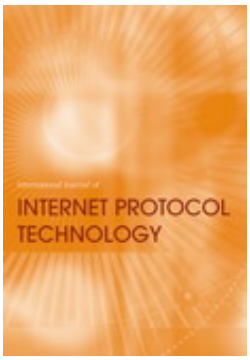
INDERSCIENCE PUBLISHERS
Linking academia, business and industry through research



International Journal of Internet Protocol Technology

ISSN online: 1743-8217 - ISSN print: 1743-8209
https://www.inderscience.com/ijipt

Deep learning algorithms providing security for wireless sensor networks against malicious attacks

Dinokumar Kongkham

DOI: 10.1504/IJIPT.2024.10067606

Article History:

Received:	04 November 2023
Last revised:	25 March 2024
Accepted:	08 June 2024
Published online:	06 January 2025

Deep learning algorithms providing security for wireless sensor networks against malicious attacks

Dinokumar Kongkham

Department of Electronics and Communication Engineering,
Saveetha School of Engineering,
Saveetha Institute of Medical and Technical Science (SIMATS),
Saveetha University,
Chennai, Tamil Nadu, India
Email: dinokumarkongham.sse@saveetha.com

Abstract: Small sensor nodes that have limited energy are the building blocks of wireless sensor networks, often known as WSNs. WSNs are self-sufficient and space-distributed. A WSN is vulnerable to security concerns because it lacks a central authority and deploys its nodes in a random fashion across the network. A malicious assault is a well-known kind of attack in WSN. This type of attack involves a hacked node impersonating as one of the network nodes and fooling other nodes. Either via the use of cryptographic techniques or by the synchronisation of time, a variety of strategies are created to defend against these attacks. However, due to the autonomous nature of WSNs, these strategies may not be successful. To protect against malicious assaults, this article presents a technique that is both effective and efficient, which is known as the Hamming Residue Method (HRM).

Keywords: deep learning; security; WSN; wireless sensor network; malicious attacks.

Reference to this paper should be made as follows: Kongkham, D. (2024) 'Deep learning algorithms providing security for wireless sensor networks against malicious attacks', *Int. J. Internet Protocol Technology*, Vol. 17, No. 1, pp.1–8.

Biographical notes: Dinokumar Kongkham is currently working as an Assistant Professor (SG) in the Department of Electronics and Communication Engineering at Saveetha School of Engineering, (SIMATS-Saveetha University), Chennai. He received his MTech degree in Communication System from SRM University, Chennai and a PhD degree in Wireless Communication from Bharath Institute of Higher Education and Research (BIHER-Bharath University, Chennai) in 2021. He has three years of Post PhD teaching and research experience and he has published more than 10 International journals (Scopus Indexed and SCIE indexed) and Conferences. His research interest areas include wireless communication, cognitive radio networks, 5G new radio, networking, etc.

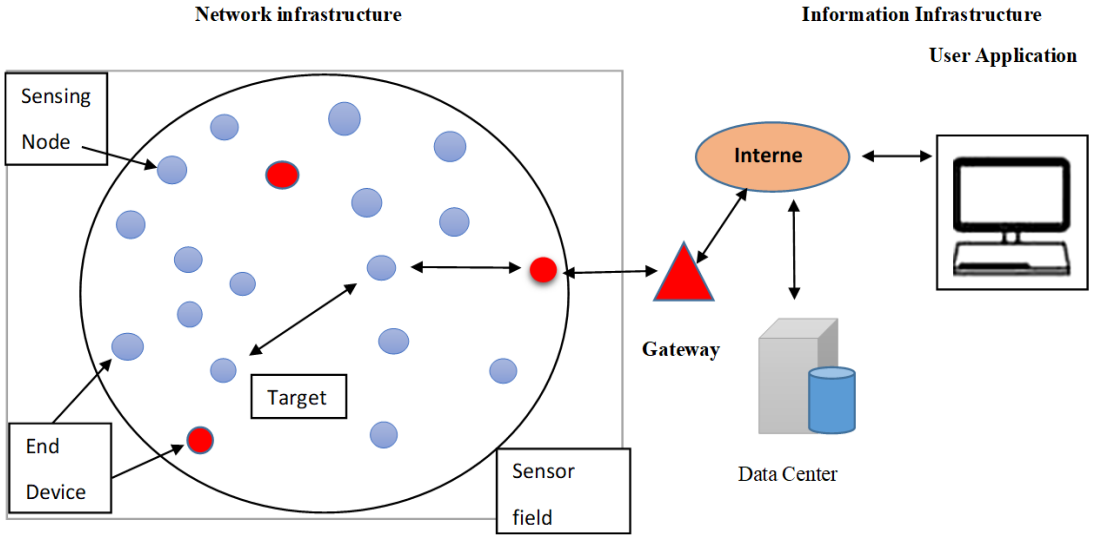
1 Introduction

Data collection, sharing, analysis and display can be done for a specific use case through the Internet of Things (IoT). Internet of Things (IoT) systems are used for many things, such as figuring out where things are and sharing local information, keeping track of mobile assets, sensing the environment, keeping an eye on medical patients from afar, ad hoc networking and safe communication (Chen et al., 2014). They have a server, an information infrastructure, and a network infrastructure (Acosta et al., 2022).

The nodes receive data from the real world and send it to a place called a sink node for storage. When you build an Internet of Things network topology based on WSN, you have to think about a lot of different technology areas (Li and Yang, 2006). Communication and wireless sensor networks, information, modulation theory, Radio Frequency (RF) circuits and stoctic design are some of these areas. Moreover, planning the infrastructure for the internet of Things (IoT) network usually needs the help of professional planners who fully understand all the different parts of putting an app into a production setting (Anastasi, 2009). How accurate and well the model works is directly related to how much the creator knows about and is comfortable with the modelling tools. For studying and simulating, the Internet of Things network infrastructure models are simplified versions of how the networks work that are shown in a way that works (Stanley-Marbell et al., 2008). Our main focus has been on making the layers that make up the OSI reference model easier to understand.

Also, we came up with a classification that would help us order the literature that is already out there. In addition, this article's main goal is to support Internet of Things designers who already know a lot about some layers to use models from other levels in their projects and use a cross-layer method to look into the connections between layers.

Figure 1 Architecture of IoT systems (see online version for colours)



The summary of the article is as follows:

A cross-layer view is used to look at and group the suggested Internet of Things network infrastructure models. It does this by using a system-centric method that considers metrics that aren't usually looked at in WSN model studies (Parashar et al., 2020; Amutha et al., 2020). It is shown that there are three simple taxonomies by showing and comparing the measurements of each group.

2 Related work

Within the DOIDS that is being suggested, the technique of location tracking has been employed. In situations when there is a limited number of sensor nodes deployed, the DOIDS is an appropriate choice for underwater networks. The clustering methods that are referred to as DBSCAN are used in order to identify the malicious nodes. The people who worked on (Ahmad et al., 2021) created a safe route design that works well with Underwater Sound Sensor Networks (UASNs). One hash operation and one bilinear mapping function are needed to open the trap door. The study used the NS2 computer and AquaSim, a UWSNs modelling package, to test how well the method in the study worked. To see how well GPNC and LB-AGR work side by side, we look at their output, energy use and power consumption ratio. The results show that the suggested way improves both the network's speed and safety.

In the context of Unstructured Wireless Sensor Networks (UWSNs), Javanmardi et al. (2012) developed a distributed method capable of protecting against specific routing attacks. Among the many types of attacks that may be carried out against routing protocols, the method that has been proposed is able to recognise both internal and external assaults. There are two stages involved in the method that has been suggested: detection and quiet surveillance (Muller and Valle, 2010). Surveillance of the communication between neighbouring sensor nodes is carried out by the sensor nodes for the purposes of detection and mitigation. A secure mechanism for locating neighbours is used by each sensor node during the initial deployment process in order to identify its neighbours (Ketshabetswe, 2019). Through the use of neighbour activity monitoring, the objective is to identify harmful behaviour on UWSNs. Because of the sinkhole attack, it is possible that the packets that are received may be changed or may be discarded. The rogue node would have to lose or change the packets for the signatures to not match. This would reveal that an attack had taken place (Khazaei et al., 2009). The suggested method in this work can spot active strikes, but not silent ones. For instance, the method that has been talked about can't spot a bad node that can record data for analysis but doesn't drop or mess with it. Comparing fingerprints is another way that the proposed method may detect attacks that exploit contained and out-of-bound wormholes. If a bad node is found in the UWSN setting, a method of separation is used to keep the network away from the bad node (Barati et al., 2008). Because of this, the bad node can't take part in UWSN events or mess up route processes. This is what happened because of the last line.

The study idea was put into practice with the OMNET++-based Castalia emulator. This research could go even further in the future by coming up with new ways to attack in the UWSN setting (Babayo et al., 2017).

For the context of UASNs, a technique for the safe detection of neighbours was proposed in Akbari et al. (2022).

If the attacker discovers that a neighbouring property is susceptible to assault, they have the ability to launch a wormhole attack regardless of the hostile environment. The assault that was carried out by the wormhole has resulted in unwanted effects that cannot be rectified by the use of cryptographic techniques (Ghosh and Das, 2008). A family of protocols that are resistant to wormholes and perform secure neighbour finding in Ubiquitous Ad Hoc Network (UASN) environments was proposed in this research paper. The approach of arrival-signal direction serves as the foundation for the procedures that have been proposed by this study. It is possible for the approach that has been presented to resist assaults from wormholes. The following are the four protocols that are included in the system that has been suggested (Waite, 2002). Listed below are the results of the evaluations conducted on the four different protocols: In the first place, there is a very

good possibility that B-NDP will be able to stop dishonest neighbours from forming ties with one another.

With regard to the context of UASNs, Ahmad et al. (2021) suggested a security suite that included both mobile and static nodes on its roster of components (Sah and Amgoth, 2018). One component of the security suite is comprised of cryptographic fundamentals and secure routing protocols. The researchers first proposed the FLOOD method as a potential solution. SeFLOOD, which stands for secure flood, is a new protocol that has been established with a secure variation. The SeFLOOD protocol's performance was tested to find out how much extra work the FLOOD protocol needed to be safe. It has been shown by the results of the trials that the proposed suite is suitable for the context of the UASNs programme. Both the degree of communication overhead and the amount of power that is required are reduced in the suite that is being suggested. Not only does the proposed suite have reduced connectivity expenses, but it also requires less power. Here is a list of the suggested protocol suite's most important accomplishments. The suggested suite works well in part because the cypher text growth doesn't have a big effect. In the finding part of the secure protocol, there is 6% less delay than in the non-secure protocol. In the safe protocol, the change step did not add any extra work. In the unsafe protocol, it did. The safe process was made based on Lampson's ideas about how computers should be put together.

The experts found in Pranitha and Anjaneyulu (2014) that DOS attacks were their main target. In the group of strikes called spread denial of service attacks are flooding, man-in-the-middle, and destruction. Attacks called MITM in UWSNs take over data that is being sent between sensor nodes for some reason. When it comes to UWSNs, MITM tactics like the tunnel attack, the Sybil attack and selective sending are all possible. During a flooding attack, the bad node or nodes cause delay by sending a steady stream of packets to the base station. The base station gets too much info because of this. When it comes to Under-Wired Sensor Networks (UWSNs), the flooding attack changes how the network works as a whole. Changes or meddling with the settings of the sensor node are examples of a bombing attack in wireless sensor networks (WSNs). This destroys the network as a whole. A big part of the destruction attack is the amount of physical protection that was used. The out-of-coverage problem and the wrong friend identification problem are two problems that mobile sensor nodes have to deal with when they are working in UWSN. This study's authors used Aqua-Sim as a simulation tool in order to conduct their research (Jodeh et al., 2018).

The multilayered network structure that MuLSi offers is a network structure. Owing to the optimal location of the washbasin, it is possible to eliminate communication that requires several hops (Yoneki and Bacon, 2005). The forwarder nodes that are believed to be the best are those that are located closest to the washbasin. No information on the location of the node is required for the approach that is described in this study. When employing MuLSi, the performance of the network is improved; but, since MuLSi only has a single connection, the operation has a low degree of reliability (Yuan et al., 2017). For this reason, MuLSi-Co makes use of collaboration methods, in which the receiver is provided with multiple copies of the stored data (Zhang and Cuiping, 2012).

3 Proposed methodology

The ANN-based method we showed has three layers. The CICIDS2017 data set, which was given to us by the Canadian Institute of Cybersecurity, was used to train and test the suggested model. This was done because it's hard to find real statistics for Wireless Sensor Networks (WSNs). The majority of the attack scenarios that are presently relevant are included in this data set, which was sufficient to assess the efficiency of the solution that we have suggested. The four characteristics that were retrieved from CICIDS2017 were used to train the model that was under consideration. Twenty thousand input vectors are included in the data set that is used in our proposed mechanism.

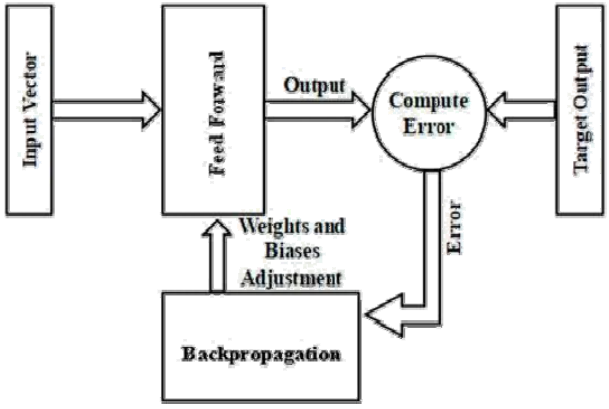
3.1 Training

Through training, the system will be able to acquire the knowledge necessary to understand how to categorise nodes into the required category. Figure 2 shows the two steps that are used to train the suggested system.

Table 1 Summary of related work

Technique	*Contribution*	*Tool used*
IDS for Opportunistic Routing in UWSNs	Proposed an Intrusion Detection System (IDS) for reducing the bad influence of malicious nodes on the transmission of data. The mechanism of location monitoring is adopted in the proposed DOIDS. The obtained results show that proposed algorithm significantly improved the accuracy rate of detection from 3% to 15% in different scenarios.	Not mentioned
Secure routing scheme for UASNS	Recommended secure routing for UASNs. Signature algorithm is proposed for authentication between source and destination node. A trap-door scheme is used in order to achieve anonymity of the nodes.	NS2 with AquaSim
Securing network from routing attacks	Proposed distributed approach for detecting and mitigating the routing attacks in UWSNs. An analytical model is proposed for the said purpose.	Castalia simulator
Secure discovery of neighbour in UASNs	Proposed protocols suite for secure neighbor discovery in UASNs. The proposed protocols are based on the Direction of Arrival (DoA) signals approach.	C++
Secure suite for UASNs	Proposed scheme includes secure routing protocol and cryptographic primitives. Proposed protocols suite has limited power consumption and overhead; that is why it is suitable for UASNs.	Real data used

Figure 2 Proposed mechanism training



3.1.1 Feed-forward

A feed-forward setup gives the system a vector of values to work with along with the value it should produce when it's done. Because these values change the results that are needed, picking starting weights and bias values is not easy. The linked weights then send this input vector to the buried layer.

$$IH1 = \sum_{2-1}^{4} \sum_{yx-2x}^{y} X_1 W_x + b_1 \tag{1}$$

Table 2 VECTOR input parameters

Attack type	*Input vector parameter for ANN*
Black-hole attack, gray-hole attack and wormhole attack	Number of packets received by the node Number of packets sent by the node Energy-consumption details Trust value of the node in the network

Represented in equation (1), which can be found above.

$$CH_1 = 1 \Big/ \left(1 + e^{-IH_k}\right) \tag{2}$$

Equation (1) is responsible for calculating the value of 'IHk'. Each output layer node will get its input from the result of equation (2), which will be applied to each hidden layer node on its own.

$$D1 = \sum_{2-1}^{3} \sum_{m-x}^{f} OH_2 W_m + b_1 \tag{3}$$

$$Out_1 = 1 \Big/ \left(1 + e^{-(\infty_k)}\right) \tag{4}$$

This is the last step in the feed-forward training process for the suggested mechanism. To figure out how far off the estimated output (AO) is from the planned output (DO), use the following equation (5):

$$E = \frac{1}{2} \sum_{1-1}^{4} \left(DO_1 - AO_1\right)^2 \tag{5}$$

During the numerous tests that were carried out for the purpose of determining the best error values.

3.1.2 Backpropagation

If the error found in the feed-forward stage is greater than 0.20, it will be sent backwards to change the connected weights of all neurons in the hidden and input layers, as well as the bias value that is linked to neurons in the hidden layer, until the error rate is reduced to the required level.

$$CGadi = AO_i \left(1 - AO_i\right) + Error$$

$$\Delta w_j = LRCGadj + Hl_{o1} \tag{6}$$

$$\Delta b_j = LR + CGadj \tag{7}$$

$$HGadi = Hl_{o1}\left(1 - Hl_{o1}\right) + \left(\sum_{j-1}^{3} OutGadj + W_j\right) \tag{8}$$

$$\Delta w_k = LR + HGadi + X_1 \tag{9}$$

$$\Delta b_i = LR + HGadi \tag{10}$$

The feed-forward will now start over from scratch. Figure 4 shows that the suggested system will no longer need to be trained after the new weights and biases are made. After that, testing will start right away with the three thousand input vectors that make up the leftover data sets.

Figure 3 First input vector forward pass

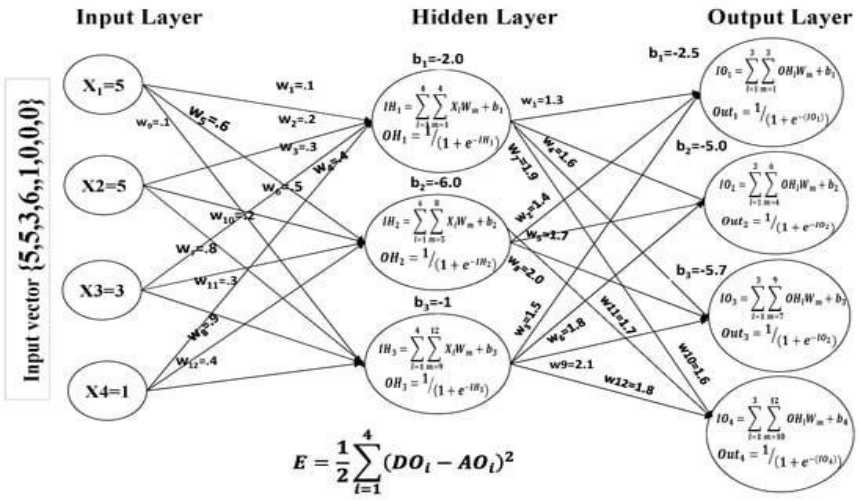
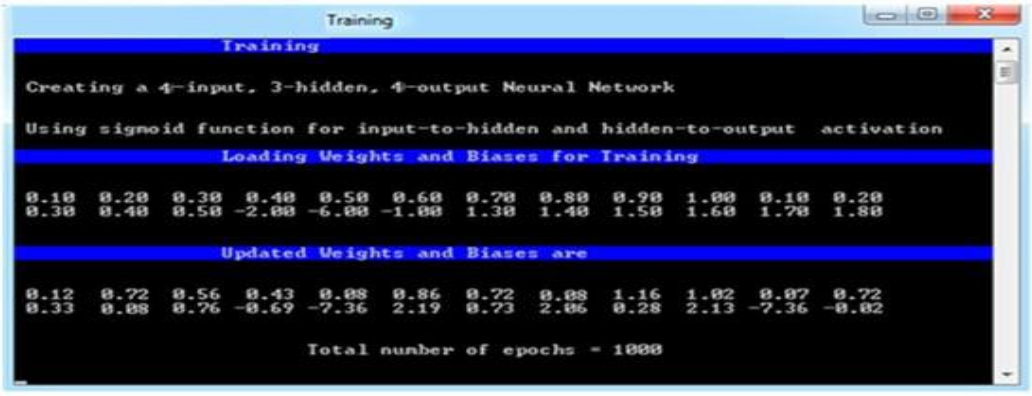


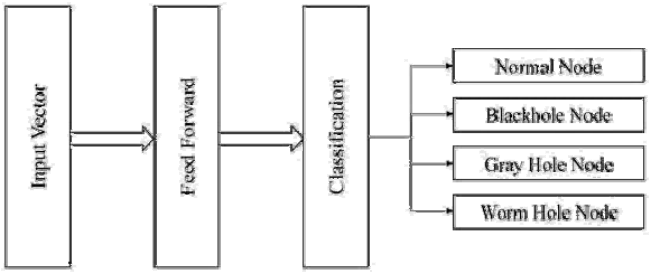
Figure 4 Recalculating input vector weights (see online version for colours)



3.2 Testing

Once it has been trained, the system will use the completely new weights and bias values to put the last 3000 input vectors from the data set into the right group. The machine will learn how to do this before it does it. While the system is being tried, it's important to remember that the learning rate, momentum, mistake rate and number of epochs all stay the same. The testing phase is different from the training phase in that input vectors are not given to the system in order to get the desired results.

Figure 5 Proposed mechanism testing



The new weights and bias values are used by the system to put the last 3000 input vectors from the data set into the right groups after training is over. That's why the learning rate, momentum, mistake rate and number of epochs are all set to the same amount.

4 Results and discussion

This part summarises the most important things found during the research. It was decided that the game would last for 60 seconds and that the AODV routing protocol would be used.

Different situations were taken into consideration when the simulation was being run.

In the last scenario, the planned system was put into action, and all of the different kinds of malicious nodes that were employed in the other situations, in addition to the normally functioning nodes, were used. As shown in Figure 7, the base station received about the same quantity of packets during each and every assault as it did during the usual scenario, which did not include any attacks.

Table 3 Simulated parameters

S. No.	*Parameters*	*Range/value*
1	Area	1000 x 1000 m
2	Nodes	500
3	BS location	1300–1400 m
4	Initial energy	1.5 J
5	Trust	1 or 0
6	Routing Protocol	AODV
7	Simulation Time	120 s
8	Bandwidth	25 Kbps
9	Transmission range	50 m
10	Packet size	512 Bytes

Figure 6 Base-station packets received (see online version for colours)

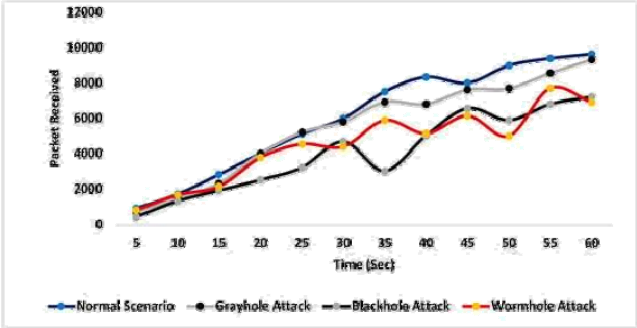
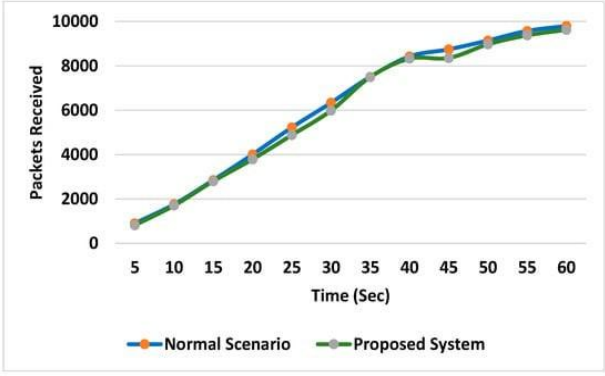
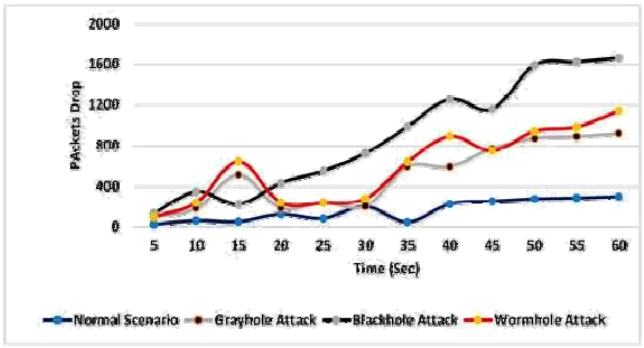


Figure 7 Base station packet counts compared (see online version for colours)



In the black-hole attack, both the rate of data loss and the amount of energy used by the network were very high, as shown in Figures 8 and 10. This was true for all of the situations that were talked about.

Figure 8 Non-proposed system packet loss rates (see online version for colours)



The data shown in Figure 9 demonstrates that the network's energy consumption was 85 J when there was no attack, but that it rapidly grew when there was an assault on the routing path.

Figure 9 Network energy usage without the suggested system (see online version for colours)

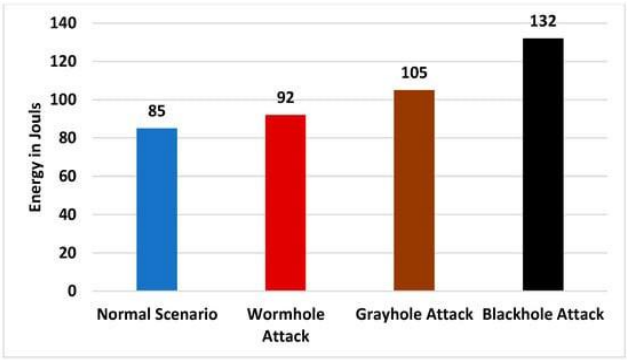
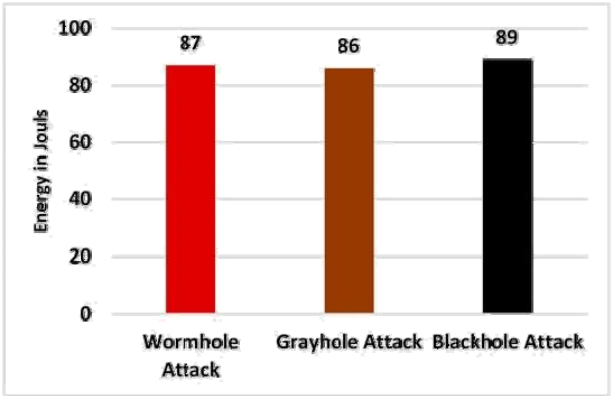
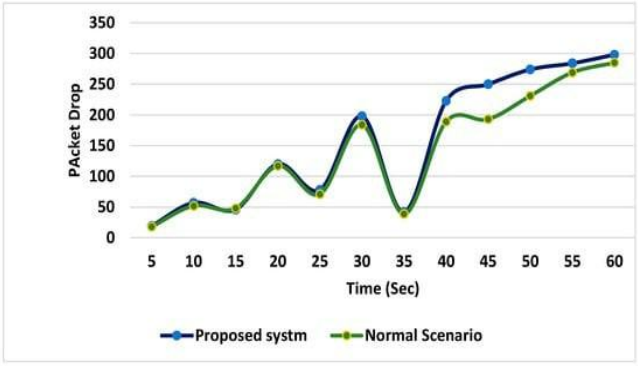


Figure 10 Energy usage during routing attacks using the suggested system (see online version for colours)



Also, the rate of data loss was very low because rogue nodes were found quickly, as shown in Figure 11. This is what happened when the suggested system was put in place and all of the route attacks were run during the exercise.

Figure 11 Compare packet dropouts (see online version for colours)



Different types of machine learning are used to see how well the suggested model works. These include Random Forest (RF), Decision Trees (DT) and Support Vector Machines (SVM). The sample that was used for validation was used for 10% of the input vectors. You can see how the information for all attacks and normal cases are spread out in Table 4.

Table 4 Data set input vector distribution

Out all	*Training*	*Valuation*	*Training*	*Total*
Bladebele	3990	480	780	48-20
Arsyhole	3580	495	720	47-45
Wocnhofs	3620	430	710	4320
Total	11,190	1405	2210	4350

In Table 5, you can see the outcomes of a review that used an uncertainty matrix to check how well the suggested system worked during the testing process.

Table 5 Confusion matrix

			Predicted Class			
	S. No.	*Classes*	*Normal*	*B.H*	*G.H*	*W.H*
Actual class	1	Normal	806	1	2	1
	2	B.H	1	755	3	1
	3	G.H	1	2	716	1
	4	W.H	1	1	3	705
		Precision	99.63	99.47	98.90	99.58

Investigated for a variety of assault scenarios. Specifically, the following mathematical expression applies to these matrices:

$$DR \ or \ \ TPR = \frac{TP}{TP + FN} \tag{11}$$

Table 6 Proposed system performance assessment

S. No.	*Class*	*DR*	*FPR*	*Precision*	*F1Score*	*Accuracy*
1	Normal	99.34	0.43	99.62	99.48	
2	B.H	99.08	0.50	98.82	98.95	
3	G.H	99.13	0.50	98.89	99.01	90.49
4	W.H	99.29	0.50	99.71	99.50	
5	Average	99.21	0.48	99.26	99.23	

A TPR of 1, which implies that every incursion is correctly recognised, is exceedingly uncommon for a classifier to acquire. This is because it is highly unusual.

$$FPR = \frac{FP}{FP + TN} \tag{12}$$

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{13}$$

$$Pricission = \frac{TP}{TP + FP} \tag{14}$$

The level to which an algorithm makes correct guesses is called its accuracy.

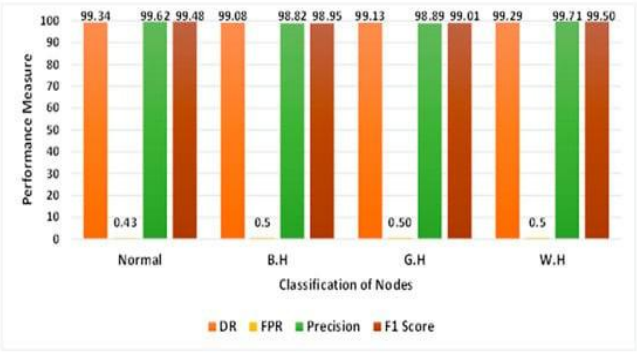
$$F1score = 2 \times \frac{(DR \times Pricission)}{(DR + Pricission)} \tag{15}$$

Table 7 and Figure 12 show the results of the study that looked at how well the suggested method would work. That being said, almost every type of attack has been put in the right category.

Table 7 Comparing routing attack detection rates

S. No.	*Techniques*	*Normal*	*Gray Hole*	*Black Hole*
1	SVM	84	55.7	63.6
2	RF	85	59.1	65
3	DT	86	73	73.6
4	Proposed system	99.34	99.13	99.08

Figure 12 Proposed system performance assessment (see online version for colours)



It has been shown that the model has an average detection rate of 99.21%. This is 1.38% better than the proposed technique's detection rate of 97.8%. These differences can be seen in Figure 13. This is a big improvement over how often these other programmes found things.

Figure 13 Comparison of detection rates (see online version for colours)

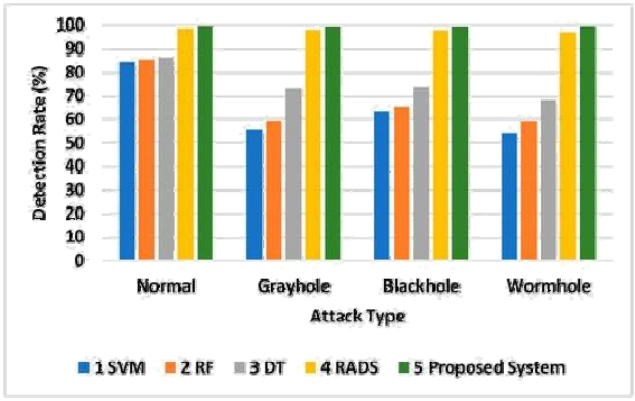
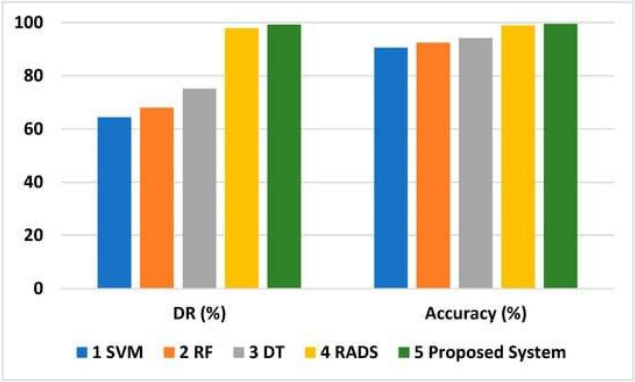


Figure 14 A comparison of average detection rate and accuracy (see online version for colours)



5 Conclusion

This technology, known as the Hamming residue technique, is used to increase the security of wireless sensor networks. If there are a greater number of competing nodes located at various hops in the network, the technique that has been shown is not only straightforward but also very successful. Every node generates a unique security codeword, which enhances the effectiveness of the recommended approach and increases the degree of secrecy between the nodes. It also facilitates the identification of any rival nodes that could be present inside the network. A reduction in the mathematical complexity is another benefit of the technique that has been described.

References

Acosta, J.P.C., Mojica, R.A.U., Mosquera, L.C.D.B., Paez-Rueda, C. and Fajardo, A. (2022) 'Design and implementation of a cost-effective object tracking system based on LoRa, firebase, and Mapbox', *IEEE Latin America Transactions*, Vol. 20, pp.1075–1084.

Ahmad, B., Jian, W., Enam, R.N. and Abbas, A. (2021) 'Classification of DoS attacks in smart underwater wireless sensor network', *Wireless Personal Communications*, Vol. 116, pp.1055–1069.

Ahmad, I., Rahman, T., Zeb, A., Khan, I., Ullah, I., Hamam, H. and Cheikhrouhou, O. (2021) 'Analysis of security attacks and taxonomy in underwater wireless sensor networks', *Wireless Communications and Mobile Computing*. Doi: 10.1155/2021/1444024.

Akbari, M.R., Barati, H. and Barati, A. (2022) 'An overlapping routing approach for sending data from things to the cloud inspired by fog technology in the large-scale IoT ecosystem', *Wireless Network*, Vol. 28, pp.521–538.

Amutha, J., Sharma, S. and Nagar, J. (2020) 'WSN strategies based on sensors, deployment, sensing models, coverage and energy efficiency: review, approaches and open issues', *Wireless Personal Communications*, Vol. 111, pp.1089–1115.

Anastasi, G., Conti, M., Di Francesco, M. and Passarella, A. (2009) 'Energy conservation in wireless sensor networks: a survey', *Ad Hoc Network*, Vol. 7, pp.537–568.

Babayo, A.A., Anisi, M.H. and Ali, I. (2017) 'A review on energy management schemes in energy harvesting wireless sensor networks', *Renewable and Sustainable Energy Reviews*, Vol. 76, pp.1176–1184.

Barati, A., Dehghan, M., Movaghar, A. and Barati, H. (2008) 'Improving fault tolerance in ad-hoc networks by using residue number system', *Journal of Applied Sciences*, Vol. 8, pp.3273–3278.

Barati, A., Movaghar, A. and Sabaei, M. (2016) 'RDTP: reliable data transport protocol in wireless sensor networks', *Telecommunication Systems*, Vol. 62, pp.611–623.

Chen, S., Xu, H., Liu, D., Hu, B. and Wang, H. (2014) 'A vision of IoT: applications, challenges, and opportunities with China perspective', *IEEE Internet Things of Journal*, Vol. 1, pp.349–359.

Ghosh, A. and Das, S.K. (2008) 'Coverage and connectivity issues in wireless sensor networks: a survey', *Pervasive and Mobile Computing Journal*, Vol. 4, pp.303–334.

Javanmardi, S., Barati, A., Dastgheib, S.J. and Attarzadeh, I. (2012) 'A novel approach for faulty node detection with the aid of fuzzy theory and majority voting in wireless sensor networks', *International Journal of Advanced Smart Sensor Network Systems*, Vol. 2, pp.1–10.

Jodeh, H., Mikkawi, A., Awad, A. and Othman, O. (2018) 'Comparative analysis of routing protocols for under-water wireless sensor networks', *Proceedings of the 2nd International Conference on Future Networks and Distributed Systems*, Amman, Jordan, pp.1–7.

Ketshabetswe, L.K., Zungeru, A.M., Mangwala, M., Chuma, J.M. and Sigweni, B. (2019) 'Communication protocols for wireless sensor networks: a survey and comparison', *Heliyon*, Vol. 5. Doi: 10.1016/j.heliyon.2019.e01591.

Khazaei, E., Barati, A. and Movaghar, A. (2009) 'Improvement of fault detection in wireless sensor networks', *Proceedings of the ISECS International Colloquium on Computing, Communication, Control, and Management*, Sanya, China, Vol. 4, pp.644–646.

Li, M. and Yang, B. (2006) 'A survey on topology issues in wireless sensor network', *Proceedings of the International Conference on Wireless Networks*, Las Vegas, NV, USA, p.503.

Muller, C. and Valle, M. (2010) 'System verification of Flexray communication networks through behavioral simulations', *Proceedings of the IEEE International Behavioral Modeling and Simulation Workshop*, San Jose, CA, USA, pp.1–6.

Parashar, V., Mishra, B. and Tomar, G. (2020) 'Energy aware communication in wireless sensor network: a survey', *Materials Today: Proceedings*, Vol. 29, pp.512–523.

Pranitha, B. and Anjaneyulu, L. (2014) 'Analysis of underwater acoustic communication system using equalization technique for ISI reduction', *Procedia Computer Science*, Vol. 167, pp.1128–1138.

Sah, D.K. and Amgoth, T. (2018) 'Parametric survey on cross-layer designs for wireless sensor networks', *Computer Science Review*, Vol. 27, pp.112–134.

Stanley-Marbell, P., Basten, T., Rousselot, J., Oliver, R.S., Karl, H., Geilen, M., Hoes, R., Fohler, G. and Decotignie, J.D. (2008) *System Models in Wireless Sensor Networks*; Eindhoven University of Technology, Eindhoven, The Netherlands, pp.1–29. Available online at: https://research.tue.nl/files/2965278/710928.pdf (accessed on 20 November 2022).

Waite, A.D. (2002) *Sonar for Practising Engineers*, Wiley, Hoboken, NJ, USA.

Yoneki, E. and Bacon, J. (2005) *A Survey of Wireless Sensor Network Technologies*, UCAM-CL-TR-646. Available online at: https://www.academia.edu/download/42844186/Survey_of_Wireless_Sensor_Networks_UC AM-CL-TR-646.pdf (accessed on 2 November 2022).

Yuan, D., Kanhere, S.S. and Hollick, M. (2017) 'Instrumenting wireless sensor networks – a survey on the metrics that matter', *Pervasive and Mobile Computing Journal*, Vol. 37, pp.45–62.

Zhang, H. and Cuiping, L. (2012) 'A review on node deployment of wireless sensor network', *International Journal of Computer Science Issues*, Vol. 9, pp.378–383.