Date of publication JUN-30, 2025, date of current version JAN-03, 2025.
www.computingonline.net / computing@computingonline.net

Print ISSN 1727-6209
Online ISSN 2312-5381
DOI 10.47839/ijc.24.2.4017

Malicious Node Detection in Wireless Sensor Networks Using Neural Networks

LARA DARWISH[1], MOHAMMAD NASSR[1], MOHAMMAD ANBAR[1], HAMID ALI ABED ALASADI[2,3]

[1]Department of telecommunication technology engineering, University of Tartous, Tartous, Syria 2
[2]Computer Science Department, CEPS, Basrah University, Basrah, 61004, Iraq
[3]Dept. of Communications Engineering, Iraq University College, Basrah, Iraq

Corresponding author: Hamid Ali Abed Alasadi (e-mail: 865.hamid@gmail.com; hamid.abed@uobasrah.edu.iq).

ABSTRACT Throughout the past decade, wireless sensor networks (WSNs) have become a focus of observation in the wireless and mobile computing research community. The WSN has many uses, and applications from inside (starting from measuring temperature, pressure, humidity, and similar application inside the home) to outside (extension on the tactical battleground). Due to the distributed nature and spread in distant region, these networks are prone to many security attacks, which in turn negatively influences performances of these networks. Therefore, securing wireless sensor networks against these threats is becoming more and more important. A malicious node within the network is one of these threats. In this study, a procedure for detecting a malicious node in the network using the NS-2 simulator is presented. The proposed method uses Artificial Neural Network (ANN), built using MATLAB, for the purpose of prediction. Training, validation and testing have been performed on a dataset containing throughput, power consumption and time delay as inputs of the used ANN. Experimental results show the efficiency of the proposed method in security threats detection.

KEYWORDS Neural Networks (NN); Wireless Sensor Network WSN; malicious node; throughput; power consumption; delay.

I. INTRODUCTION

Wireless Sensor Networks (WSNs) consist of sensor nodes. The nodes in WSN have the ability of sensing physical phenomena that happen around them. A specific sensor node might be capable of sensing temperature, pressure, or the movement of any moving object around it. Sensors can also sense colors and vibrations which take place around them.

Sensor nodes can participate with each other in collecting the sensed and acquired data. The collected data are converted into digital signals to process the physical phenomena which occur close by the sensors. A sensor node could behave both as data originator and data router [1].

Sensor networks are classified into two types:

- Stationary Sensor Networks: All the nodes in this type of network are fixed, i.e., none of the nodes in the network move.
- Mobile Sensor Networks: In this type, network architecture has two main classes. The first one is a single-hop network architecture, and the other one is a multi-hop network architecture [2], as shown in Figures 1 and 2. Few or, in some cases, all the nodes in the network are mobile.

There are numerous uses of Mobile Sensor Networks. One of the most famous uses of Mobile Sensor Networks is deploying these sensors over the oceans. There are many types of mobile ocean networks available, suitable for different applications. In addition to traditional platforms such as ships and coastal submarine, these used for unmanned mobile monitoring, such as autonomous underwater vehicles. Ocean mobile sensor network is an extension of wireless sensor network in the ocean. Generally, it includes network parts deployed on the sea surface and network parts deployed underwater. The most widely used surface wireless mobile sensor network uses radio waves for communication and networking, which can be used to monitor wind direction, wave height, tide, water temperature, light, water pollution and other information related to the ocean [3]. Ocean's waves and currents cause the nodes to move continuously.

WSNs play a role in smart cities, smart grids, and smart healthcare systems due to the rapid advancement of Internet. Moreover, Underwater Sensor Networks (USNs) have enormous applications extending from port security to observation underwater pipelines. Recently, these networks are used and utilized in fish farms making these networks more popular. WSNs have spread and played an important role in many fields, from military applications to monitoring forests (fires that occur in them) and buildings [4].

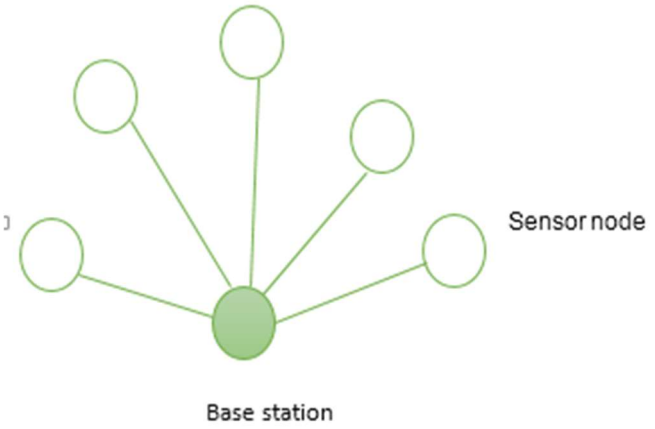


Figure 1. Single-hop network

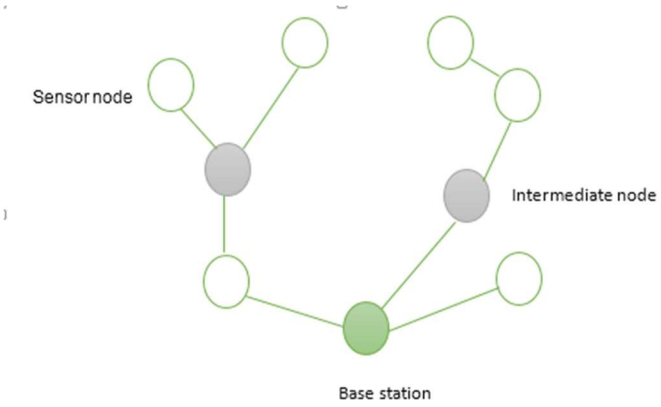


Figure 2. Multi-hop network

WSNs design depends on the application for which the network will be used. Moreover, it is related to several factors such as the environment, application design goals, hardware cost, and operating system [5].

Each sensor node contains sensing unit (sensor and analog to digital converter), processing unit (processor and storage), communication unit (transceiver), and power supply unit. Structure of each node contains antenna for sending receiving and data packets [6], see Figure 3.

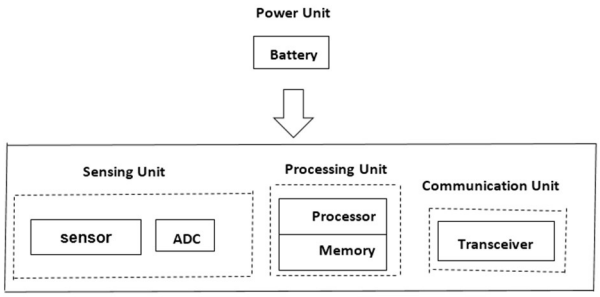


Figure 3. Sensor node structure

It is necessary that sensor network nodes have the ability to self-organize and share the network information. This is because in general, sensor networks are deployed randomly in the environment to be monitored [7].

Wireless sensor network are frequently deployed in an antagonistic condition and the enemy because of the requirements, for example, battery lifetime, could effectively imperil work without human supervision, singular node, and smaller memory space and constrained processing capacity. Security in WSN has been a standout amongst the most vital subjects in the WSN research network. [8].

II. RESEARCH OBJECTIVE

The purpose of the study is to detect if there is a malicious node in WSN network. Performance of the network may be enhanced by discovering if there is a malicious node.

It is important to say that WSN security is vital in protecting client data and keeping shared data secure. It is important also to ensure reliable access and network performance as well as protection against threats.

III. RESEARCH METHODOLOGY AND MATERIALS

This research follows an analytical approach. It is based on studying different related works such as [9-13] and comparing with them. The basic idea is discovering the existence of the malicious node based on Neural Network (NN) approach.

The empirical study in this research is conducted using NS-2 network simulator. NS-2 is used for collecting the WSN data in the first stage of the study. The results obtained from the first stage are used as input for NN simulation using MATLAB. The final objective is to determine if there is the malicious node in the network.

A. ARTIFICIAL NEURAL NETWORKS (ANN)

Artificial Neural Networks acquire knowledge by training and store it based on weights.

For training NN, a set of input data is entered besides a set of target data (expected data) [14].

Neural networks are trained to execute sophisticated functions in diversity of applications involving pattern recognition, identification, classification, speech, vision and control systems [15].

The general structure of ANN is created of an input layer, a hidden layer, and an output layer. There is no specific criterion for selecting the number of neurons in the hidden layer. Both the hidden layer and the output layer use the sigmoid transfer function. Sigmoid function is an S-shaped mathematical function that computes the output from the network inputs. Its formula is given in equation (1).

$$s(x) = \frac{1}{1 + e^{-x}}. \tag{1}$$

For measuring ANN prediction accuracy, the Mean Squared Error (MSE) function is used. MSE formula is given in equation (2).

$$MSE = \frac{\sum_{i=1}^{N} (y_i - o_i)^2}{N}, \tag{2}$$

where y_i is the predicted output, o_i is the actual output, N is the number of samples. The closer the MSE is to zero, the more accurate the model is.

B. GRADIENT DESCENT ALGORITHM

As long as the weight, inputs, and transfer functions have derivatives, this algorithm is used for training any neural network, where the derivatives of the weights and bias variables are calculated.

Gradient Descent Algorithm aims at reducing the error function which represents the variance between the actual output and the predicted output.

Each iteration of a gradient descent attempts to approach the minimizer cost function by using the gradient's objective function information [16].

C. LEVENBERG-MARQUARDT ALGORITHM (LM)

It is used to update weight and bias values depending on LM optimization. It is based on Newton's algorithm that provides better and faster optimization than Gradient methods. The principle of Newton's method is computing the Hessian matrix (second derivatives) for the performance index at current values of weights and biases. Hessian matrix can be approximated to equation (3).

$$H = J^T.J. \tag{3}$$

The gradient can be calculated according to equation (4).

$$g = J^T.e, \tag{4}$$

where J – is the Jacobian matrix containing the first derivatives of the network errors with respect to weights and biases, and e – is a vector of the network errors.

LM algorithm uses this approximation to the Hessian matrix in the update given in equation (5).

$$X_{k+1} = X_k \ - [J^T.J \ + \ \mu I \ ]^{-1} J^T.e. \tag{5}$$

If the scalar number μ is zero, it is Newton's method only. Employ an approximate Hessian matrix, when μ is large, it becomes a Gradient descent the size of a small step.

When one of the conditions listed below is met, the training is stopped:

1. Reaching the maximum number of epochs iterations.
2. The maximum time has been exceeded.
3. Performance is reduced to the target.
4. The performance gradient is under min-grad.
5. mu overrides mu-max.
6. Validation performance has increased more than times the maximum failure.

D. MALICIOUS NODE

Any wireless sensor network includes a quite big number of nodes with paths among them. It is necessary to discover the existence of a malicious node inside this network, due to the large threats to the nodes in the network.

Tampering with this malicious node makes attackers able to perform many internal and external attacks. This issue may destroy data collection procedure by manipulating the data and release different DoS attacks, etc. [11].

The hacker could send false information because of the Malicious Nodes presence [17].

A malicious node is known as the node which denies service to the rest nodes in the network. Data that is changed by any node in the network in the process of transmission (i.e., before, during, or after the process). This node is de-facto a malicious node. Also, when the malicious node is within the path, it is not chosen as a safe path for data and is thus defined as a malicious path. Hence, data is not send through it [18].

When the signal strength of the message being transmitted does not match the original geographical location, then it is considered as a suspicious message [19].

Features of the malicious node are similar to those of the normal nodes in the network. This issue makes the process of detecting it a difficult and complex task [20].

A malicious node has several malicious behaviors, including the following:

1. Packet Drop: A malicious node drops packets that it receives, rather than forwarding them to the next node.
2. Battery Out: Due to malicious behaviors and needless actions by malicious node, the battery is drained.
3. Buffer overflow: The requested data might not be stored due to overwhelming the buffer by the data generated by the malicious node.
4. Bandwidth consumption: The allocated bandwidth is used by the malicious node. As a result, the other nodes in the network cannot use that bandwidth.
5. Stale packets: Old and meaningless packets are inserted by the malicious node.
6. Delay: malicious nodes can purposely delay packets.
7. Link Break: A malicious node can prevent two normal nodes from communicating in the network if it exists as an intermediate node between them.
8. Message changing: malicious nodes modify the content of the messages.
9. False routing: malicious node transmits false paths.
10. Information theft: malicious node can rob the exchanged messages [21].

E. WSN PERFORMANCE PARAMETERS

Three performance parameters of WSN are considered in this research. These parameters are throughput, power, and network time delay. This work clarifies how values of these parameters differ in existence and non-existence of a malicious node. The presence of a malicious node in the network can be indicated by abnormal changes in the aforementioned three parameters (i.e., increase or decrease).

In data transmission, network throughput is the amount of data moved successfully from one place to another in a given time period. Network throughput is typically measured in bits per second (bps). Power consumption is the amount of energy used per unit time, it is of great importance in digital systems. A network delay is the amount of time required for one packet to go from its source to a destination. It is also called the end-to-end delay.

This study offers a procedure for discovering the existence of malicious node in WSN. The proposed procedure is based on measuring the above mentioned parameters. It must be stated that selection an only parameter may not aid in detecting the existence of malicious node. For instance, distance, packet size, and path selection may result in higher energy consumption. As well, some other causes can affect the other two parameters [10].

More nodes can make the model more susceptible to over-fitting, meaning that it memorizes the training data well and fails to generalize to new data. Over-fitting can also make the model more difficult to interpret and explain, as it can create spurious or irrelevant connections between nodes. Conversely, choosing a smaller number of parameters would be completely useless in actually predicting anything. Therefore, choosing three inputs was the optimal number that minimized error in

both training and validation data, without sacrificing model simplicity and clarity [22].

IV. THE PROPOSED METHOD AND EXPERIMENTAL RESULTS

A. WSN SIMULATION USING NS2

Figure 4 shows the model of WSN used in this research. It is assumed that data packets are of constant size. Any node can generate traffic randomly. This issue results in random traffic which cannot be evaluated. All sending nodes begin sending packets of constant size (500 bytes) at any given time, which results in some kind of undefined traffic in the receiving node.

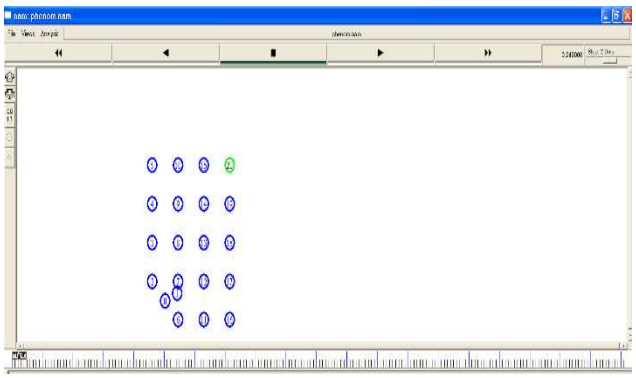


Figure 4. WSN model used in the research

Table 1 contains network simulation parameters considered in this research.

Table 1 parameters of the simulated WSN

Area of Network	1338*616
Number of nodes	21
Number of sources	20
Number of sinks	1
Number of malicious nodes	1
Placement of nodes	Random
Movement of nodes	No

The three parameters were studied in the existence and non-existence of malicious node.

First set of empirical study has been conducted in the non-existence of any malicious node. The measured throughput in the studied WSN in this case was high. This is due to the fact that sending node can communicate with the receiving node without any interference [23].

When the packet size grows in the base station, the throughput grows. After reaching the certain point, it decreases because of congestion or delay [24].

Moreover, each node consumes acceptable power due to the normal operations such as transmitting, receiving, and sleeping [25]. In this situation, the nodes do not waste needless power [26]. In case of non-existence of malicious node, time delay value is in its acceptable limits.

The second set of experiments has been performed in the existence of malicious node. When a malicious node exists, the throughput is influenced where the malicious node attempts to connect to other nodes in order to be able to the destination. Furthermore, this node will consume more power because it will try to connect and transfer the packets [27]. Moreover, when a malicious node exists, delay in packets arrival to the destination has increased. This is because of the nature of the malicious node in this network.

This stage of empirical study (simulation with the existence of malicious node and without the existence of this node) has been used for building the required dataset. This dataset has been used for training an ANN for prediction purpose. The built dataset contains values of the three parameters (throughput, power consumption, and delay). Another dataset has been built as test dataset and contains different values of the parameters. The applied ANN has been built and simulated using MATLAB. Input of the built ANN has been taken from the training dataset. Output of the ANN is the decision of existence or nonexistence of a malicious node in the WSN.

B. THE APPLIED ARTIFICIAL NEURAL NETWORK (ANN)

Figure 5 clarifies the model of the applied ANN in this research. Input layer includes three neurons. The hidden layer includes 10 neurons. Output layer includes one neuron. Output of this ANN is either (0: in case of non-existence of a malicious node in the network) or (1: in case a malicious node exists in the network). Input of this ANN is the previously mentioned three parameters (throughput, power consumption, and delay).

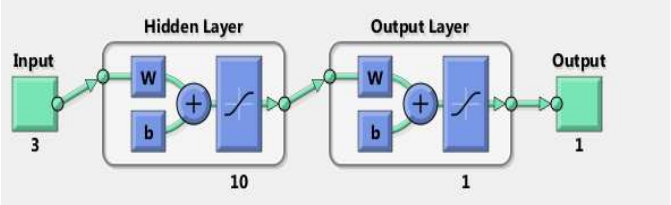


Figure 5. Structure of the used ANN

Training process has been carried out using Descent Gradient and Levenberg-Marquardt algorithms.

C. PROPOSED PREDICTION SYSTEM PERFORMANCE RESULTS

Based on ANN has passed through three stages. These stages are training, validation and testing. The training data is applied to train the neural network, validation data is applied to check the training effect of the neural network, and the test data is applied to show the quality and accuracy of the neural network.

Figure 6 shows the system performance in cases of training, validation, testing, and the best result when using Descent Gradient algorithm. Commonly, errors are reduced after more epochs of training. Errors may begin increasing in validation stage as the network starts over-fitting the training data.

To prevent over-fitting, we chose one hidden layer in our network architecture, which reduces the complexity of the prediction model.

At the beginning of the work, we chose only two parameters as inputs to the neural network, which caused under-fitting so the inputs were increased to three parameters.

Network training stops after six sequential increments of validation error, by default. The epoch with the lowest validation error indicates the best performance in the network.

Figure 7 shows the system performance in cases of training, validation, testing, and the best performance is when using Levenberg-Marquardt algorithm.

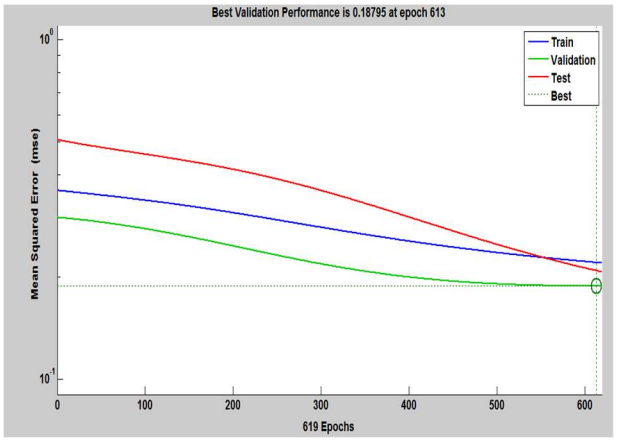


Figure 6. System performance using Gradient Descent algorithm

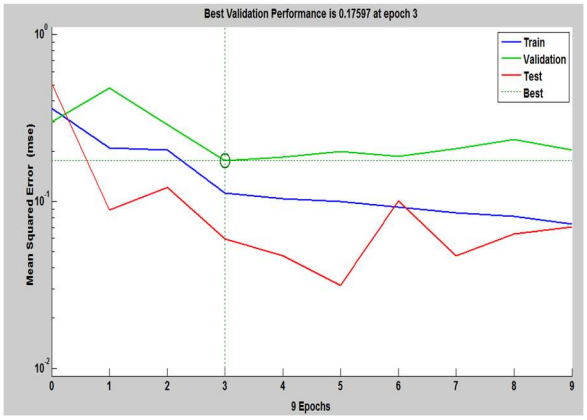
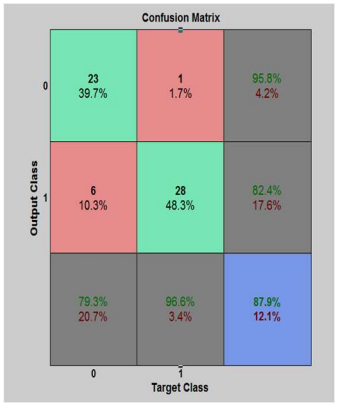


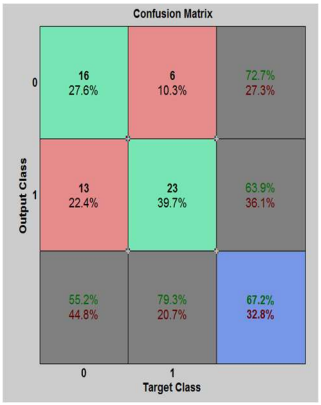
Figure 7. System performance using Levenberg-Marquardt algorithm

D. CONFUSION MATRIX

Confusion matrix is used to assess system accuracy. Rows of this matrix clarify the predicted values while the columns clarify the actual values. Diagonal cells represent the correctly categorized values. The cells outside the diameter are incorrectly labeled values. The rightmost column contains percentages of the predicted values which are correctly and incorrectly classified. The last line contains percentages of the actual values which are correctly and incorrectly classified. Figure 8 shows the obtained confusion matrix.



a) Levenberg-Marquardt algorithm

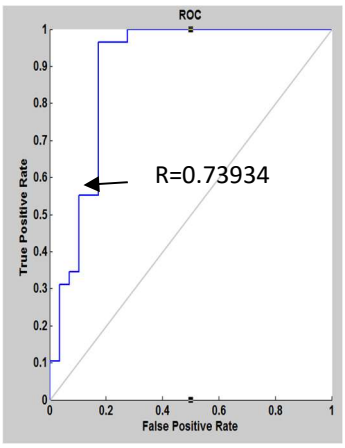


b) Gradient Descent algorithm

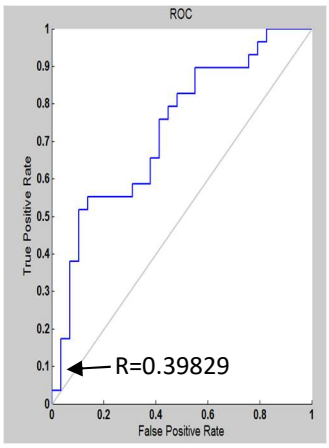
Figure 8. The obtained confusion matrix of the proposed system

E. RECEIVER OPERATING CHARACTERISTIC (ROC CURVES)

Evaluating network quality is a regression test between the network output and the respective targets. It is usually represented by a regression factor (correlation coefficient R). Its value ranges from [0-1] and when it is closer to 1, the network response is more accurate. If the curve moves towards the top, the classification is better. For perfect network performance, R should be (1.0). Practically, this case cannot be reached. Most learning models do not just track 1's or 0's when they make predictions. Instead, they output a continuous value somewhere in the range [0-1]. Values above a certain threshold are labeled 1 and values below this threshold are labeled 0. Figurer 9 shows the regression plot with R=0.73934 for Levenberg-Marquardt algorithm and R=0.39829 for Descent Gradient algorithm.



a) Levenberg-Marquardt algorithm



b) Gradient Descent algorithm

Figure 9. ROC diagram

F. Error estimation

For evaluating quality of the trained network, more information can be inferred from the error diagram. This diagram depicts division of the residuals between the target values and the actual values. This diagram is able to indicate outliers.

Bins: the number of vertical bars that appear in Figure 10.

Zero error: corresponds to the zero error value on the error axis and it separates positive values from negative values.

A positive error means that the actual output is smaller than the predicted value. Negative error means that the predicted value is smaller than actual output value. Figure 10 and Figure 11 show error diagrams of the discussed cases.

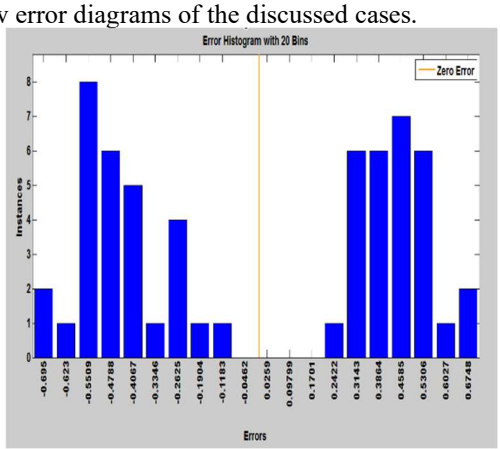


Figure 10. Error diagram in case of Gradient Descent algorithm

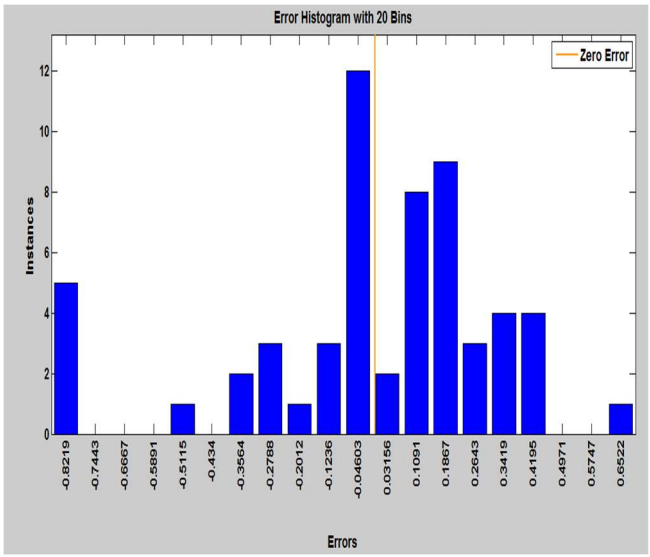


Figure 11. Error diagram in case of Levenberg-Marquardt algorithm

Comparing the results of the above two algorithms shows that the Levenberg-Marquardt algorithm is better than the Gradient Descent algorithm. The performance diagram also proves this result. In the case of Levenberg-Marquardt algorithm, the best point was reached after epoch 3 while in the Gradient Descent algorithm the best point needed more than three epochs. Moreover, from ROC diagram, the threshold value in the Levenberg-Marquardt algorithm is higher than the threshold value in case of the Gradient Descent algorithm. In addition to that, the Confusion matrix shows that the accuracy of the Levenberg-Marquardt algorithm is higher than the accuracy the Gradient Descent algorithm. Finally, the error diagram shows that the Levenberg-Marquardt algorithm is better than Gradient Descent algorithm.

V. CONCLUSIONS AND FUTURE WORK

It is well known that the existence of a malicious node in WSN degrades network performance. This research presents a method for detecting the presence of the malicious node. The proposed method is applied in two stages. The first stage is performed in the presence of the malicious node and in the absence of the malicious node. In the second stage, detection process is carried out based on Artificial Neural Network (ANN). As future work, we can use more parameters to be inputs for our proposed ANN.

References

[1] J. A. Stankovic, "Wireless sensor networks," *Computer*, vol. 41, no. 10, pp. 92–95, 2008. https://doi.org/10.1109/MC.2008.441.

[2] IntechOpen, "Wireless sensor networks," [Online]. Available at: https://www.intechopen.com/chapters/76818.

[3] Wiley Online Library, "Special issue: Wireless sensor networks," [Online]. Available at: https://onlinelibrary.wiley.com/doi/toc/10.1155/9071.si.204518.

[4] J. Sen, "Security in wireless sensor networks," *Wireless Sensor Networks: Current Status and Future Trends*, vol. 407, pp. 407–408, 2012. https://doi.org/10.1201/b13092-21.

[5] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, no. 12, pp. 2292–2330, 2008. https://doi.org/10.1016/j.comnet.2008.04.002.

[6] L. Darwish, M. Nassr, F. Ghosna, H. M. Fardoun, D. K. Voronkova, and M. Anbar, "Malicious node detection in wireless sensor networks: Comparative study," *Proceedings of the 2023 5th Int. Youth Conf. Radio Electronics, Electrical and Power Engineering (REEPE)*, 2023, pp. 1–5, https://doi.org/10.1109/REEPE57272.2023.10086790.

[7] Y. Zhang and W. Cai, "The key technology of wireless sensor network and its application in the internet of things," *Journal of Sensors*, vol. 2022, 1817781, 2022. https://doi.org/10.1155/2022/1817781.

[8] M Babu, M Ramkumar, M Shenbagapriya, "Detection of malicious nodes in wireless sensor network," *ICTACT Journal on Communication Technology*, vol. 10, issue 4, pp. 2067-2072, 2019.

[9] G. M. N. Veerabadrappa and P. M. Booma, "ESDAM-efficient and secure data aggregation against malicious nodes in IoT environment," *Int. J. Innovative Technol. Exploring Eng. (IJITEE)*, vol. 9, no. 2, pp. 2278–3075, 2019. https://doi.org/10.35940/ijitee.B6713.129219.

[10] L. Alzubaidi, J. Zhang, A. J. Humaidi et al., "Review of deep learning: concepts, CNN architectures, challenges, applications, future directions," *J. Big Data*, vol. 8, no. 53, 2021. https://doi.org/10.1186/s40537-021-00444-8.

[11] Z. Zhang, Y. Yang, W. Yang, F. Wu, P. Li, and X. Xiong, "Detection and location of malicious nodes based on homomorphic fingerprinting in wireless sensor networks," *Security and Communication Networks*, vol. 2021, 9082570, 12 pages, 2021. https://doi.org/10.1155/2021/9082570.

[12] G. Zheng, B. Gong, and Y. Zhang, "Dynamic network security mechanism based on trust management in wireless sensor networks," *Wireless Communications and Mobile Computing*, vol. 2021, Article ID 6667100, 10 pages, 2021. https://doi.org/10.1155/2021/6667100.

[13] H. S. Saini, R. Sayal, and S. S. Rawat, *Innovations in Computer Science and Engineering*, Springer Singapore, 2021.

[14] P. Singhal and A. Yadav, "Congestion detection in wireless sensor network using neural network," *Proceedings of the International Conference on Convergence for Technology*, Apr. 2014, pp. 1–4. https://doi.org/10.1109/I2CT.2014.7092259.

[15] M. H. Beale, M. T. Hagan, and H. B. Demuth, "Neural network toolbox user's guide", *MathWorks*, vol. 2, pp. 77–81, 2010.

[16] S. H. Haji and A. M. Abdulazeez, "Comparison of optimization techniques based on gradient descent algorithm: A review," *PalArch's Journal of Archaeology of Egypt/Egyptology*, vol. 18, no. 4, pp. 2715–2743, 2021.

[17] M. Kumar, P. Mukherjee, K. Verma, S. Verma, and D. B. Rawat, "Improved deep convolutional neural network based malicious node detection and energy-efficient data transmission in wireless sensor networks," *IEEE Trans. Network Sci. Eng.*, vol. 9, no. 5, pp. 3272–3281, 2021. https://doi.org/10.1109/TNSE.2021.3098011.

[18] IGI Global, "A novel secure routing protocol in MANET," [Online]. Available at: https://www.igi-global.com/dictionary/a-novel-secure-routing-protocol-in-manet/33926.

[19] W. R. Pires, T. H. de Paula Figueiredo, H. C. Wong, and A. A. F. Loureiro, "Malicious node detection in wireless sensor networks," *Proceedings of the 18th Int. Parallel and Distributed Processing Symposium*, Apr. 2004, p. 24. https://doi.org/10.1109/IPDPS.2004.1302934.

[20] R. Vijayarajeswari, A. Rajivkannan, and J. Santhosh, "Survey of malicious node detection in wireless sensor networks," *Int. J. Emerging Technol. Innovative Eng.*, vol. 2, no. 6, pp. 335-338, 2016. https://doi.org/10.5958/2249-7315.2016.01177.1.

[21] P. Anu, S. Vimala, "Optimization of open shortest path first routing," *European Journal of Molecular & Clinical Medicine*, vol. 7, no. 11, pp. 94-99, 2020.

[22] LinkedIn, "How can you identify the optimal number of nodes in ANN?" [Online]. Available at: https://www.linkedin.com/advice/1/how-can-you-identify-optimal-number-nodes-ann-v6iwf.

[23] S. D. Kadu and V. S. Deshpande, "Handling throughput in wireless sensor network," *Proceedings of the 2012 IEEE Int. Conf. Computational Intelligence and Computing Research*, Dec. 2012, pp. 1–4. https://doi.org/10.1109/ICCIC.2012.6510281.

[24] S. D. Kadu and V. S. Deshpande, "Characterization of throughput in wireless sensor network for MAC and routing protocol," *Proceedings of the 2013 Int. Conf. Cloud & Ubiquitous Computing & Emerging Technologies*, Nov. 2013, pp. 108–111. https://doi.org/10.1109/CUBE.2013.55.

[25] K. Gulati, R. S. K. Boddu, D. Kapila, S. L. Bangare, N. Chandnani, and G. Saravanan, "A review paper on wireless sensor network techniques in Internet of Things (IoT)," *Materials Today: Proceedings*, vol. 51, pp. 161–165, 2022. https://doi.org/10.1016/j.matpr.2021.05.067.

[26] H. A. A. Al-Asadi, R. Hasan, M. Nassr, and M. Anbar, "Power consumption in wireless sensor network: A machine learning approach," *Computing, Performance and Communication Systems*, vol. 6, no. 1, pp. 24–37, 2022.

[27] B. Rajasekaran and C. Arun, "Detection of malicious nodes in wireless sensor networks based on features using neural network computing approach," *Int. J. Recent Technol. Eng.*, vol. 7, no. 4, pp. 188–192, 2018.

LARA DARWISH *received her Bachelor's Degree in Communication Technology from Tartous University, Syria, in 2019. Faculty of Information and Communication Technology Engineering, Department of Communication Technology, Tartous University, Tartous, Syria.*

MOHAMMAD NASSR *received his Bachelor's Degree in Electronics and Communications Engineering from Damascus University, Syria, in 2007, M.Tech. in Applied Electronics from Damascus University in 2012, and Ph.D. in Applied Electronics from Damascus University 2015. He is currently working as Associate professor in Communication Department, Tartous university, Tartous, Syria. He is also working as contracted lecturer with the department of Communication Engineering Alitihad Privet University, Damascus, Syria. Doctor Nassr has at least 8 years of teaching experience for different courses in Electronics and communication engineering. His research interests include Wireless Communication, Mobile Communication, Optical Communication, IoT.*

MOHAMMAD ANBAR *received his Bachelor's Degree in Electronics Engineering from Tishreen University, Lattakia, Syria, in 2003, M.Tech. in Computer Science from Jawaharlal Nehru University, New Delhi, India in year 2007, and Ph.D. in computer science from the school of Computer & Systems Sciences, Jawaharlal Nehru University, New Delhi. He is currently working as HOD of the Department of Telecommunication Technology Engineering, Tartous university, Tartous, Syria. He is also working as contracted lecturer with the department of Information Technology Engineering, Qasyoun Private University, Syria. Doctor Anbar has at least 12 years of teaching experience for different courses in computer and communication engineering.*

Doctor Anbar has more than 15 published research papers in various international journals and conferences such as Wiley, IGI-Global USA, etc. He was in the editorial board of the book "Technologies and Protocols for the Future Internet Design: Reinventing the Web", IGI-Global USA. His research interests include Wireless Communication, Mobile computing, Soft Computing techniques, IoT. Etc.

HAMID ALI ABED ALASADI *was born in Iraq. He received the B.Sc and M.S. degrees in electrical engineering and communication engineering from Basra University, Basra, Iraq, in 1987 and 1994, respectively, and the Ph.D. degree from the University Putra Malaysia in Communication Network Engineering in 2011. From 1995-2018, he is a head of department of Computer science, Basra University. In 2014, he joined the Basra University as a Full Professor. Since November 2018 he has been head of the Department of communication engineering in the Iraq University College, Iraq. His research interests include optical communications, optical fiber, information theory, Wireless Network, Sensor Network, Fuzzy Logic and Neural Networks, Swarm Intelligence, computer engineering, and Artificial intelligence. He is member of scientific and reviewing committees of many journals and international conferences in the domains of Computer and communications engineering.*