 1, pp. 45–51, 2019, doi: 10.26682/sjuod.2019.22.1.7.

[12] S. G. Shareef and A. L. 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, 2016, doi: 10.25271/2016.4.1.29.

[13] 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*, vol. 24, no. 3, pp. 1654–1660, 2021, doi: 10.11591/ijeecs.v24.i3.pp1647-1653.

[14] 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*, vol. 23, no. 2, pp. 1100–1109, 2021, doi: 10.11591/ijeecs.v23.i2.pp1100-1109.

[15] N. S. Mohamed, M. Mamat, M. Riyadi, and S. H. M.Shaharudin, “A new hyhbrid coefficient of conjugate gradient method,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 18, no. 3, pp. 1454–1463, doi: 10.11591/ijeecs.v18.i3.pp1454-1463.

[16] L. Zhang, W. Zhou, and D. Li, “Some descent three-term conjugate gradient methods and their global convergence,” *Optimisation Methods and Software*, vol. 22, no. 4, pp. 697–711, 2007, doi: 10.1080/10556780701223293.

[17] L. Zhang, W. Zhou, and D.-H. Li, “A descent modified polak–ribière–polyak conjugate gradient method and its global convergence,” *IMA Journal of Numerical Analysis*, vol. 26, no. 4, pp. 629–640, 2006, doi: 10.1093/imanum/drl016.

[18] A. L. Ibrahim and S. G. Shareef, “A new class of three-term conjugate gradient methods for solving unconstrained minimization problems,” *General Letters in Mathematics*, vol. 7, no. 2, pp. 79-86, 2019, doi: 10.31559/glm2019.7.2.4.

[19] A. S. Ahmed, H. M. Khudhur, and M. S. Najmuldeen, “A new parameter in three-term conjugate gradient algorithms for unconstrained optimization,” *Indones. J. Electr. Eng. Comput. Sci*, vol. 23, no. 1, 2021, doi: 10.11591/ijeecs.v23.i1.pp338-344.

[20] A. Y. Al-Bayati and M. M. Ali, “New multi-step three-term conjugate gradient algorithms with inexact line searches,” *Indonesian Journal of Electrical Engineering and Computer Science*, vol. 19, no. 3, pp. 1564–1573, 2020, doi: 10.11591/ijeecs.v19.i3.pp1564-1573.

[21] J. Barzilai and J. M. Borwein, “Two-point step size gradient methods,” *IMA journal of numerical analysis*, vol. 8, no. 1, pp. 141-148, 1988, doi: 10.1093/imanum/8.1.141.

[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. 46, 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, 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, 2012, doi: 10.1007/s10957-011-9960-x.

[25] N. Andrei, “An unconstrained optimization test functions collection,” *Adv. Model. Optim*, 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 reached via email at mohammed.guhdar@uoz.edu.krd.