Energy evaluation of dependent malicious nodes detection in Arduino-based internet of things networks

Moath Alsafasfeh[1,2], Abdullah Alhasanat[1], Samiha Alfalahat[1]
[1]Department of Computer Engineering, College of Engineering, Al-Hussein Bin Talal University, Ma'an, Jordan
[2]Department of Electrical and Computer Engineering, College of Engineering, Tuskegee University, Alabama, United States

Article Info

Article history:

Received Nov 26, 2024
Revised Jun 23, 2025
Accepted Jul 12, 2025

Keywords:

Arduino
Energy efficiency
Internet of things
Malicious detection
Wireless sensor network

ABSTRACT

Detection of malicious nodes in the internet of things (IoT) network consumes power, which is one of the main constraints of the IoT network performance. To evaluate the energy-security trade-off for malicious node detection, this paper proposes an Arduino-based system for dependent malicious nodes (DMN) detection. The experimental work using Arduino and radio frequency (RF) modules was implemented to detect dependent malicious nodes in an IoT network. The detection algorithms were evaluated in terms of energy efficiency. The experiment comprises a coordinator node with five sensor nodes and varying malicious nodes. The results assess the detection algorithms in terms of distinguishing between normal and malicious behaviors and their impact on energy efficiency. The experiment demonstrated that the detection system could identify the malicious nodes. Additionally, the effect of increasing the number of sensors or malicious nodes on the suggested detection algorithm's energy usage is evaluated.

Corresponding Author:

Samiha Alfalahat
Department of Computer Engineering, College of Engineering, Al-Hussein Bin Talal University
Mohammad Alkhattab St., Ma'an, Jordan
Email: samiha mosa@ahu.edu.jo

1. INTRODUCTION

The internet of things (IoT) emerges as a prominent technology reshaping the digital landscape, facilitating seamless communication among myriad interconnected nodes via a shared network [1], [2]. With its versatility, IoT finds applications across various domains, notably in enhancing target detection within military and security domains [3]. Specifically, wireless sensor networks (WSNs) are deployed to monitor potential targets, known as target-detection WSNS (TD-WSNs), wherein sensor nodes collect data on target activities [4]–[6]. The gathered data is centralized at a central entity (CE) responsible for determining the target's status. However, the presence of malicious nodes can compromise detection accuracy by injecting false data [7]. As is often known, IoT devices are powered by batteries; however, changing and recharging them may be costly and tedious [8]. Thus, power consumption is a crucial factor that affects IoT system performance [9], [10]. The presence of a malicious entity may impact both the performance of the IoT network and the power consumption of all participating nodes. Identifying malicious nodes is critical for system and network operation [11]; thus, it is vital to discover unusual behavior in IoT devices and develop adaptive and creative anomaly detection algorithms or methods for detecting hazardous nodes [12]. Our proposed architecture includes several distributed nodes that broadcast their decisions depending on the target status to the central authority via time division multiple access (TDMA) scheduling. The central authority collects these decisions and applies the K-of-N rule to determine the global state of the target [13].

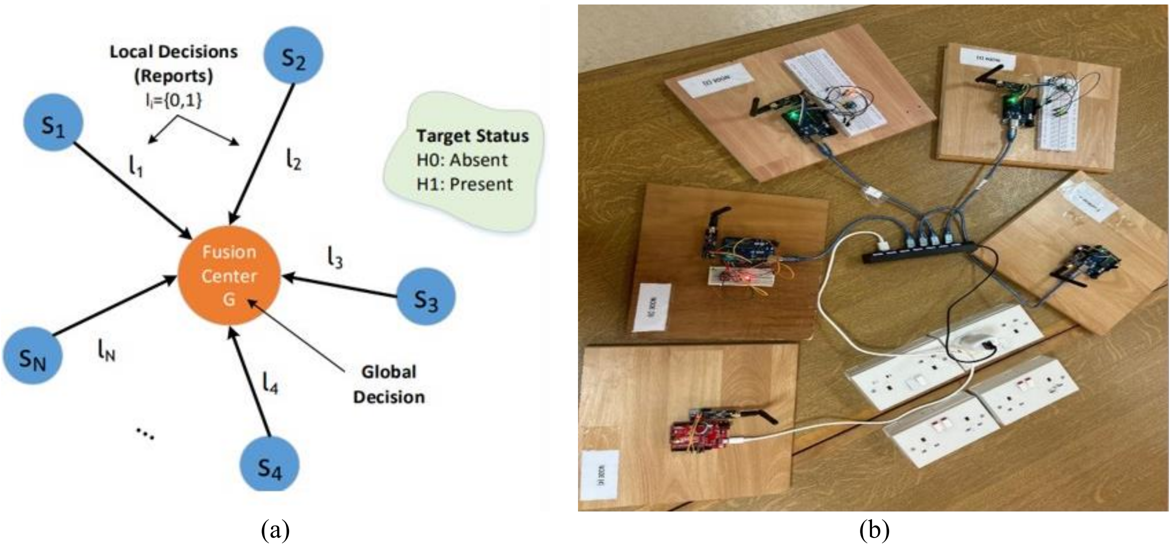
A sophisticated form of malicious nodes, termed dependent malicious nodes (DMNs), has garnered recent attention [7], [14]–[16]. DMNs adjust their attack strategy based on others' performance, making detection challenging. These nodes listen to sensing results from other nodes, altering their own data to influence global decisions only when necessary [17], [18]. A study in [19] emphasizes the importance of the number of nodes a DMN can hear. To detect DMNs in TD-WSNs utilizing a TDMA reporting style, a scheme is proposed to change reporting orders and monitor node performance. Nodes showing divergent performance when reporting order changes are identified as DMNs. While this scheme shows promise, it primarily addresses binary target states, whereas practical applications often involve multiple states. Another study extends this scheme to multistate TD-WSNs [13].

Balbudhe *et al.* [20] proposed a project that intends to reduce industrial energy consumption by increasing IoT for remote energy parameter monitoring systems. The system includes a microcontroller, global positioning system (GPS) and global system for mobile communications (GSM) system, smart meter, internet of things device, and current transformer (CT). The project's purpose is to remove excessive human labor required for energy audits while also providing a complete energy monitoring solution. The study [21] aims to develop an industrial internet of things (IIoT) and edge computing based system for monitoring energy use in a manufacturing floor using wireless and wired energy meters. The system uses the message queuing telemetry transport (MQTT) protocol to deliver data at a one-minute interval, which is saved on a database server and analyzed by an edge instance to extract analytical metrics, focusing on kilowatt-hour (kWh) for comparison analysis. The study found that deactivating data processing reduced central processing unit (CPU) utilization but maintained constant memory usage, suggesting the system could improve corporate edge–fog computing technologies for remote applications. The study in [22] used energy monitoring (EnerMon), an internet of things long range (LoRa) system, to track power use and waste in multiple places. The investigation discovered wasteful power use in auditoriums, pool heaters, water pumps, and electric boilers, stressing the need for more effective and efficient energy management systems.

Abba *et al.* [23] have developed a low-cost autonomous sensor interface for a smart IoT-based irrigation monitoring and control system. The system senses the environment and reacts based on sensed data, allowing for dynamic system management without human intervention. A microcontroller receives signals from soil sensors, turning off relay circuits controlling the water pump. The data is sent to a cloud for user viewing. Maintaining moisture levels between 100% and 400% is crucial for efficient irrigation. Kanakaris *et al.* [24] presents an IoT system that monitors temperature and luminosity in a data center using MQTT. It analyzes power consumption by Wemos, a firmware application in C, and uploads data to the NodeRed MQTT Broker. The NodeMCU serves as a station between routers and nodes, receiving data from Wemos and loaded onto Raspberry Pi 2. The system displays real-time data, including power usage and missed packets, and prevents retransmissions to extend battery life. The system demonstrated comparable power consumption performance over 21 iterations. The platform designed in [25] manages solar-powered wireless sensor nodes in industrial IoT applications, focusing on low-cost voltage sensor accuracy. It checks and analyzes Arduino prototypes. The experiment focuses on sensor accuracy and voltage-related metrics. The solution improves the efficiency of solar power generation and offers optimal IIoT operation settings. The results indicate that it can properly estimate solar panel production. This paper elaborates on the previous studies [19], [13] by conducting a practical testbed using Arduino nodes and radio frequency (RF) modules. The main objectives of the experiment are to assess various dependent malicious nodes' impact on target-detection wireless sensor networks, explore effective parameters influencing TD-WSN performance, and evaluate the energy efficiency of a malicious detection algorithm proposed in the work conducted in [13]. Abedin *et al.* [26] offer an energy-efficient technique for scheduling the duty cycles of various sensors in green internet of things (Green-IoT) systems. The algorithm operates in three phases: on-duty, pre-off duty, and off-duty. On-duty devices have full functionality, whereas pre-off devices have reduced computational capabilities. Off-duty states conserve energy in a variety of ways, including hibernation, sleep, and power-down. The suggested energy-efficient algorithm successfully schedules the duty cycle of numerous sensors and appliances, resulting in fewer devices and higher service quality. Another study in [27] demonstrates long range wide area network (LoRaWAN)'s appropriateness for low-power, long-range networks, establishing it as a feasible protocol for Internet of Things applications. It minimizes transmission time, data rate, and bit error rate while optimizing battery life, spreading factor (SF), and bandwidth (BW).

2. METHOD

The proposed system architecture comprises three primary parts: sensor nodes, RF communication links, and a coordinator, as shown in Figure 1(a) and 1(b). In this experimental setup, the system consists of five nodes in total: four normal and one malicious node. Each node makes its decision based on the sensed data and communicates it to the coordinator node via RF wireless communication. The coordinator is responsible for aggregating and analyzing these decisions and identifying potential malicious behavior. In the following sections, the structure of the main types of nodes, sensor nodes, malicious nodes, and the coordinator, is explained in detail.

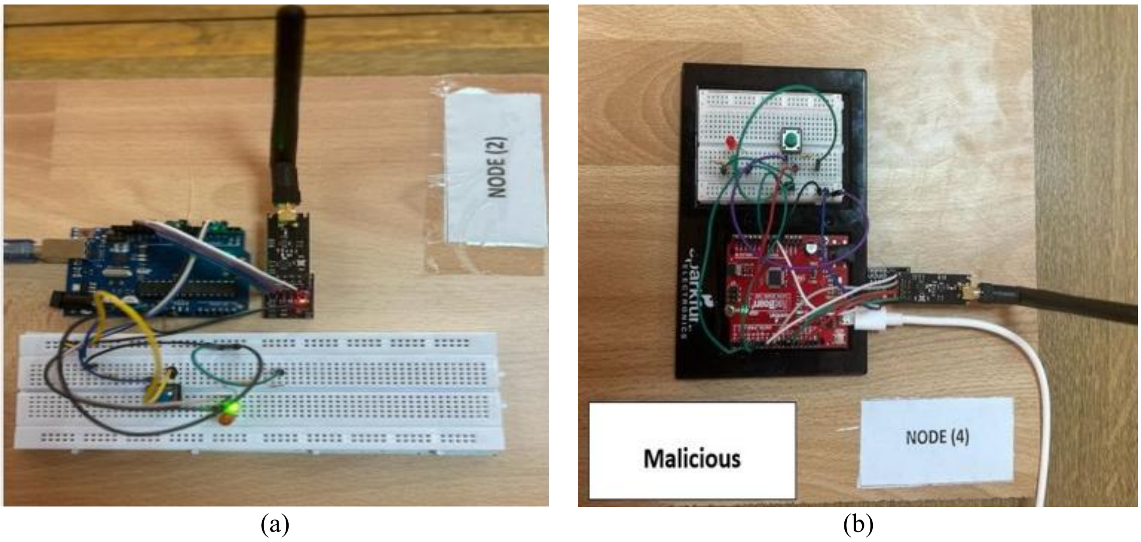


(a) (b)

Figure 1. The proposed system architecture (a) system model (b) implemented model

2.1. Sensor nodes

The normal sensor node consists of Arduino Uno, a pushbutton, and an RF transceiver as shown in Figure 2(a) and 2(b). Arduino Uno is used as the main microcontroller for the sensor node. Instead of using a sensor to sense the target, we have used push buttons and LEDs to indicate the target status as present or absent. The red color indicates the target is present while the green color indicates that the target is absent. The RF transceiver is used to send a node's local decision every communication round to the coordinator. We have used RF24Ln as an RF transceiver model in this experiment.



(a) (b)

Figure 2. Sensor nodes (a) normal sensor node (b) malicious node

2.2. Malicious nodes

The malicious node consists of Arduino Uno, LEDs, and RF transceivers. Arduino Uno is used as the main microcontroller of the malicious node. The LED is used to indicate the output of the target status as present or absent. The red color indicates the target is present, while the green color indicates that the target is absent. The RF transceiver is used to send a malicious decision to the coordinator during every communication round. The malicious node could be either static (independent) or dynamic (dependent). The static malicious nodes could further appear as independent always-one (IAO), independent always-zero (IAZ), or independent always-false (IAF). Pseudo-codes of Algorithm 1 show these malicious types' of functions, respectively.

Algorithm 1 yields the behavior of an independent malicious sensing node. The IAO node sends a state of the present target regarding the actual sensing, yielding a high false alarm probability but a high detection probability when the target is present. In contrast, the IAZ node sends an absent target where the system generates no false alarms. Yet, it has poor detection since the target is continually ignored. In Algorithm 1, the node tries to flip the true target status by IAF node sending an incorrect state, and this is the most disruptive tactic.

Algorithm 1. Independent malicious node types

```
if $M_{type}$ == lo then                    ▷ Always-One type
    Decision ← 1
else if $M_{type}$ == lz then               ▷ Always-Zero Type
    Decision ← 0
else if $M_{type}$ == lf then               ▷ Always-False Type
    Decision ← False Decision
end if
```

Similarly, the dynamic malicious nodes may act as dependent always-one (DAO), dependent always-zero (DAZ), or dependent always-false (DAF). Pseudo-codes 4 to 6 show the function of these malicious types, respectively. The Algorithms 2, 3, and 4 demonstrate the behavior of a dependent malicious node based on always-one, always-zero, and always-false, in which this node decides to send one or zero only when its decision can influence the global decision taken by the coordinator, and send the wrong decision only when it can impact the outcome. By adopting this behavior, the node acts maliciously selectively to avoid detection. In our experimental scenarios, various numbers and types of malicious nodes have been used. For the previous algorithms, initialization should be started by defining: the number of nodes N, $Target_{status}$, $True_{decision}$, $False_{decision}$, M_{order}, M_{type}, and $R_{decision}$.

Algorithm 2. DAO malicious node types

```
if $M_{type}$ == DAO then
    if $M_{order} \le \frac{N}{2}$ then
        Decision ← 1
    else
        s ← 0
        for i = 1 to $M_{odrder} - 1$ do
            if Rx-Decision[i − 1] == TRUE-Decision then
                s ← s + 1
            end if
        end for
        if True-Decision == 0 and s > $\frac{N}{2}$ then
            Decision ← True-decision
        else
            Decision ← 1
        end if
    end if
end if
```

2.3. Coordinator

The coordinator node consists of an Arduino Mega and an RF transceiver. Arduino Mega is used as the main microcontroller of the coordinator node. The coordinator node's main task is to receive the nodes' local decisions to make global decisions. In addition, the coordinator node is responsible for detecting and identifying malicious nodes. As shown in Algorithm 5, after initialization, the coordinator sends a beacon message to the selected node, including the node identifier. Hence, all nodes identify the order and return their decision to the coordinator. A decision buffer is updated with fresh decision values. The coordinator then checks if all nodes have been contacted before proceeding to the next round. After completing the maximum number of rounds, the malicious node detection mechanism is enabled, where there is a shuffle in the nodes' order for the next turn. Finally, reporting is resumed.

Algorithm 3. DAZ malicious node types

```
if $M_{type}$ == DAZ then
    if $M_{order} \le \frac{N}{2}$ then
        Decision ← 0
    else
        s ← 0
        for i = 1 to $M_{order}$ − 1 do
            if Rx-Decision[i − 1] == TRUE-Decision then
                s ← s + 1
            end if
        end for
        if True-Decision == 1 and s > $\frac{N}{2}$ then
            Decision ← True-decision
        else
            Decision ← 0
        end if
    end if
end if
```

Algorithm 4. DAF malicious node types

```
if $M_{type}$ == DAF then
    if $M_{order} \le \frac{N}{2}$ then
        Decision ← False-Decision
    else
        s ← 0
        for i = 1 to $M_{order}$ − 1 do
            if Rx-Decision[i − 1] == TRUE-Decision then
                s ← s + 1
            end if
        end for
        if True-Decision == 0 and s > $\frac{N}{2}$ then
            Decision ← True-Decision
        else if True-Decision == 1 and s > $\frac{N}{2}$ then
            Decision ← True-Decision
        else
            Decision ← False-Decision
        end if
    end if
end if
```

Algorithm 5. Coordinator

```
Initialization:
Define Number of Nodes (N ), Max Iteration ($T_{max}$), Initial Iteration ($T_n$ = 0), index
Start Reporting
while $T_n < T_{max}$ do
    for index = 0 to N − 1 do
        Send message to node with data.ID
        Receive decision from the node
    end for
    if index > (N-1) then
        Proceed to the next round
        Reset index
        $T_n \leftarrow T_n + 1$
    end if
    if $T_n \ge T_{max}$ then
        Perform malicious node detection
        Shuffle the order of nodes for the next turn
        $T_n \leftarrow 0$
        Resume Reporting
    end if
end while
```

In Algorithm 6, the malicious node type is identified by calculating the probability of one for each node. There are three main categories: independent malicious, dependent malicious, and normal node. The node is classified as a dependent malicious node if the current probability value is greater than the previous value plus a threshold value or less than the previous value minus the threshold. If this condition is met, then the probability magnitude determines the dependent malicious, whether it's dependent always one, dependent always zero, or dependent always false. If the condition is not met, then the node is classified as an independent malicious node. Also, the magnitude of the likelihood defines the Independent malicious node type, which is the same as the dependent malicious. The node is considered a normal node if the behavior is within an acceptable range around the target probability H0.

Algorithm 6. Malicious detection

```
Initialization:
Define: σ=0, Threshold (Δ), Max Iterations ($T_{max}$), Target Probability (H0), Independent
One (IO), Independent-Zero (Iz), Independent-False (If), Dependent One (Do), Dependent
Zero (Dz), Dependent False (Df), Number of Nodes (Nm)
for each node i in N do
    for each round j up to $T_{max}$ do
        Calculate new value of σ
        σ ← σ + D buff[j][i]
        Estimate $P_{i,n}$
        if $Pi,n > (Pi,n,temp + Δ)$ or $Pi,n < (Pi,n,temp - Δ)$ then
           if $P_{i,n}$ == 0 then
              Dz ← Dz + Sn
           else if $P_{i,n}$ == 1 then
              Do ← Do + Sn
           else
              Df ← Df + Sn
           end if
        else
           if $P_{i,n}$ == 0 then
              lz ← lz + Sn
           else if $P_{i,n}$ == 1 then
              I0 ← I0 + Sn
           else if $(H0 - Δ) ≤ P_{i,n} ≤ (H0 + Δ)$ then
              If ← If + Sn
           else
              Nm ← Nm + Sn
           end if
        end if
    end for
end for
```

3. RESULTS AND DISCUSSION

In this section, the performance of the proposed malicious detection algorithm is assessed. The evaluation process considers the accuracy of dependent malicious node detection and the energy cost associated with the detection process. The required energy consumption for successful detection is measured to evaluate the energy efficiency (EE) of the system. The evaluation included experiments conducted with different numbers of sensors and different malicious nodes.

3.1. Performance of the detection algorithm

The performance of the detection algorithm 6 is evaluated for each node individually. The performance is measured in terms of the P1 value, which according to [19] calculated using 1.

$$P1 = \frac{\sum_{i=1}^{T} l_i}{T} \tag{1}$$

where *l* denotes the local decision of a node at round *i*, and T is the maximum number of rounds. Two scenarios are considered. In the first scenario, we consider three normal sensor nodes and two malicious nodes with one IAF and IAZ. Note that the target status is set to 0, false-alarm probability of the target is 0.2. The performance of each node is plotted in Figure 3. Since the target is assumed absent and the false-alarm is 0.2, it is expected that the value of P1 of normal nodes will be nearly 0.2; otherwise, the node will be identified as malicious. It is clear from this figure that Node 1, 2, and 3 are behaving normally, however, Nodes 4 and 5 demonstrated malicious behavior. Specifically, the P1 value due to Node 5 is 0, indicating that the node is IAZ, while the P1 value due to Node 4 is nearly 0.8, indicating that this node is malicious or IAF.

In the second scenario, nodes 3 and 4 act as DAZ and DAF; respectively, while other nodes are kept normal. Then the performance of all nodes is plotted in Figure 4. It is clear from this figure that Nodes 1, 2, and 5 have almost identical performance. On the other hand, Nodes 3 and 4 have abnormal behavior.

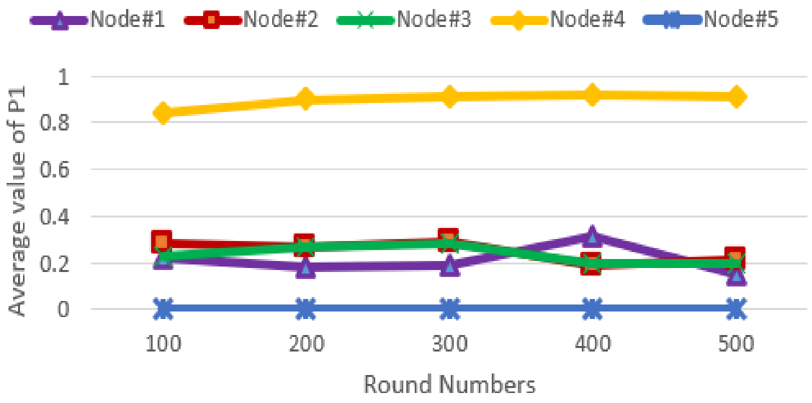


Figure 3. Performance of nodes in terms of P1 at every 100 rounds (3 normal nodes and two malicious with IAZ and IAF)

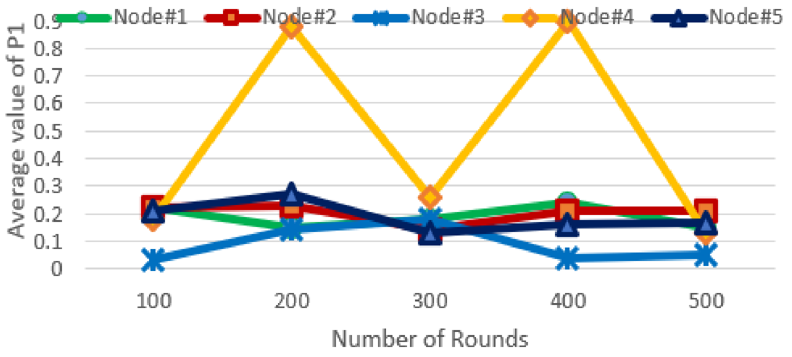


Figure 4. Performance of nodes regarding P1 at every 100 rounds (3 normal nodes and two malicious with DAZ and DAF)

3.2. Impact of the number of nodes on EE

In the first measurement set, we study the impact of increasing the number of nodes on the energy consumption of the coordinator. As shown in Figure 5, the energy consumption increased from about 0.75 Wh at three nodes to 2 Wh at five nodes. On the other hand, it seems that the number of malicious nodes has no significant impact on energy consumption. This is clear for both three and five-node scenarios.

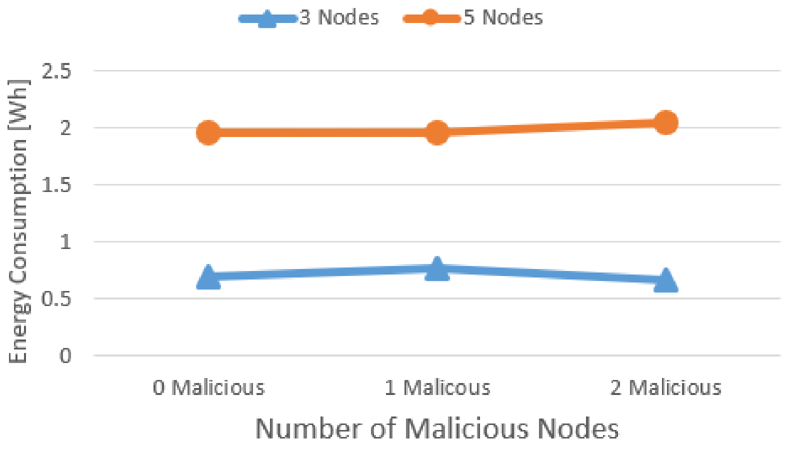


Figure 5. Number of malicious nodes versus energy consumption at three and five nodes scenarios

3.3. Impact of using malicious detection on EE

Next, we study the impact of using a malicious detection algorithm on the EE. Figure 6 shows the energy consumption of the three and five-node scenarios: one with the malicious detection algorithm and the other without. It is clear that when there is a small number of nodes (e.g., three nodes scenario) there is a negligible difference in energy consumption with and without the presence of a malicious detection algorithm. However, when the number of nodes increased to five nodes, the malicious detection algorithm increased energy consumption from 1 to 1.3 Wh, which is about 30%.

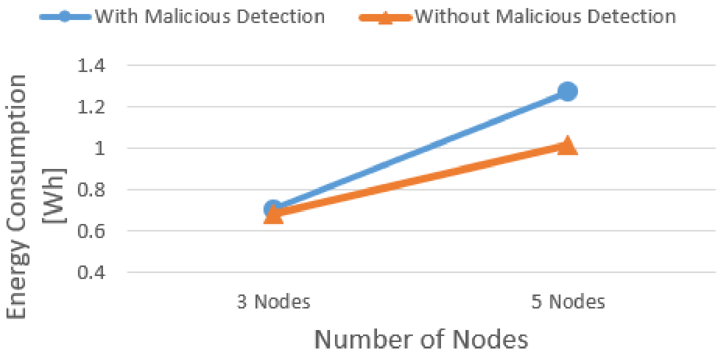


Figure 6. Number of sensor nodes versus energy consumption with and without malicious detection algorithms

4. CONCLUSION

In this paper, an experimental study was conducted to evaluate the performance of the malicious node detection and identification algorithms proposed in previous research. The primary objective was to assess both the accuracy and energy efficiency of these algorithms within an IoT Network. The experimental setup included one coordinator and five sensor nodes, with various numbers and types of malicious nodes introduced to simulate realistic attack scenarios. The results demonstrated that the detection algorithm was effective in distinguishing between normal and malicious nodes, successfully identifying both the presence and type of malicious behavior. Additionally, the energy efficiency of the algorithm was evaluated, with the analysis revealing how its energy consumption varied with changes in network configurations specifically when increasing the number of malicious nodes or the total number of sensor nodes. These findings offer valuable insights into the scalability and practical deployment of the proposed method in larger or more dynamic WSN environments. The results validate the algorithm’s effectiveness in terms of both accuracy and energy efficiency. Future enhancements could focus on improving detection speed and testing the algorithm under a broader range of operational and environmental conditions.

ACKNOWLEDGEMENTS

The authors would like to thank the NATO/Science for Peace and Security Program for co-funding the project, Developing Physical Layer Security Schemes for Internet of Things Networks, under grant number SPS G5979.

REFERENCES

[1] F. Li and P. Xiong, “Practical secure communication for integrating wireless sensor networks into the internet of things,” *IEEE Sensors Journal*, vol. 13, no. 10, pp. 3677–3684, Oct. 2013, doi: 10.1109/JSEN.2013.2262271.

[2] D. Princy, D. Kalaivani, and T. Vijayaraghavan, “A design thinking approach of metaheuristic empowerment for energy-efficient and optimized routing protocol in IoT-enabled wireless sensor networks,” in *Proceedings - 3rd International Conference on Smart Technologies, Communication and Robotics 2023, STCR 2023*, Dec. 2023, pp. 1–6. doi: 10.1109/STCR59085.2023.10396880.

[3] P. Spachos and D. Hatzinakos, “Real-time indoor carbon dioxide monitoring through cognitive wireless sensor networks,” *IEEE Sensors Journal*, vol. 16, no. 2, pp. 506–514, Jan. 2016, doi: 10.1109/JSEN.2015.2479647.

[4] Y. Yu, F. Han, Y. Bao, and J. Ou, “A study on data loss compensation of WiFi-based wireless sensor networks for structural health monitoring,” *IEEE Sensors Journal*, vol. 16, no. 10, pp. 3811–3818, May 2016, doi: 10.1109/JSEN.2015.2512846.

[5] R. Lara, D. Benítez, A. Caamaño, M. Zennaro, and J. L. Rojo-Álvarez, “On real-time performance evaluation of volcano-monitoring systems with wireless sensor networks,” *IEEE Sensors Journal*, vol. 15, no. 6, pp. 3514–3523, Jun. 2015, doi: 10.1109/JSEN.2015.2393713.

[6] R. Viswanathan and V. Aalo, “On counting rules in distributed detection,” *IEEE Transactions on Acoustics, Speech, and Signal Processing*, vol. 37, no. 5, pp. 772–775, May 1989, doi: 10.1109/29.17574.

[7] X. Chen, K. Makki, K. Yen, and N. Pissinou, “Sensor network security: A survey,” *IEEE Communications Surveys and Tutorials*, vol. 11, no. 2, pp. 52–73, 2009, doi: 10.1109/SURV.2009.090205.

[8] A. Shahini, A. Kiani, and N. Ansari, “Energy efficient resource allocation in EH-Enabled CR networks for IoT,” *IEEE Internet of Things Journal*, vol. 6, no. 2, pp. 3186–3193, Apr. 2019, doi: 10.1109/JIOT.2018.2880190.

[9] F. Li, Y. Shi, A. Shinde, J. Ye, and W. Song, “Enhanced cyber-physical security in internet of things through energy auditing,” *IEEE Internet of Things Journal*, vol. 6, no. 3, pp. 5224–5231, Jun. 2019, doi: 10.1109/JIOT.2019.2899492.

[10] K. Nair *et al.*, “Optimizing power consumption in iot based wireless sensor networks using Bluetooth Low Energy,” in *2015 International Conference on Green Computing and Internet of Things (ICGCIoT)*, Oct. 2015, pp. 589–593. doi: 10.1109/icgciot.2015.7380533.

[11] B. Li, R. Ye, G. Gu, R. Liang, W. Liu, and K. Cai, “A detection mechanism on malicious nodes in IoT,” *Computer Communications*, vol. 151, pp. 51–59, Feb. 2020, doi: 10.1016/j.comcom.2019.12.037.

[12] K. K. S. Liyakat, “Detecting malicious nodes in IoT networks using machine learning and artificial neural networks,” in *2023 International Conference on Emerging Smart Computing and Informatics, ESCI 2023*, Mar. 2023, pp. 1–5. doi: 10.1109/ESCI56872.2023.10099544.

[13] S. Mousa, S. Althunibat, A. Alhasanat, and M. Alsafasfeh, “Detecting dependent malicious nodes in multi-state target detection wireless sensor networks,” in *2023 International Telecommunications Conference, ITC-Egypt 2023*, Jul. 2023, pp. 657–662. doi: 10.1109/ITC-Egypt58155.2023.10206285.

[14] A. Antonopoulos and C. Verikoukis, “Misbehavior detection in the internet of things: a network-coding-aware statistical approach,” in *IEEE International Conference on Industrial Informatics (INDIN)*, Jul. 2016, pp. 1024–1027. doi: 10.1109/INDIN.2016.7819313.

[15] Y. Zhang, N. Meratnia, and P. Havinga, “Outlier detection techniques for wireless sensor networks: A survey,” *IEEE Communications Surveys and Tutorials*, vol. 12, no. 2, pp. 159–170, 2010, doi: 10.1109/SURV.2010.021510.00088.

[16] Y. Wang, G. Attebury, and B. Ramamurthy, “A survey of security issues in wireless sensor networks,” *IEEE Communications Surveys and Tutorials*, vol. 8, no. 2, pp. 2–22, 2006, doi: 10.1109/COMST.2006.315852.

[17] Y. Sei and A. Ohsuga, “Malicious node detection in mobilewireless sensor networks,” *Journal of Information Processing*, vol. 23, no. 4, pp. 476–487, 2015, doi: 10.2197/ipsjjip.23.476.

[18] A. Ahmad, M. Hababeh, A. Abu-Hantash, Y. AbuHour, and H. Musleh, “Reduce effect of dependent malicious sensor nodes in WSNs using pairs counting and fake packets,” *INTERNATIONAL JOURNAL OF COMPUTERS COMMUNICATIONS &amp; CONTROL*, vol. 15, no. 5, Aug. 2020, doi: 10.15837/ijccc.2020.5.3825.

[19] S. Althunibat, A. Antonopoulos, E. Kartsakli, F. Granelli, and C. Verikoukis, “Countering intelligent-dependent malicious nodes in target detection wireless sensor networks,” *IEEE Sensors Journal*, vol. 16, no. 23, pp. 8627–8639, 2016, doi: 10.1109/JSEN.2016.2606759.

[20] P. Balbudhe, B. Jais, P. Burbure, K. Kalbende, S. Warkad, and A. Hemane, “Design and implementation of energy audit with IoT and Arduino,” *International Journal of Analytical, Experimental and Finite Element Analysis (IJAEFEA)*, vol. 6, no. 1, Apr. 2019, doi: 10.26706/ijaefea.1.6.20190308.

[21] A. A. Mirani, A. Awasthi, N. O’Mahony, and J. Walsh, “Industrial IoT-based energy monitoring system: using data processing at edge,” *Internet of Things*, vol. 5, no. 4, pp. 608–633, Sep. 2024, doi: 10.3390/iot5040027.

[22] D. Santos and J. C. Ferreira, “IoT power monitoring system for smart environments,” *Sustainability (Switzerland)*, vol. 11, no. 19, Sep. 2019, doi: 10.3390/su11195355.

[23] S. Abba, J. W. Namkusong, J. A. Lee, and M. L. Crespo, “Design and performance evaluation of a low-cost autonomous sensor interface for a smart iot-based irrigation monitoring and control system,” *Sensors (Switzerland)*, vol. 19, no. 17, p. 3643, Aug. 2019, doi: 10.3390/s19173643.

[24] V. Kanakaris, G. A. Papakostas, and D. V. Bandekas, “Power consumption analysis on an IoT network based on wemos: A case study,” *Telkomnika (Telecommunication Computing Electronics and Control)*, vol. 17, no. 5, pp. 2505–2511, Oct. 2019, doi: 10.12928/TELKOMNIKA.v17i5.11317.

[25] D. Dobrilovic *et al.*, “Data acquisition for estimating energy-efficient solar-powered sensor node performance for usage in industrial IoT,” *Sustainability (Switzerland)*, vol. 15, no. 9, p. 7440, Apr. 2023, doi: 10.3390/su15097440.

[26] S. F. Abedin, M. G. R. Alam, R. Haw, and C. S. Hong, “A system model for energy efficient green-IoT network,” in *International Conference on Information Networking*, Jan. 2015, vol. 2015-January, pp. 177–182. doi: 10.1109/ICOIN.2015.7057878.

[27] C. Bouras, V. Kokkinos, and N. Papachristos, “Performance evaluation of LoraWan physical layer integration on IoT devices,” in *2018 Global Information Infrastructure and Networking Symposium, GIIS 2018*, Oct. 2018, pp. 1–4. doi: 10.1109/GIIS.2018.8635715.

BIOGRAPHIES OF AUTHORS

Moath Alsafasfeh is an assistant professor at Tuskegee University, Tuskegee, Alabama, USA. He was an associate professor of electrical and computer engineering at Al-Hussein Bin Talal University, Jordan, from 2017 until August 2024. He earned his Ph.D. from Western Michigan University (WMU) in 2017, an M.Sc. in communication and computer engineering from the National University of Malaysia (UKM) in 2011, and a B.Sc. degree in computer engineering from Mutah University in 2009. Alsafasfeh’s research interests are in parallel processing, computer vision, image processing, and machine learning for wireless sensor networks (WSNs) and internet of things (IoT) network applications. Alsafasfeh has published more than 24 articles in prestigious Scopus-indexed journals. Dr. Alsafasfeh was awarded several national prizes and scholarships to earn a Ph.D. and a bachelor’s degree. He has strong relationships with different local, regional, and international researchers. Alsafasfeh has been working as a co-PI for two internationally funded research projects and one project for capacity building in higher education. He can be contacted at email: malsafasfeh@tuskegee.edu.

Abdullah Alhasanat was born in Jordan in 1981. He received a B.Sc. degree in computer engineering from the University of Aden in Yemen in 2004, his M.Sc. degree in computer engineering from Jordan University of Science and Technology in Jordan in 2007, and his Ph.D. degree in wireless networks from the University of Newcastle in the UK in 2012. In 2012, he joined the Department of Computer Engineering at Al-Hussein Bin Talal University, Jordan. Currently, Dr. Alhasanat is a full professor in wireless networking. He has published many articles in high-level scientific journals and at international conferences. His main research interests include routing and localization in ad hoc networks, wireless sensor networks, mobile ad hoc networks (MANET), vehicular ad hoc networks (VANET), delay-tolerant networks, parallel computing, and signal and image processing. He can be contacted at email: abad@ahu.edu.jo.

Samiha Alfalahat holds an M.Sc. degree in computer engineering from Al-Yarmouk University, Irbid, Jordan, in 2015, and a B.Sc. degree in computer engineering from Jordan University of Science and Technology, Irbid, Jordan, in 2006. She currently serves as an instructor and research assistant at Al-Hussein Bin Talal University, Jordan, where she teaches various courses in computer systems, networking, and internet of things (IoT), including distributed systems, network programming, and embedded systems. Her research interests include IoT systems, machine learning, network automation, and cybersecurity. Samiha has contributed to European-funded projects such as the NATO PHYSEC initiative, focusing on physical layer security for IoT, and the IREEDER project, aimed at aligning Jordanian undergraduate curricula with EU standards. She is an IEEE member and a proactive contributor to curriculum development for engineering departments. She can be contacted at email: samiha_mosa@ahu.edu.jo.