International Journal of
INTELLIGENT SYSTEMS AND APPLICATIONS IN ENGINEERING
ISSN:2147-6799 www.ijisae.org Original Research Paper

Jamming Detection Technique Using Improved Exponentially Weighted Moving Average (IEWMA) Algorithm for WSN

S. G. Hymlin Rose[1], Selvaraj Janani[2]

Submitted: 25/05/2023 **Revised**: 15/07/2023 **Accepted**: 26/07/2023

Abstract*:* WSN is emerging field with various application areas including health monitoring, military operations, agriculture, and environmental monitoring, have recently come to light as extremely promising solutions. Given the potential widespread use of Wireless Sensor Networks (WSNs), there is a growing concern about the security vulnerabilities faced by WSN sensor nodes, particularly due to their deployment in resource-constrained and hostile environments. One prevalent and disruptive form of attack is the DoS attack, with the jamming attack being a subcategory. The jamming attack is in which radio frequency signals are emitted to interfere with and disrupt the normal functioning of sensor nodes in the WSN, leading to service denial. To tackle this concern, a suggested approach is presented, utilizing a sequential method rooted in the Statistical Process Control (SPC) method for the identification of jamming attacks. For more exact detection of variations sin the strength of jamming attacks, an Improved Exponentially Weighted Moving Average (IEWMA) technique is suggested. This is achieved by analyzing a parameter called the packet Break Advent Time (BAT), which is derived from packets received from sensor nodes. Through simulation experiments, outcomes demonstrate that proposed methodology can accurately and efficiently detect jamming attacks in the sensor network, with minimal overhead, reduced energy consumption, and high precision.

Keywords: *Jamming attack, Statistical Process Control, Improved Exponentially Weighted Moving Average, Break Advent Time.*

1. Introduction

WSN is collection of nodes that are accomplished of detecting and potentially controlling their environment, enabling communication between individuals or systems and the surrounding environment. WSNs also find utility in high-security applications[25]. They are especially useful in framework of IoT because of improvements in wireless and electrical transmission. The basic elements of wireless sensor nodes typically.

WSNs have gained significant recognition in a extensive collection of applications primarily due to their cost-effectiveness [1]. Some notable examples include military applications, habitat and environmental monitoring, medical field applications, civilian monitoring of vehicular networks, and home automation. These applications demonstrate the versatility and potential of WSNs in various domains [27].

The safeguarding procedure is important because the sensor network have two types of difficulties faced in network. One is the conventional security problems which are different from sensor network problems [30]. The medium of communication is prone to several categories of attacks. The attacks have been categorized based on power consumption, routing, privacy, service availability, and data integrity. Data integrity is a crucial property that ensures the secure transmission of information, while data confidentiality restricts access to authorized users only [2-5]. However, several attacks pose a threat to both integrity and confidentiality. These attacks include Denial of Service (DoS), node capture attack, and eavesdropping attack[6].

A DoS attack disrupts the entire network, causing services to collapse and cease functioning. Jamming, on the other hand, is a specific type of DoS attack that targets wireless sensor networks. Due to their susceptibility, it is imperative to secure these networks against jamming attacks. Such attacks can be initiated by external jammers or even by compromised internal nodes that may turn into adversaries in the future[35].

Generally, jamming can be defined as a deliberate act of interfering with wireless networks. This involves trying to prevent users from accessing network services. In Wireless Sensor Networks (WSNs), the jamming attack may occur by a group of malicious nodes within the network. These attacks are intended to obstruct or interrupt the dependable broadcast and reception of appropriate signals midst the nodes within the sensor network [8-10]. However, in the context of WSNs, the static network environment is also vulnerable to various threats and assaults, including sybil attacks, HELLO floods, selective forwarding, black holes, jamming, and wormholes[2].

A highly concerning type of attack in the realm of network security is the jamming attack, which poses a significant threat by severely disrupting and collapsing wireless sensor networks (WSNs)[7]. By sending out radio signals on the

[1] *Department of Electronics and Communication Engineering, R.M.D Engineering College, Kavaraipettai, Chennai 601 206, Tamil Nadu, India*
[2] *Department of Electronics and Communication Engineering, Periyar Maniammai Institute of Science & Technology (Deemed to be University), Vallam, Thanjavur 613 403, Tamil Nadu, India*
** Corresponding Author Email: hymlinrose@gmail.com*

similar frequency as the network nodes, a jammer effectively prevents communication between nodes in the network. As a result, there are significantly more failed packet re-transmissions and packet dropouts than usual[11]. Additionally, because WSNs operate in resource-limited contexts, jamming attacks hasten the energy depletion of nodes and ultimately cause network failure[34].

As a result, considering the aforementioned limitations, it becomes evident that the development and implementation of a clear-cut anti-jamming technique are imperative on a global scale [12-14]. Additionally, when a node affected by jamming emits a signal that is itself jammed, a certain subcategory of nodes within the clustered network becomes entirely obstructed. Consequently, the data packets transferred by the obstructed nodes are discarded, preventing their delivery to the intended destination [15-17]. This subsequently leads to sender nodes getting caught in a cycle of repeated packet re-transmissions, ultimately depleting energy reserves of nodes and diminishing network's lifespan. This progression eventually culminates in the network becoming non-operational. [11,32].

There exist multiple categories of jamming attacks. Among these, proactive jammers are prevalent type of jamming attack due to their ease of implementation. They emit jamming signals without considering traffic patterns, making them a widespread threat [19, 29]. However, they suffer from certain drawbacks, including susceptibility to damage, low likelihood of being detected, and energy limitations stemming from channel resources. Another class of jammers is the deceptive jammer, which consistently sends coherent data packets instead of emitting random data bits [21-24]. Distinguishing a deceptive jammer from a constant jammer is particularly challenging, as deceptive jammers transmit appropriate packets as an alternative of random data bits. In contrast, an uninterrupted signal is emitted by a constant jammer at regular intervals. Furthermore, there is the random jammer, which continuously emits a persistent jamming signal at arbitrary moments [25-28]. During the sleep phase, it remains inactive for a certain period before becoming active again to engage in jamming, after which it returns to the sleep state. [1,20].

This approach focuses on ensuring the effective operation of WSN even when jamming attack is present. The primary idea is to prevent the whole network from failing due to partial disruption, such as when just certain nodes are jammed. Research analyzes stationary WSN that is vulnerable to proactive jamming attacks. Depending on the network's stability, they identify jammed zone and establish an alternative route for the unaffected nodes by effectively isolating the jammed area.

Numerous metrics are accessible for identifying jamming attacks in data transmission scenarios. The metrics offer valuable information regarding the degree of jamming present within the network. A crucial metric utilized in this algorithm is the Packet Break Advent Time (BAT), which signifies the time interval between reception of packet and the subsequent packets within the WSN.

2. Related Works

Extensive utilization of WSN has led to investigation of jamming attacks, wherein harmful radio signals are transmitted. These attacks have been extensively explored in existing literature due to their capability to disturb genuine communication and exploit network resources.31. Due to the inherent characteristics of sensor nodes such as their structural design, vulnerable arrangement, and unpredictable routing protocols, they are prone to jamming attacks. Misra et al. utilized a centralized fuzzy-based system, as described in their research[1]. Received packet count, dropped packet count, and RSSI were the three inputs collected from each sensor node in the network to formulate this method. By comparing the actual RSSI reading to the projected RSSI value, the base station determined the received power and the severity of the jamming assault. These numbers were also used by the base station to calculate the SNR and PDPT, which served as additional inputs for the fuzzy inference system. This system offered a threshold jamming index from 0 to 100 that represented the degree of the jamming attack, from total jamming to no jamming. However, a significant drawback of this technique is that it only detects jamming attacks at the base station level.

In their work, Strasser et al [2] focused on identifying the source of error bits in individual packets by analyzing the RSSI. The objective of this method was to identify reactive jamming attacks within sensor networks by leveraging default data, minimal cabling, and predefined error codes. The experimental outcomes underscored the efficiency of this approach in identifying intricate reactive jamming attacks without introducing substantial extra load. Nevertheless, a notable limitation of this system lies in the restricted precision associated with the employed error codes.

During synchronisation phase, a method was developed by Spuhler et al.[3] to evaluate probability of successful packet delivery. By forecasting the probability of packet delivery, this method is intended to identify jammers that attack the physical layer of WSNs. This prediction is based on assessed chip error rate, which is consequent from the received preamble symbols.

On a similar note, Guan and Ge[4] devised a distributed technique for detecting multichannel jamming attacks in WSNs. Their method utilizes data collected from multiple channels to identify random attacks, forming an integral part of a homogeneous Markov Chain model. This Markov

Chain model characterizes a complex multichannel jamming system with two levels of switching. Two types of switching describe the variations in the probability of changing between states.

Cordero et al.[5] introduced a heterogeneous Wireless Sensor Network (WSN) and employed a non-cooperative game theory approach involving two players to assess jamming attacks. The game problem was addressed by considering signal interference and noise ratio at the receivers, with utilities being a crucial factor. The outcomes highlighted the significance of factoring in the contact distance between network components when designing techniques for detecting jamming attacks.

Mpitziopoulos and Gavalas [6] proposed a theoretical strategy for mitigating jamming attacks in WSNs by utilizing a hybrid approach named Frequency Hopping Spread Spectrum (FHSS). This approach involved the utilization of FHSS to create a series of channels within the 5 GHz band, covering a total of 51 frequency channels. The outcomes of simulations demonstrated that the nodes produced through this method could attain a satisfactory PDR while consuming less energy in a jammed WSN setting, to traditional configurations of sensor networks.

Alnifie et al. [7] presented the Multi-channel Exfiltration Protocol (MULEPRO) system, which detects jamming nodes by relocating data from the region affected by a radio-jammed DoS attack in a WSN. This system dynamically designates nodes within the jammed area to the attacker. While it demonstrates a commendable detection rate, it incurs a significant energy cost due to data evacuation in the jammed zone.

Del-Valle-Soto [8] explored the adaptability of widely used routing protocols, namely, AODV, MPH and DSR, to reactive jamming attacks. The authors anticipated a modified versions of these protocols that guarantee jamming detection by segregating the affected node and allowing routing protocols to adapt their paths accordingly. They also introduced QUJDA, which employs anomaly approach suitable for distributed node deployment [9]. QUJDA distinguishes between networks with active jamming nodes and those without by utilizing metrics. Additionally, QUJDA encourages sensor nodes to communicate with their neighboring nodes, enhancing the overall detection rate.

In order to enhance the PDR and lifetime in a WSN multi-jammer attack, a new game theory approach based on Stackelberg game theory was proposed [7].Sensor nodes, acting as trackers, optimize their transmission power to protect neighboring nodes. By employing TC-JAM, the system aims to achieve a high network lifetime and efficient PDR in the existence of jamming attacks. It should be noted that it results in an increased energy consumption and reduced network lifetime[25].

In their study, Bhavathankar et al. [26] focused on preemptive jamming in WSNs. To mitigate this issue, they proposed a new evading mechanism that is link value aware. This technique attempts to avoid the network's congested regions by using a unique route selection approach between each source-destination pair. The goal is to increase network efficiency by avoiding the problematic areas where jamming is present. Lightweight security mechanisms for detecting and localising jamming attacks in wireless sensor networks were presented by Nashab Alikh and Amir Rajabzadeh in their 27 paper. The three-stage process they suggest. In this initial stage, we look for signs of a jamming assault. Step two entails using a mapping procedure to gauge the size of the affected area. Phase 3 involves using geometric calculations informed by the mapping methodology to pinpoint the precise location of the jammer. Improve WSN safety by quickly pinpointing the source of jamming assaults using this method.

In their paper[28], Pang and colleagues introduced a localization method based on particle swarm optimization (PSO). The goal was to optimize the localization accuracy in a WSN using PSO-based techniques. Sudha et al. [30] proposed a swarm intelligence algorithm to adapt to changes in network topology and traffic. This technique involved forwarding ants either in a unicast or broadcast manner at each node. If a channel is available, the ant chooses the next step accordingly. However, if the channel is not available, information from the backward ant is used to verify the presence of an attacker, allowing the node to avoid using that particular channel for transmission. The objective was to enhance the efficiency and reliability of communication in the presence of dynamic network conditions.

Feriel Cherifi et al.[31] introduced an efficient and lightweight anti-jamming protocol, called CFD-BD. This protocol combines FHSS and DSSS techniques. It is specifically designed for healthcare and medical environments. The CFD-BD protocol aims to mitigate the effects of jamming attacks by utilizing a combination of FHSS and DSSS, providing improved resistance against interference and ensuring secure and reliable communication.

Xiao et al[32] put forth a mobile offloading scheme based on reinforcement learning for mobile edge computing systems, designed to mitigate the impact of jamming attacks. This scheme is centered around enhancing the quality of service for computational tasks when confronted with jamming attacks. By utilizing reinforcement learning techniques, the scheme aims to minimize energy consumption and computational latency of mobile devices

by effectively offloading computation to the edge servers. This approach enhances the overall performance and resilience of mobile edge computing systems against jamming attacks.

A method for identifying jamming attack channels in WSNs was proposed by Muhammad Adil et al. [33] using various transmission frequencies and RTT of transmitting signals. By communicating data across the same deployed WSNs, other two edge nodes can check media serviceability in the event that attacker jams one of transmission channels. System can identify the jamming attack channel and conduct the necessary countermeasures to ensure reliable communication within the WSN by comparing the transmission frequencies and RTT of the signals. This technique aims to enhance the robustness and resilience of WSNs against jamming attacks.

Hymlin Rose et al. [36] proposed a timestamp-based method for jamming detection, specifically designed for clustered networks. Two major contributions are presented in this paper. The first contribution is the Node Coverage Range (NCR) technique, which is used for clustering the nodes in the network. This technique helps in organizing the nodes into clusters, which facilitates efficient communication and management within the network. The second contribution is the use of timestamps for jamming detection. In this method, when a node transmits data, it expects to receive an encrypted acknowledgment within a specific timestamp. If the acknowledgment does not reach the receiver within the expected timestamp or if the received signature does not match, it indicates the presence of jamming. In such cases, the transmission is halted, and the data is redirected along an alternative path to avoid the jammed region. It should be noted that the clustering method employed in this approach may introduce some delay as it relies on the Node Coverage Range (NCR). However, the timestamp-based jamming detection mechanism aims to enhance the overall reliability and robustness of the network against jamming attacks.

In many existing techniques, metrics from different aspects are utilized to detect jamming attacks. However, weak links and interference caused by network intrusions, such as crashes and packet failures, can exhibit similar behaviors. Consequently, some of the metrics used for jamming attack detection in previous literature may result in high levels of false alarms. In this research paper, they introduced application of Improved Exponentially Weighted Moving Average (IEWMA) algorithm as an effective method to differentiate amid normal traffic forms and jamming attacks. This is achieved by utilizing the Bytes Arrival Time (BAT) packet as the sole metric. The IEWMA algorithm, a statistical monitoring technique, is employed to estimate the mean of the data and progressively assigns greater significance. Through the utilization of the BAT packet, we successfully identify various types of jamming attacks in WSNs. Importantly, this marks the first instance of utilizing the IEWMA algorithm based on the packet BAT for the detection of diverse forms of jamming attacks in WSNs.

3. Proposed System

Fig. 2 illustrates the overall block diagram depicting the jamming detection method employing the OLSR protocol. The procedure encompasses several stages, including Network establishment, Clustering based on energy levels, Detection of malicious nodes, Configuration of routes using the OLSR algorithm, Transmission of data involving jamming attack, utilization of an enhanced Dijkstra's algorithm, culminating in the successful transmission of data to the intended destination.

3.1 NETWORK SETUP

Networking represents a static framework housing specific functions for data transmission. It serves as a foundational mechanism for fundamental data processing. Typically, the network is structured with pre-defined components including the source node, intermediate nodes, and a sink. The source node is responsible for initiating data transmission throughout the network, primarily for straightforward data exchange. Subsequently, energy-endowed intermediate nodes come into play.

The arrangement of nodes in the environment is assumed to be random, with nodes remaining fixed and neither added nor removed from the network once positioned. This assumption extends to the uniformity of computing and communication capabilities, storage capacity, initial energy levels, and communication range among all network nodes. Communication is feasible only when two nodes come within each other's communication range.

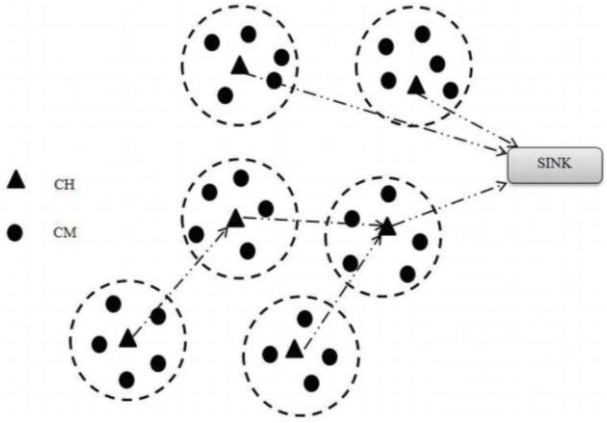

Fig 1 Network model

Under these presumptions, one can envision Wireless Sensor Networks (WSNs) as a graphical representation denoted by G = (N, E), in which N represents the entirety of nodes and E corresponds to the array of edges

interlinking them. Every edge signifies the proximity of two nodes existing within the communication scope of each other. Each individual node retains a roster of adjacent nodes, encompassing their distinct identities, contact particulars, and dependability. This configuration implies the reliability of both source and destination nodes, while any malicious nodes remain disconnected from one another. Additionally, the communication channel remains steadfast and enables two-way communication.

Network Model

The following algorithm illustrates the network model algorithm.

- If there is no current node, then build one as the root
- if the value is less than the current node's data, then insert into the left subtree
- if the value is more than the current node's data, then insert into the right subtree.

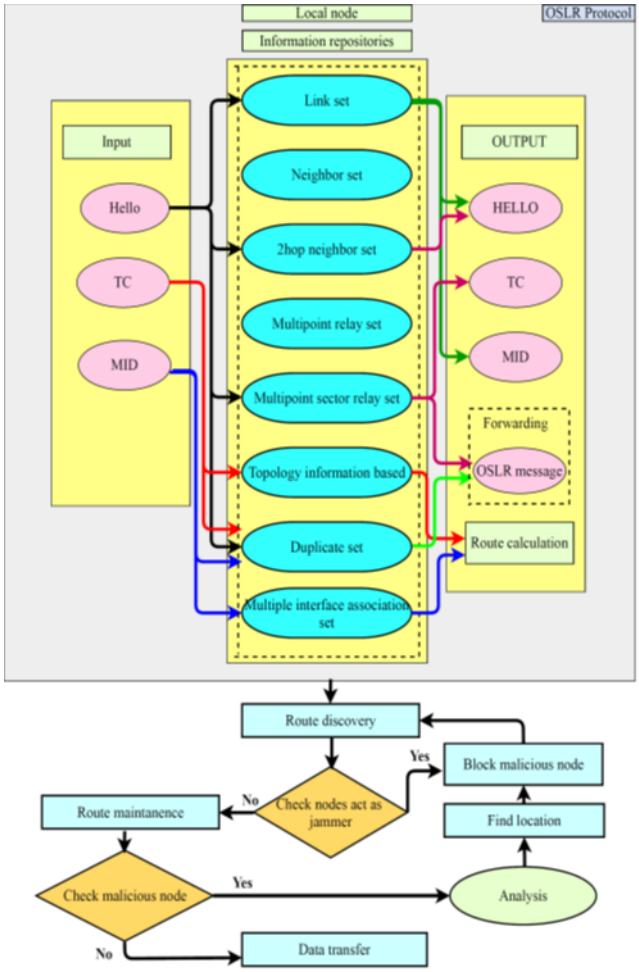

Fig 2 OLSR protocol

When sensor nodes energy in a WSN is totally drained, forcing them to stop operating, the entire sensor network fails. As a result, the primary concern in WSN is energy conservation. Consequently, it is essential to design protocols that utilize minimal energy and are employed for data recognition, processing, and transmission. In a WSN, effectively managing energy consumption emerges as a dynamic responsibility for individual sensor nodes. This encompasses the dual functions of data sensing and transmission, targeting either the nearest node or the central base station. A critical objective of WSNs is to minimize energy usage while extending the network's overall lifespan. Nodes engaged in communication halt their transmissions when the sensor nodes' battery power drops below 24 units.

In order to facilitate effective data transmission, it is essential to establish a routing mechanism that connects nodes within the network. To address this need, the proposed model adopts the Optimized Link State Routing (OLSR) mechanism, as depicted in Fig 2. This approach ensures efficient data transmission across the network and guarantees a streamlined and secure process for transmitting data.

3.2 ENERGY BASED CLUSTERING

The subsequent phase involves the implementation of energy-based routing, which serves to organize nodes into clusters. Clustering entails the act of grouping nodes existing within the network. To accomplish this, a robust clustering algorithm is employed. This algorithm is tasked with the responsibility of effectively grouping all nodes present within the network.

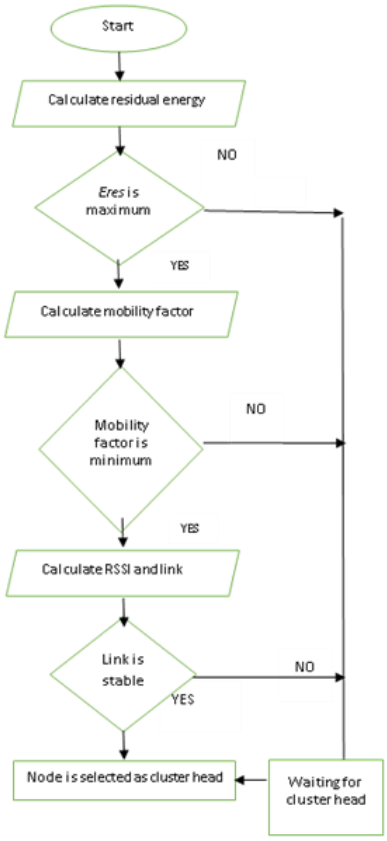

Fig 3 Flowchart for selecting Cluster head

The organization of sensor nodes into groups, known as clusters, is determined based on energy levels of each node. Every individual sensor node is allocated a randomized energy value, and nodes possessing comparable energy levels are subsequently assembled into cohesive groups. This clustering mechanism enables monitoring of neighboring nodes within the cluster. The cluster mechanism incorporates two important characteristics: cluster head and cluster members. Cluster head is selected from nodes with higher energy, while all the other nodes in the group are designated as cluster members. This process is illustrated in the flowchart depicted in Fig 3.

The cluster mechanism demonstrates its reliability when all cluster members dutifully send data to the nearest node. Nonetheless, the situation changes if a malicious node exists within the network, as the cluster head then becomes capable of detecting the cluster member that is behaving improperly. Consequently, the cluster mechanism serves as a reliable method for detecting misbehaving nodes within the network.

The primary benefits of EECH are outlined as follows:

- Novel algorithm for selecting Cluster Heads (CHs) in Wireless Sensor Networks has been developed, aimed at establishing a connected network.

- Thorough simulation experiments have been conducted to evaluate the performance of the EECH algorithm across various metrics.

The remaining energy within node at current moment is referred to as residual energy (Eres). In order for node to be designated as a Cluster Head (CH), it is imperative that it possesses a higher amount of residual energy in comparison to its neighboring nodes. Let Ei stand for initial energy level in sensor node. Equation 1 calculates energy used by node (E(t)) after a time of t.

$$E(t)=(n\_tpkts\times a)+(n\_rpkts\times \beta) \quad (1)$$

Where

ntpkts refers to the numerical value of transmitted packets, while nrpkts represents the numerical value of received data packet. α and β are constants that fall within range of (0,1). Residual energy (Eres) of node at a specific time t is calculated using Equation (2)

$$Eres = Ei - E(t) \quad (2)$$

3.4 ROUTE SETUP USING OLSR

An intentionally designed routing protocol for navigating MANET is OLSR. OLSR runs proactively, which means it continuously distributes to each node in network the topology knowledge of its neighbours and determines the best local transmission channels. OLSR has a lower median end time, making it more efficient. Additionally, OLSR is user-friendly and easy to implement. It is particularly suitable for networks with fast changes in source and destination pairs, making it well-suited for network sensors. OLSR does not require a separate connection for control messages. The adaptability inherent in the OLSR routing protocol permits straightforward incorporation into prevailing operating systems without necessitating alterations to the IP header structure. This protocol mandates solely a connection to the host's routing table. The efficient OLSR routing protocol comprises three essential modules: the system for detecting neighbors and links, the system for multipoint relay, and the mechanisms for link-state computation and path determination. These modules work together to facilitate efficient routing. Below is the pseudocode for the OLSR routing protocol:

1. Generate a mobile node.

2. Define the node as the sender.

3. Specify the node as the destination.

4. Configure the routing protocol as OLSR.

5. Commence the simulation timer.

6. Initialize the communication range by setting the radio range.

7. Set node "s" as the intrusion prevention node, which senses all neighboring nodes and captures their behavior.

If node "n" has updated routing packets or does not forward data to the destination:

Create a table to store all misrouted nodes, called "all misroute node n".

Send a reply packet to the source node informing it about the misbehavior of node "n".

Block the misbehaving node "n" at node "s".

Invoke the "recompute_path()" function.

Else, establish a secure path.

8. The "recompute_path()" function is defined with parameters (sender, destination, route-pkt).

If node "m" is either within the radio range or considered a neighboring node, or if node "n" is marked as false:

- Generate a route table.

- Accept the route packet at the destination.

Alternatively, if the node is beyond the communication range or the destination is unreachable:

- Address the scenario as appropriate.

9. Dispatch an acknowledgment to the sender node.

10. The sender node transmits a data packet through the established secure path.

11. Conclude the session.

This pseudocode outlines the basic operations performed by the OLSR routing protocol to maintain an efficient and updated routing table within the network.

3.5 MDSP

Dijkstra's technique is used to find the shortest paths between graph nodes. The shortest route is always calculated by this approach, whether the edge weights are positive or negative. The original Dijkstra's method found the shortest route between any two given nodes; today's more popular implementation uses a single node as the "source" and finds the shortest path from that node to every other node in the graph.

The time complexity of this algorithm can be analyzed as follows: The initial for loop runs in O(N) time. During each iteration of the while loop, extracting the minimum value from the heap takes logN time. The validity of this algorithm has been extensively established. Nevertheless, with an increasing number of nodes in a graph, the runtime of the algorithm proportionally extends. The application of this algorithm primarily revolves around finding the path between source and destination vertices (or nodes) within a graph while factoring in the decreasing weights of its under lying edges.

In the context of a graph G, composed of sets N (nodes) and E (edges), the algorithm follows these steps:

1. Initialize all node distances as infinity, except for the source node which is set to 0.

2. Insert the source node into a min-priority queue using (distance, node) format, where priority is based on node distances.

3. Extract the node with the minimum distance from the priority queue (initially, this will be the source node).

4. Update the distances between connected nodes: If the distance plus edge weight of the current node is less than the distance of the following node, update the latter and move the node with the new distance[5]. If the popped node has been visited before, skip it.

6. Repeat the algorithm until the priority queue becomes empty, exploring the graph for shortest paths.

In cases where jamming is detected, an alternative node is selected to carry out the data transmission. To determine the shortest path for this data transmission, a Modified Dijkstra's Shortest Path Algorithm (MDSP) is employed.

One of the key benefits of MDSP is its ability to provide reasonably accurate location estimation, even when based on limited distance information. This is achieved by calculating the distance between two nodes 'i' and 'j' using the formula:

$$dij(x)=\sum_{a=1}^{m} \left[\left[(tia-tja)\right]^2\right]^{(1/2)} \quad (3)$$

Where tia is the location of node ‘i’ and tja is the location of node ‘j’.This equation calculates the distance between the two nodes based on their respective coordinates in a multi-dimensional space. This approach allows for accurate determination of distances even with restricted information, contributing to effective location estimation in the presence of jamming.

1. Initialize:
- Start the loop to iterate through each node 'n' in the Graph.
- Set 'alternate_path[i]' for each node 'n' to NULL initially.
- Set 'dist[n]' for each node 'n' to infinity initially.
- Call the 'weight_update' function with the parameter 'choice' to update weights.
2. Weight Update:
- Inside the 'weight_update' function, the weights of edges or paths are updated based on the specified 'choice'. The exact details of this function aren't provided, but it appears to handle weight updates based on certain criteria.
3. Update Distances:
- Iterate through each node 'n' in the Graph.
- If the current node 'n' is either the source or the destination:
- Start another loop to iterate through each neighbor 'u' of node 'n'.
- Check if 'alternate_path[i]' (initially set to infinity) is greater than the sum of 'dist[u]' (current distance value of neighbor 'u') and the distance between 'u' and 'n'.
- If the above condition holds true, update 'alternate_path[i]' with 'dist[u]' + distance(u, n).

Algorithm: MDSP

The success of the MDSP algorithm is estimated by determining the shortest path through an estimation of node complexity and time duration.

3.6 Detection of Jamming Using IEWMA

The Improved Exponentially Weighted Moving Average (IEWMA) method proves effective for detecting minor shifts within time-series data. It stands out as an efficient statistical technique. IEWMA boasts distinct advantages over other Statistical Process Control (SPC) techniques, primarily due to its adept utilization of current and historical data. This enables it to swiftly and accurately sense slight shifts in time-series data.

The core mechanism of IEWMA begins with setting a threshold, which governs acceptable behavior. It consistently updates the average based on the identified data traffic, combining both historical and real-time observations. This involves a weighting constant denoted as gamma (γ). The gamma value determines the balance between the significance of recent and past observations, thereby influencing its sensitivity to gradual or minor shifts. The IEWMA can be mathematically computed using the formula:

$$r(t)= \gamma.s(t)+(1-\gamma).r(t-1) \quad t=1,2,3..n \tag{4}$$

Here,

r(t) represents the mean of historical data,

s(t) is the observation at time t,

n denotes the number of iterations being considered, including r(0),

γ is the smoothing constant with the constraint ($0 < \gamma <= 1$).

The choice of γ significantly influences the effectiveness of IEWMA. A higher γ value (close to 1) assigns more weight to recent observations, diminishing the weight of older ones, and vice versa. A value around 0.5 is often chosen for γ. Nonetheless, a very small γ might render the technique insensitive to attacks with rational intensity or minor scope. Generally, a γ value between 0.2 and 0.5 is recommended for balanced sensitivity.

The predictable variance of the IEWMA statistic can be calculated using:

$$\sigma z^2 = \sigma x^2 * (\gamma / (2 - \gamma)) \tag{5}$$

Here, σx is the standard deviation derived from historical data.

The control chart provides the centerline, which is commonly known as the target value. The Upper Control Limit (UCL) and Lower Control Limits (LCL) can be determined through the following equation

$$UCLz = z_0 + f.\sigma z \tag{6}$$

$$LCLz = z_0 - f.\sigma z \tag{7}$$

Where the factor f is set to be equal to 3-sigma control limits.

While detecting a jamming attack, the main emphasis is placed on analyzing situations where the upper threshold is exceeded. In some cases, observations reveal that traffic drops below the lower control boundary, which may be due to network anomalies. The proposed jamming detection technique comprises two parts. The first part is the training phase, where outdated Beacon Announcement Traffic (BAT) is collected from suitable member nodes within the cluster to establish a baseline profile. No alteration-based detection methods are performed during this phase.

In the second part, known as the test phase, configuration changes are detected on a per-packet basis using the Incremental Exponentially Weighted Moving Average (IEWMA) algorithm, as depicted in Fig 4. Once a jamming node is found during a jamming attack, an alarm is set off and the malicious node is unplugged from the network. Furthermore, the associated data packet is refused, resulting in packet loss, if the total broadcast length (end-start) exceeds a predefined threshold (20 ms). This mechanism is particularly tailored to address instances of elevated malicious interference, wherein a substantial portion or even the entirety of lost packets can be attributed to deliberate disruptive actions.

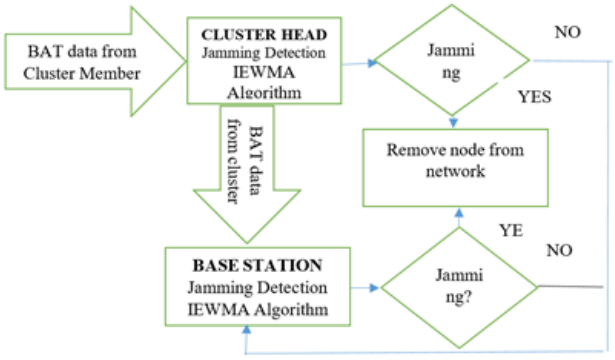

Fig 4 Jamming Detection

4. Results and Discussion

4.1 Simulation setup

The effectiveness of the proposed approach is evaluated through the utilization of Network Simulator-2 (NS-2), by integrating the simulation setups detailed in Table 1. The efficiency of the Incremental Exponentially Weighted Moving Average (IEWMA) technique is assessed within a simplified network environment, where jamming nodes are randomly positioned, impacting network traffic with varying data rates ranging from 200 Kbps to 1000 Kbps.

The network model comprises 50 nodes, including a source node, a sink node, a malicious node, and a grouping based on energy-based clustering with cluster heads and cluster members, as illustrated in Fig. The simulation involves a network area of 250m × 250m accommodating the deployment of 50 sensor nodes. All sensor nodes possess an identical communication range (X = 50m) and storage capacity.

Furthermore, a mobile destination node is introduced, adhering to a uniform traffic pattern, as it navigates the network in a randomized fashion to collect data from sensor nodes. Moreover, the effectiveness of the suggested IEWMA algorithm is evaluated by contrasting it with a timestamp-based approach. This comparison serves to gauge the algorithm's efficiency. Overall, the assessment involves a detailed evaluation of the proposed scheme using NS-2, considering various network configurations and scenarios to gauge its practicality and effectiveness.

Table I: Simulation parameter configuration

Type of parameter	Value
Communication Range	250m
No. of nodes	50
Routing Protocol	OLSR
Channel	Wireless
Packet Size	64
Initial Time Value	0.05
BAT	38
Simulation Area	500×500
Communication Range	250m
Simulation Time	5 secs
Propagation model	Two ray model

After the formation of the network, each node is assigned its own energy level. The nodes are then grouped based on their energy levels, and the node with the highest energy within each group is designated as the cluster head. Additionally, each node possesses its own routing table, which includes information such as its node ID, cluster number, neighboring nodes, and the number of hops required to reach other nodes. Moreover, each node broadcasts its node ID to all other nodes in the network, facilitating communication and information exchange. In Fig 5, Node 0 is designated as the data center or source node, Node 36 functions as the sink node, and Node 37 assumes the role of the packet attacker or jammer node.

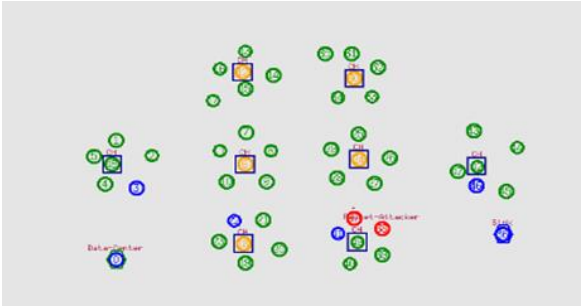

Fig 5 Network model

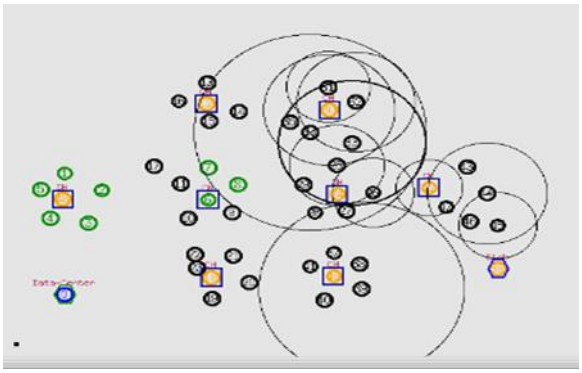

Fig 6 Transmission of Node ID

Once the jamming node has been located, the source node establishes a number of paths under the condition that nodes can reliably communicate within a predetermined range. To accomplish this, an adapted Dijkstra's algorithm is employed. This algorithm selects the most suitable route for transmitting data to the destination.

Calculation of IEWMA

The observed time series data will vary based on the standards that come before or after in the data series. In real-world scenarios, this is analyzed using the time series data r(t), where t = 1, 2, 3... n, as depicted in Table II.

$r(1)=0.9s(1)+0.1r(0)$

$r(2)=0.9s(2)+0.1r(1)$

. .

. .

. .

$r(n)=0.9s(n)+0.1r(n-1)$

Time series data typically exhibit an increasing rate and are not mutually auto correlated, which suggests that the estimated values do not experience abrupt changes. This gradual increase enables the establishment of upper and lower control limits. In the event of a jamming attack, traditional Statistical Process Control (SPC) techniques typically reset the intensity of the value. Nonetheless, in this methodology, the need for such resetting is eliminated as the parameters' influence is automatically regulated within the predefined upper and lower control thresholds.

Table II: Calculation of r(t) and γ

Sl.no	**r(t)**	**γ**
1	8	0.72
2	9	0.72
3	10	0.72
4	11	0.72
5	12	0.71

6	13	0.71
7	14	0.72
8	15	0.72
9	16	0.72
10	17	0.72
11	18	0.72
12	18.5	0.73
13	19	0.73
14	19.5	0.73
15	20	0.73
16	20.5	0.73
17	21	0.74
18	21.5	0.74
19	22	0.74
20	22.5	0.75

Table III : Calculation of standard deviation

Sample	B	Sample	B
1	32	11	29.6
2	27	12	28.1
3	33	13	29.9
4	29.3	14	31.3
5	30.1	15	30.1
6	27	16	31.2
7	31	17	32.6
8	30.1	18	33.3
9	31.2	19	34.8
10	30.5	20	29.9

Based on the average moving series, the standard deviation value can be evaluated and is presented in Table III. When considering an average moving series of 2, the target standard deviation value is the average range divided by 1.128. Within the provided criteria, the prevailing average standard deviation value is 1.95. As demonstrated in Fig 7, the control chart limits go beyond the upper control limit (UCL) in the presence of jamming.

z

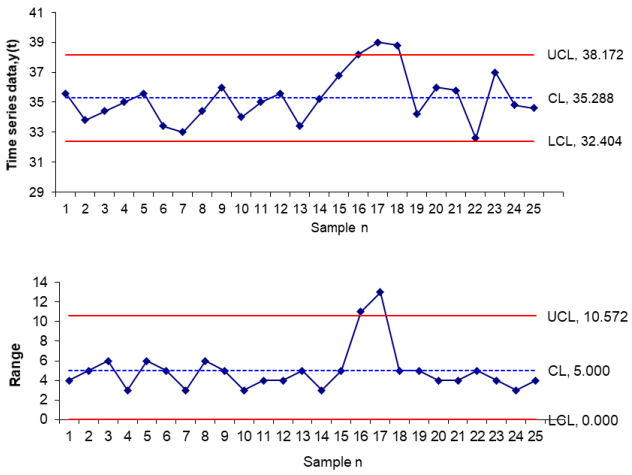

Fig7 Control chart based on IEWMA (before rerouting)

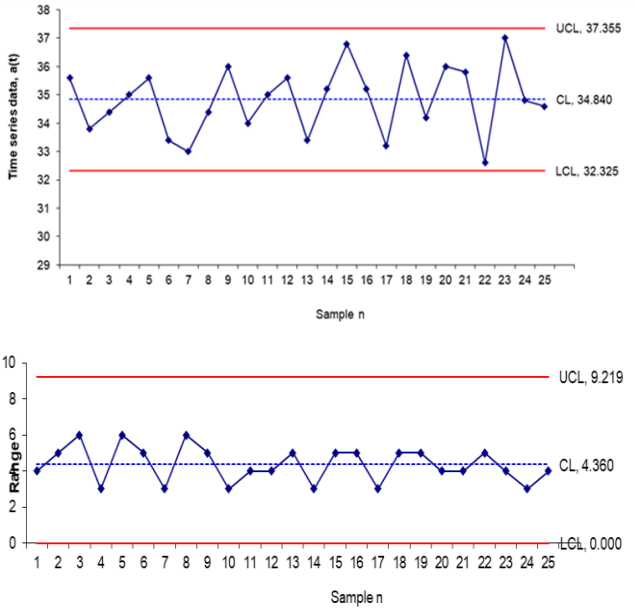

Fig 8 : Control chart based on IEWMA (After rerouting)

Upon detecting jamming, the data transmission seamlessly shifts to an alternative route using the OLSR protocol. Subsequently, upon examination, the control limits will fall within the upper control limit (UCL) and lower control limit (LCL), as depicted in Fig 8.

4.2 Simulation results

In this section, we provide an evaluation of the performance of the novel IEWMA technique. Our emphasis is on evaluating metrics like Packet Loss Rate, Energy Consumption, and Network Throughput. We conduct a comparative study against the timestamp technique to assess the effectiveness of our proposed approach.

4.2.1 Analysis of Energy consumption

Illustrating the suggested approach entails the consideration of an initial broadcast energy amounting to 0.2J. The scientific model is structured around network throughput, a critical metric where Γ is greater than zero. The computation of energy consumption for nodes within the Wireless Sensor Network (WSN) is provided as follows:

$$Ec = \sum_{s=o}^{t} S \times n \times (1 + R(s).P) \times \frac{R(s)}{(m+A)\times T} \quad (8)$$

Where, P is the probability function $a + \Delta, \Delta$ is the random number, a is the energy enhancement of nodes, $R(s) = R_o \times (1 + a)$, R_o is the initial energy, m is the number of nodes, S is the transmitted power.

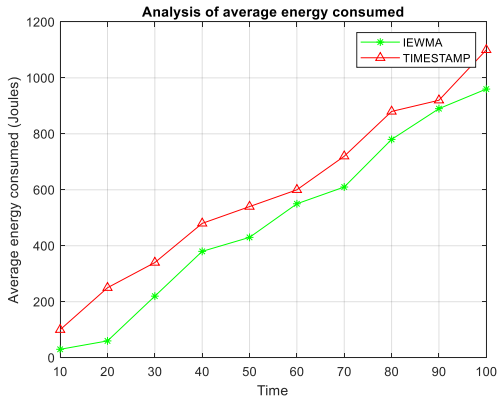

Fig 9: Analysis of Average Energy Consumption

The results indicate that the proposed IEWMA technique outperforms the timestamp technique in terms of the aforementioned metrics.

Fig 9 presents the average energy consumption in a network with varying numbers of malicious nodes, highlighting that the IEWMA technique outperforms the timestamp approach. The key factor driving energy savings in IEWMA is its effective jamming detection mechanism. By identifying the jamming node, the technique enables all active nodes in the Wireless Sensor Network (WSN) to cease sending data to the receiver, thereby reducing energy consumption

4.2.2 Analysis of Packet Delivery Ratio

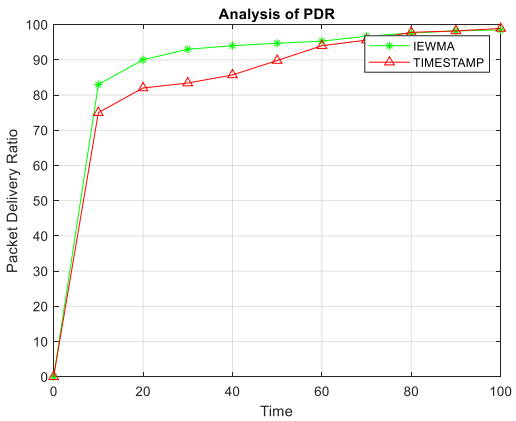

Fig 10 Analysis of Packet Delivery Ratio

The packet delivery ratio can be defined as the ratio of data packets successfully received at their intended destinations to the total number of data packets generated. The progression of the packet delivery ratio over time can be expressed as follows:

$$PDR, \delta_t = \sum_{s=0}^{t} 1 - \left(1 - \frac{m}{|n|}\right)^s \quad (9)$$

In Fig.10, the PDR pattern at various times is depicted, considering a network with m nodes and an interval of time s. The figure clearly illustrates a decreasing trend in the packet loss rate as time progresses. The analysis underscores the effectiveness of the proposed IEWMA algorithm in efficiently identifying the optimal shortest path. This approach demands fewer nodes and less time for computation compared to established algorithms. As a result, the algorithm contributes to increased PDR within a shorter time frame. However, it is the value noting that as the duration of time increases, the improvement in PDR becomes less pronounced.

4.2.3 Analysis of Network Throughput

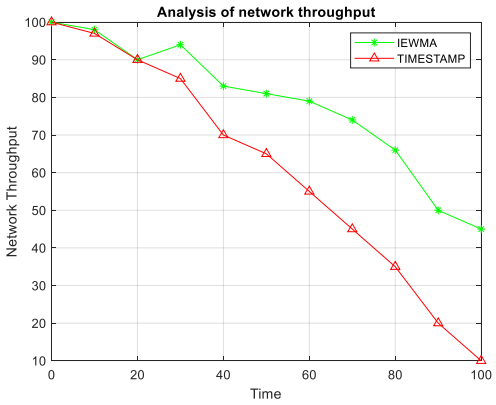

Fig 11 Network Throughput analysis

Network throughput, which represents the speed at which packets or bits efficiently travel through a network channel, can be expressed mathematically as follows: In certain scenarios, network throughput could drop to zero due to a jamming attack. Let "tr" symbolize the transmission range, and "s" stand for time in seconds. The mathematical formulation for network throughput can be elaborated as follows:

$$T = \sum_{s=o}^{t} \delta_t (1 - \frac{tr}{s})/n \quad (10)$$

Fig. 11 depicts the network throughput analysis, demonstrating the noteworthy achievement of elevated network throughput by the proposed algorithm. This accomplishment can be attributed to the absence of any need for supplementary information to execute dynamic updates, a contrast to prevalent methodologies. The IEWMA scheme integrates an adept route damage

reporting approach, effectively countering susceptibilities posed by jamming attacks. Consequently, the network throughput experiences a substantial enhancement.

4.2.4 Analysis of Node Lifetime

The lifespan of a node, denoted as Lnx, at a given time t, Lnxt, is represented as the quotient of the remaining energy (E_t) divided by the initial energy capacity of the node (E_init). This measurement is presented as a percentage value.

$$L^{n_{xt}}, = \frac{E_t}{E_{init}} *100\% \qquad (11)$$

In Fig 12, it is observed that the lifetime of nodes in the IEWMA technique is longer compared to the lifetime of nodes in the timestamp technique. This longer node lifetime contributes to enhanced stability among the nodes within the network. Consequently, there is a reduced need for frequent updates of the routing table, resulting in lower overhead. It is crucial for the node's lifetime to remain stable, ensuring that it does not become depleted of energy. If a node is drained and loses its energy, it becomes hidden and can cause significant interference on the transmission within the network. Therefore, the prolonged node lifetime in the IEWMA technique improves stability and minimizes transmission disruptions caused by drained nodes.

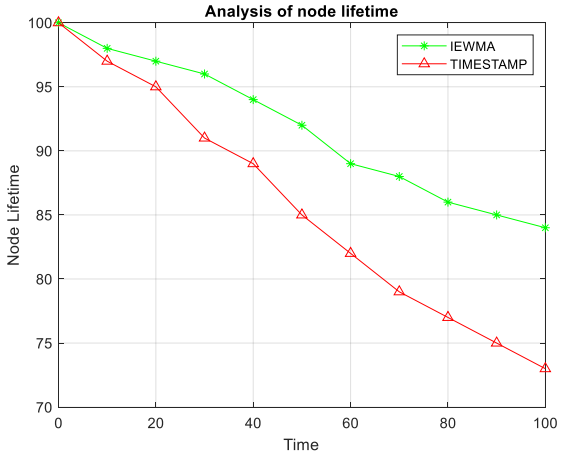

Fig 12 Node lifetime Vs Time

4.2.5 Analysis of Network Lifetime

This segment centers on the assessment of the proposed IEWMA algorithm's performance concerning the duration of the network's lifespan. It assesses the network's resilience by analyzing the occurrence of network breakages and the stability of the links established between the nodes. The evaluation aims to provide insights into the algorithm's effectiveness in maintaining a reliable and long-lasting network infrastructure.

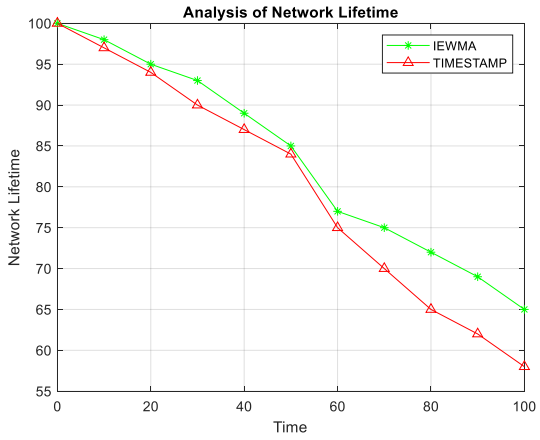

Fig 13 Analysis of Network lifetime

The network's lifespan is determined by computing the cumulative lifespan of each individual node, and it is formulated as follows:

$$L^{Ne_t} = \frac{\sum_{i=1}^{n} L_{n_{xt}}}{E_{init}} *100\% \qquad (12)$$

Within the framework of the IEWMA technique, the node's longevity is derived from the cumulative lifespans of all nodes within the network. This approach yields extended lifetimes in comparison to the timestamp system, as depicted in Fig 13. In contrast, the current scheme results in heightened network traffic, consequently leading to an elevated risk of network failure. However, the proposed method establishes links with a uniform time interval, ensuring the network's endurance over a prolonged span.

5. Conclusion

This paper presents an IEWMA scheme aimed at mitigating the impacts of Jamming attacks by identifying and mitigating malicious nodes engaged in data transmission. The MDSP technique goes a step further in enhancing this scheme by assisting the network in identifying the optimal route to the sink node. The suggested methodology uses a rigorous process to identify different types of jamming attacks. The jamming detection mechanism is used by both the cluster head and the base stations to detect attacks on the member nodes. The IEWMA method is used in conjunction with the packet BAT metric to identify anomalies in packet sequences caused by jamming assaults. Through simulations, the proposed method's effectiveness is evident, showcasing adept jamming attack detection with minimal overhead in Wireless Sensor Networks (WSNs). This approach results in reduced packet loss rates, decreased energy consumption, and prolonged node and network lifetimes.

References

[1] Misra, S.; Singh, R.; Mohan, S.V. Information warfare-worthy jamming attack detection mechanism for wireless sensor networks using a fuzzy inference system. Sensors 2010, 10, 3444–3479.

[2] Strasser, M.; Danev, B.; ˇCapkun, S. Detection of reactive jamming in sensor networks. ACM Trans. Sens. Netw. (TOSN) 2010, 7, 16.

[3] Spuhler, M.; Giustiniano, D.; Lenders, V.; Wilhelm, M.; Schmitt, J.B. Detection of reactive jamming in DSSS-based wireless communications. IEEE Trans. Wirel. Commun. 2014, 13, 1593–1603.

[4] Guan,Y.;Ge,X.DistributedSecureEstimationoverWirelessSensorNetworksAgainstRandomMultichannel Jamming Attacks. IEEE Access 2017, 5, 10858–10870.

[5] Cordero, C.V.; Lisser, A. Jamming Attacks Reliable Prevention in a Clustered Wireless Sensor Network. Wirel. Pers. Commun. 2015, 85, 925–936.

[6] Mpitziopoulos, A.; Gavalas, D. An effective defensive node against jamming attacks in sensor networks. Secur. Commun. Netw. 2009, 2, 145–163.

[7] Alnifie, G.; Simon, R. A multi-channel defense against jamming attacks in wireless sensor networks. In Proceedings of the 3rd ACM Workshop on QoS and Security for Wireless and Mobile Networks, Chania, Crete Island, Greece, 22 October 2007; pp. 95–104.

[8] Del-Valle-Soto, C.; Mex-Perera, C.; Monroy, R.; Nolazco-Flores, J.A. MPH-M, AODV-M and DSR-M Performance Evaluation under Jamming Attacks. Sensors 2017, 17, 1573.

[9] Çakirog˘lu, M.; Özcerit, A.T. Design and evaluation of a query-based jamming detection algorithm for wireless sensor networks. Turk. J. Electr. Eng. Comput. Sci. 2011, 19, 1–19.

[10] Bhavathankar, P.; Mondal, A.; Misra, S. Topology control in the presence of jammers for wireless sensor networks. Int. J. Commun. Syst. 2017.

[11] A. Azim et al., "Efficient Jammed Area Mapping in Wireless Sensor Networks," IEEE Embedded Sys. Lett., 2014, vol. 6, no. 4, pp. 93–96.

[12] P. Ganeshkumar, K. P. Vijayakumar, M. Anandaraj (2016)., "A novel jammer detection framework for cluster-based wireless sensor networks" EURASIP Journal on Wireless Communications and Networking, pp.1-25 , https://doi.org/10.1186/s13638-016-0528-1

[13] Sang Quang Nguyen, Hyung Yun Kong (2015), "Combining Binary Jamming and Network Coding to Improve Outage Performance in Two-Way Relaying Networks under Physical Layer Security", International Journal on Wireless Pers Communication, Volume 85, Issue 4, pp 2431–2446.

[14] Michael Riecker, Sebastian Biedermann, Rachid El Bansarkhani, Matthias Hollick (2014), "Lightweight energy consumption-based intrusion detection system for wireless sensor networks" International Journal of Information Security, Volume 14, Issue 2, pp 155–167.

[15] Domenico Giustiniano, Vincent Lenders, Jens B.Schmitt (2013), "Detection of Reactive Jamming in DSSS-based Wireless Networks", International conference on Security and privacy in wireless and mobile networks, Pp 43-48.

[16] Michael Spuhler, Domenico Giustiniano, Vincent Lenders(2014), "Detection of Reactive Jamming in DSSS-based Wireless Communications", IEEE Transactions On Wireless Communications, Vol. 13, No. 3,pp.1593-1604.

[17] Yuzhe Li, Ling Shi, Peng Cheng, Jiming Chen, and Daniel E. Quevedo,(2015)" Jamming Attacks on Remote State Estimation in Cyber-Physical Systems: A Game-Theoretic Approach" , IEEE Transactions On Automatic Control, Vol. 60, No. 10, pp. 2831-2836.

[18] Haijun Zhang,Hong Xing,Julian Cheng (2016) , "Secure Resource Allocation for OFDMA Two-Way Relay Wireless Sensor Networks Without and With Cooperative Jamming" IEEE Transactions on Industrial Informatics, Volume: 12, Issue: 5, , PP. 1714 – 1725.

[19] Xingkun Xu ,Kunlun Gao, Xiaokun Zheng ,Ting Zhao (2012), "A zero-sum game theoretic framework for jamming detection and avoidance in Wireless Sensor Networks" International Conference on Computer Science and Information Processing (CSIP) pp.265-270.

[20] Donggang Liu , Joshua Raymer ,Andy Fox (2013), "Efficient and timely jamming detection in wireless sensor networks", IEEE 9th International Conference on Mobile Adhoc and Sensor Systems (MASS), pp.335-343

[21] Ahmad YusriDak , Noor Elaiza Abdul Khalid , SaadiahYahya (2013) "A novel framework for jamming detection and classification in wireless networks" International Conference on Computing and Networking Technology (ICCNT),pp.240-246.

[22] Ying Xuan, Yilin Shen, Nam P. Nguyen , My T. Thai (2012) "A Trigger Identification Service for

Defending Reactive Jammers in WSN" IEEE Transactions on Mobile Computing, Volume: 11, Issue: 5, pp. 793 - 806 .

[23] Aasha Nandhini, Kishore Rajendiran, Radha Sankararajan (2014) "A Novel Frequency Hopping Spread Spectrum Technique using Random Pattern Table for WSN" , International Journal of Ad Hoc & Sensor Wireless Networks,pp. 255-275.

[24] Xiaofeng Liu, Liusheng Huang, Hongli Xu, Wenbo Shi, Dashan Wang (2011)"An Energy-Efficient k-connected Scheme for Wireless Sensor Networks" International Journal of Ad Hoc & Sensor Wireless Networks ,pp 1-21.

[25] Amirthasaravanan Arivunambi and Arjun Paramarthalingam(2022)," Intelligent slime mold algorithm for proficient jamming attack detection in wireless sensor network, Global Transitions Proceedings(3)pp.386-391.

[26] P Bhavathankar, S Chatterjee, S Misra, "Link-quality aware path selection in the presence of proactive jamming in fallible wireless sensor networks", IEEE Trans. Commun. 66 (4) (2018) 1689–1704 .

[27] Nashab Alikh and Amir Rajabzadeh," Using a lightweight security mechanism to detect and localize jamming attack in wireless sensor networks", Optik(2022),pp.1-14.

[28] L. Pang, X. Chen, Z. Xue, R. Khatoun, A novel range-free jammer localization solution in wireless network by using PSO Algorithm, Commun. Comput. Inf. Sci. vol. 728 (2017) 198–211

[29] G. Pan, et al., On secrecy performance of MISO SWIPT systems with TAS and imperfect CSI, IEEE Trans. Commun. vol. 64 (9) (2016) 3831–3843.

[30] I. Sudha , Mohammed Ahmed Mustafa , R. Suguna ,Sathishkumar Karupusamy , Veeraswamy Ammisetty , Shavkatov Navruzbek Shavkatovich , M. Ramalingam , Pratik Kanani ," Pulse jamming attack detection using swarm intelligence in wireless sensor networks", Optik,272(2023),pp.1-13.

[31] Feriel Cherifi , Mawloud Omar , Tinhinane Chenache , Sylia Radji ," Efficient and lightweight protocol for anti-jamming communications in wireless body area networks", Computers and Electrical Engineering(2022),pp.1-11.

[32] Xiao L, Lu X, Xu T, Wan X, Ji W, Zhang Y. Reinforcement learning-based mobile offloading for edge computing against jamming and interference. IEEE Trans Commun 2020;68(10).

[33] Muhammad Adil , Mohammed Amin Almaiah , Alhuseen Omar Alsayed and Omar Almomani ," An Anonymous Channel Categorization Scheme of Edge Nodes to Detect Jamming Attacks in Wireless Sensor Networks",Sensors(2020), 20, 2311,pp.1-19.

[34] Cortés-Leal, A.; Del-Valle-Soto, C.; Cardenas, C.; Valdivia, L.J.; Del Puerto-Flores, J.A. Performance Metric Analysis for a Jamming Detection Mechanism under Collaborative and Cooperative Schemes in Industrial Wireless Sensor Networks. Sensors 2022, 22, 178. https://doi.org/10.3390/s22010178

[35] Jennifer S. Raj, Joy Iong-Zong Chen, Ivan Kotuliak, Khaled Kamel, Caps Net-based computing in cognitive communications, International Journal of Communication Systems, 10.1002/dac.5066, 35, 2, (2021).

[36] Hymlin Rose S G, Jayasree T," Detection of jamming at-tack using the timestamp for WSN" Journal of Adhoc Net-works" Elsevier,2019.

[37] Dhanwanth, B. ., Saravanakumar, R. ., Tamilselvi, T. ., & Revathi, K. . (2023). A Smart Remote Monitoring System for Prenatal Care in Rural Areas. International Journal on Recent and Innovation Trends in Computing and Communication, 11(3), 30–36. https://doi.org/10.17762/ijritcc.v11i3.6196

[38] Juan Lopez, Machine Learning-based Recommender Systems for E-commerce , Machine Learning Applications Conference Proceedings, Vol 2 2022.

[39] Sherje, N. P., Agrawal, S. A., Umbarkar, A. M., Dharme, A. M., & Dhabliya, D. (2021). Experimental evaluation of mechatronics based cushioning performance in hydraulic cylinder. Materials Today: Proceedings, doi:10.1016/j.matpr.2020.12.1021