



FDDDA: False data detection and data aggregation protocol in wireless sensor networks

 **Mukesh Kumar**¹

¹School of Computer Science and Engineering, Galgotias University, Greater Noida, India.

 **Monika Sethi**²

²Email: mukeshk.charvla@gmail.com

 **Chander Prabha**³

³Chitkara University Institute of Engineering and Technology, Chitkara University, Rajpura, Punjab, India.

 **Shilpa Hudnurkar**⁴⁺

⁴Email: sethi.monika4@gmail.com

 **Durgesh Nandan**⁵

⁵Email: prabhanice@gmail.com

⁴Symbiosis Institute of Technology, Pune Campus, Symbiosis International (Deemed University) (SIU), Pune 412115, India.

Email: shilpa.hudnurkar@sitpune.edu.in

⁵School of CS and AI, SR University, Warangal, Telangana, India.

Email: durgeshnandano51@gmail.com



(+ Corresponding author)

ABSTRACT

Article History

Received: 21 January 2026

Revised: 12 May 2026

Accepted: 3 June 2026

Published: 29 September 2026

Keywords

ANN

Cluster head authentication

Data aggregation

False data detection attack

MAC verification

SHA-based integrity

Wireless sensor network.

This research paper investigates false data injection attacks, which represent a critical threat to wireless sensor networks because their operational limitations and security weaknesses reduce network reliability, particularly in hierarchical data aggregation systems. The limitations of WSNs make them vulnerable to multiple attacks, which include wormhole attacks, Sybil attacks, and malicious data manipulation and fabricated reports that distort aggregated data accuracy. The authors used systematic analysis methods to investigate security vulnerabilities in hierarchical aggregation systems. They studied methods that can identify real data changes through aggregation while detecting unauthorised data changes via injection attacks. The system uses false data detection methods to secure data aggregation processes, which protect the entire system from data breaches that might compromise data integrity. The results show that attackers who control data aggregation systems can manipulate aggregated data, which results in base stations and end users receiving false information that jeopardizes network safety and disrupts their ability to make choices. The research proves that organizations need to use security and aggregation solutions that operate together to protect their systems from current threats. It highlights the necessity of implementing an integrated security framework that combines robust detection and aggregation strategies to enhance reliability, improve resilience against adversarial attacks, and ensure trustworthy data dissemination in resource-constrained WSN environments.

Contribution/Originality: The paper's primary contribution is to identify studies that examine secure methods to distinguish between genuine data aggregation and malicious data manipulation. The paper presents its first complete logical examination of combined security systems. The research demonstrates improved protection against hostile attacks targeting resource-limited wireless sensor networks.

1. INTRODUCTION

Wireless Sensor Networks (WSNs) function as fundamental technology that enables environmental monitoring together with industrial system tracking, smart city development, healthcare monitoring, and battlefield surveillance operations. Real-world examples from military surveillance systems, disaster monitoring platforms, and smart grid systems show how compromised sensor data results in major operational and financial damage [1, 2]. Sensor nodes face extreme risk from false data injection, clone attacks, replay attacks, and packet forgery because their processing

power, memory, bandwidth, and battery capacity are insufficient [3]. Most existing approaches to secure data aggregation solutions lack complete addressing of false data threats because they either concentrate on authentication or aggregation performance without combining false data detection in their hierarchical framework.

Existing methods fail to recognize valid aggregated changes because they treat all data alterations as suspicious, which creates an important research need for developing secure aggregation solutions that use minimal energy. The paper addresses its primary research question through the investigation of secure, energy-efficient hierarchical data aggregation methods that enable accurate detection of false data injection in resource-limited wireless sensor networks.

The paper proposes the False Data Detection and Data Aggregation (FDDDA) protocol, which combines modified SHA-based Message Authentication Codes (MAC) with neural network-based anomaly detection at the aggregator level to address this research gap. The study provides the following main contributions.

- i) In the proposed FDDDA protocol, the number of message transfers from the sensor node to the aggregator node and from the aggregator node to the base is less than that of the existing technique.
- ii) Secondly, false or bad data is detected at the cluster head or aggregator node. Thus, if false data is detected at the cluster head, unnecessary data transfers can be avoided, which in turn saves energy as well as the extra overhead caused by false aggregation at the base station.
- iii) The performance of the proposed protocol is compared with that of the existing IPHCDA. The total transmitted data in FDDDA is compared to that of the traditional data authentication scheme, as the number of data aggregators and the ratio of false data to legitimate data vary. Because the amount of overall data using data aggregation and false data detection is reduced in the proposed scheme, the amount of data transmitted in FDDDA is less than that of the traditional data authentication scheme.
- iv) FDDDA consumes less energy as data aggregation in FDDDA significantly reduces the data transmission in the network as compared to IPHCDA; the number of message transfers in the case of the proposed scheme is less, and the transmission of a bit can consume as much energy as the execution of 900 instructions.
- v) False Data Injection attack: A combined data aggregation and false data detection technique is proposed in this work to ensure that data are altered by data aggregation only. In this scheme, before performing the Data Aggregation, on each cluster head, the node authentication and the false data detection will be performed. It means the aggregation will be performed only on true data, so that the reliability of the network will be increased.
- vi) Clone attack prevention: In our proposed protocol with the use of the MAC-based mSHA algorithm, node replication or clone attack can be avoided by the presence of a proper authentication process.
- vii) Replay attacks: In a replay attack, packets are transmitted again by the attacker to achieve a malicious impact. However, the FDDDA protocol prevents this attack by time-stamping all data packets.
- viii) Forgery of aggregated Data packets: FDDDA employs MAC-based modified SHA to prevent attacks targeting data integrity. Therefore, an attacker can successfully change data if and only if he/she can forge a valid MAC for the text. As the proposed protocol uses an unforgeable MAC protocol, such as mSHA, it is very difficult for attackers to forge aggregated data packets.
- ix) Unauthorized aggregation: In the proposed data aggregation protocol, the Hash key must be known by the attacker to achieve unauthorized aggregation. Hence, in FDDDA, an attacker must capture at least one cluster head or data aggregator to access the Keys. The attacker cannot achieve unauthorized aggregation by compromising a normal sensor node because there is only one data aggregator in each cluster.

The proposed effort can be divided into four primary phases: data aggregation, false data detection, source authentication, and network setup. A distinct algorithm is defined in the following sections for every step.

2. RELATED WORK

In literature, several research works have been proposed in the context of using machine learning for error detection [4–8]. In this study, we focus on existing research on artificial neural networks (ANN) and how they are employed in WSN. It is evident that a neural network-based approach is now employed extensively in WSN for error detection. Neural networks that have been trained to detect fake data have been presented for integration into sensors [9]. Data obtained from neighbors can be utilized for training the neural network. Subsequently, it assesses the error between the two values after comparing the neural network and the sensor node's data. The sensor node's neural network was built because computation consumes a lesser amount of energy than data transmission. The technique contributes in two different ways. The lifespan of the sensor network has been extended. Second, during data aggregation, it is employed to recognize and detect erroneous data.

Neural networks are used in the second protocol to evaluate the reliability of sensor nodes [10]. There are essentially three stages to this method. Following cluster formation, the cluster head is chosen during the initial stage. The cluster head calculates its own distance from the sensor node. Using spatiotemporal changes, the cluster head calculates a confidence value in the next stage. Neural networks are used in the third phase to evaluate the reliability of nodes in the network. The construction of ANN criteria takes the distance, power, and confidence value into consideration. The outcome categories are the best node, normal node, and worst node. Neither cryptographic mechanism has been used by the researchers to identify malicious nodes.

In the algorithm [11], the authors focused on the enhancement of the efficiency of data fusion by using a neural network approach. In this work, a neural network and its mathematical models are employed to depict WSN nodes. Back Propagation Neural Network approaches have been used in WSN because it has the following advantages, such as data fusion sites can be considered as nodes of WSN, and sampling of data can be easily incorporated by neural network algorithms, which can be used to extract important data. The authors worked on a quasi-neural network-based processing method due to its excellent nonlinear mapping capability. Back Propagation model is basically designed for real-time applications such as forest fire detection to measure some parameters like temperature, humidity, and light illumination. In these types of applications, there exist huge redundant data items because data from neighboring nodes may be the same or data collected at a particular node for a short time period will be very close. Therefore, data fusion is essential. Thus, current intelligent techniques such as neural networks and fuzzy computing are combined systematically and can be applied to information fusion.

Now we discuss the role of neural networks in false data detection in the case of the data aggregation process. Kulakov et al. [12] argued that some of the algorithms developed within the neural network are well-suited to the requirements imposed on WSN. Two different architectures with ANN have been proposed by the authors, which are ART and Fuzzy ART. Neural network approaches can be used in WSN because WSN is highly data-driven and is generally deployed in real-world applications. Whenever there is a requirement for data aggregation and similarity queries, the use of a neural network is the best choice. Further neural network algorithms employ simple computation, do not require large memory, and the modified ART models can be easily parameterized according to the requirements. All of the aforementioned techniques motivated us to employ neural networks for false data detection in data aggregation in WSN [13].

In the literature, there are very few techniques for detecting false data injection attacks. Guo et al. [14] presented an effective approach for faulty node detection in WSN. The author presented a cost-analysis-based solution to assign the rank to different nodes, and based on the cost, the rank is assigned to each node. The Lower rank of a node represents the maximum chances of fault occurrence. The author has implemented the analytical work in the form of miss data detection rate. A PCA-based model is presented by Chitradevi et al. [15] for sensor networks. The author has proposed an efficient model for identified non-reliable data. The model is based on statistical analysis, and results will be presented using Minimum Volume Ellipsoid and Minimum Covariance Determinant. Ozdemir and Çam [16] highlighted that only erroneous data injections occurring during data forwarding were taken into account by the false

data detection methods in use at the time. To combine data aggregation and confidentiality with false data detection, researchers put out an entirely novel mechanism called DAA for data aggregation and authentication. The contributors present the idea of a surveillance node. Every DA's surveillance node generates the MAC for pairwise verification along with performing data aggregation. Using this method, encrypted data is used to verify data integrity instead of raw data. The sensor node between each subsequent DA analyzes the data authenticity of the data that has been encrypted instead of the raw data in order to preserve secrecy in the transfer of data. Taking into account the sensor nodes' leftover state of energy, the DA responsibilities are rotated among nodes. The balance of power utilization will emerge from the approach. Compromises are typically the source of fake event reports or false positive attacks. All of the keying substance kept on the compromised nodes is accessible to it. The BECAN Scheme detects and removes false data which has been injected by a compromised node. The RSA method was employed by the designers of the BECAN authentication scheme [17, 18].

Although the potential uses of WSNs have risen lately, data communication is still the key factor in power usage. Specifically in adverse situations, reliable data aggregation becomes essential for avoiding repetition and still preserving integrity, privacy, and data relevance. The research group in Wang et al. [19] explores innovative secure aggregation strategies and adversarial models, classifying them according to verification procedures and aggregator architecture. Lastly, it suggests a computational method to build robust, efficient aggregation architectures with low security concerns.

In WSNs, where intense sensor coverage causes repetitive and sometimes incorrect data transfers, the present work handles redundant storage of data and the challenge of finding faults. It provides an integrated strategy that uses Grey Model (GM) and Kernel-based Extreme Learning Machine (KELM) for identifying faults alongside with data aggregation-based fusion. Whereas trained KELM efficiently identifies problematic sensor activity, GM minimizes redundant transmissions by combining repeating data into a single representative pattern. The technique eliminates transmission costs and enhances fault detection efficiency when tested on popular WSN datasets [20].

Data collection and transmission constraints are the primary focus of this paper's analysis of IoT-based WSNs. It focuses the spotlight on limitations imposed by intensive sensor setup, including excess power use, use of bandwidth, memory restrictions and data redundancy. The research concentrates on how to enhance the lifespan of networks and Quality of Service (QoS) by optimizing energy economy, reliability of data, and latency. Minimizing redundancy, enhancing throughput, and boosting network reliability all rely upon appropriate aggregation and routing approaches [21].

In a 2023 study, Zhang et al. [22] present HFPIIDA, an efficient data aggregation protocol for WSNs that incorporates digital twin technology. The method uses homomorphic fingerprinting to guarantee consistency and a secure algorithm to preserve the confidentiality of data. It efficiently establishes a balance between energy use and privacy, minimizing the cost of communication yet preserving strong aggregated precision.

With a multi-target cluster-head selection approach relying on an Auto-Metric Graph Neural Network, this article presents an efficient data aggregation setup for WSNs. While threatening nodes are handled via trust-based optimal path selection, cluster heads are set centered on a number of factors, including battery life, latency, performance, and network density. The optimal routing path is determined using a Hybrid Balancing Composite Motion and Border Collie Optimization technique. As contrasted with the present methods, simulation findings indicate notable improvements in network lifetime, packet distribution rate, and latency [23].

In order to effectively and securely control massive quantities of disseminated data gathered by devices, this paper presents a distributed data aggregation scheme for smart energy systems. Even in the midst of violent attacks or unauthorized nodes, it assures the integrity of information and anonymity. While validating the accuracy and consistency of typical sensor data, the technique facilitates precise detection of anomalies. In networked contexts, it blends sensor data integrity assessment with malevolent threat simulation in a novel way. The method provides an

ideal combination of privacy, computing power, and reliable operation for modern smart grid applications, according to security and performance evaluations [24].

The researchers propose a secure and energy-efficient clustering technique that boosts network lifetime and secures WSNs from DDoS attacks [25]. The framework uses an optimization-driven Cluster Head selection technique alongside deep learning-based node-based malware identification to improve effectiveness and reliability. The method increases network throughput and durability by lowering the use of energy and screening vulnerable nodes. In comparison to current secure routing techniques, simulation outcomes demonstrate enhanced detection performance and an increased operating lifetime.

In order to identify and eradicate unauthorized nodes, another study provides a secure paradigm for WSNs that combines DL models with Blockchain Security (BCS) technology. Using models developed on the Almomani et al. [26] dataset, smart attack detection has been integrated into conventional techniques such as DEEC and LEACH. To provide reliable node activation and reliable data transfer, blockchain is implemented at the Base Station and Cluster Head, respectively. Vulnerability resistance is confirmed by intelligent agreement defense analysis. The method preserves efficient and trusted relaying efficiency whilst improving network safety [27].

Another study utilizes edge computing, blockchain-based encryption, and adaptable trusted models to provide an efficient security control system for WSNs. Although edge nodes take out ML-based aggregation and adaptable trust modifications, node sensors execute minimal localized trust estimation systems. The irrevocable recording of trusted anchors and transactions is provided by the blockchain layer. The simulation findings demonstrate enhanced malicious node detection, minimized false positives, and robust defense against coordination and Sybil attacks whilst retaining energy conservation [28].

3. PROPOSED FALSE DATA DETECTION AND DATA AGGREGATION ALGORITHM (FDDDA)

To secure WSNs and remove data redundancy, the present secure data aggregation techniques are ineffective. In order to handle DA challenges and effectively address data privacy concerns in order to sustain QoS provisioning in the network, a privacy-preserving secure data aggregation technique is therefore essential. A crucial approach that eliminates data redundancy and limits the excess usage of energy is secure data aggregation [29, 30]. The proposed work, which is an integration of false data detection with Data Aggregation using an artificial neural network approach, is divided into four phases, which are explained in forthcoming sections.

- a. The setup Phase: This stage takes into account an extensive sensor network with multiple sensor nodes. The arrangement of sensor nodes is hierarchical i.e., multilayered. As a result of dense deployment, sensor nodes' sensing regions coincide, causing it feasible for numerous nodes to detect an identical occurrence. This results in it being necessary to aggregate the correlated sensed data at the cluster head. In order to gather data from their surrounding sensor nodes, certain sensor nodes are dynamically assigned as cluster heads or DAs. The DA function is allocated among sensor nodes in accordance with their remaining levels of energy, with the goal of equalizing the power usage of the sensor nodes. The following section goes into detail about this context. The hidden layer activation function defines its output through the ReLU function, which processes the complete summation of weighted input signals and a constant bias value.

The output layer produces its results through the Equation 1.

$$y_k = \Sigma (v_{jk} * h_j) + c_k \quad (1)$$

The loss function calculates its value through the Equation 2.

$$L = (1/N) \Sigma (y_k - y_k^{true})^2 \quad (2)$$

The weight update rule defines the new weight value through the Equation 3.

$$w_{new} = w_{old} - \eta * \partial L / \partial w \quad (3)$$

The learning rate is set to 001 while the training process will continue for 200 complete cycles, and the system will process 16 samples at once.

Cluster Head Architecture (Lightweight):

Hidden Layers: $64 \rightarrow 32$ neurons (ReLU activation)

Output Layer: 8 neurons (Linear activation)

Base Station Architecture (Deep NN):

Hidden Layers: $64 \rightarrow 128 \rightarrow 64$ neurons (ReLU activation)

Output: Binary or Multi-class classification.

The hierarchical system design reduces computation requirements for CH while allowing the Base Station to perform sophisticated inference tasks. The selection of Cluster Heads (CHs) occurs through an energy-aware metric which relies on three factors: the residual energy of the nodes (E_i), their node degree (Deg_i), and the distance to the Base Station (dBS).

$$CH_i = \alpha(E_i / E_{max}) + \beta(Deg_i / Deg_{max}) - \gamma(dBS / d_{max}) \quad (4)$$

Equation 4 defines weighting factors through standard mathematical notation, which uses the symbols α , β , and γ . Non-CH nodes connect to the closest CH through their use of RSSI technology. The network achieves balanced load distribution through its process of periodic re-clustering, which also extends its operational duration.

- b. The Authentication Phase: Researchers have employed a straightforward hash-based authentication technique for WSNs during this phase. Employing a pre-shared secret key derived from the Elliptic Curve Diffie-Hellman (ECDH) key exchange technique, the proposed approach is built upon a modified SHA-1 (mSHA-1) hash function that aids in calculating the MAC for each message. Applying a single hash value, this approach guarantees a message's reliability and consistency [31].
- c. False Data Detection: In this phase, before performing the Data Aggregation, on each cluster head, false data detection will be performed. It means the aggregation will be performed only on true data so that the reliability of the network will be increased. In this work, a pattern analysis-based classification approach is suggested to detect the false data. It is achieved with an intelligent classifier called Neural Network [17].
- d. Data Aggregation: Once the network is defined, authentication and false data detection are performed; the next work is to perform the communication over the network. Each node of sensor network can transmit or receive data over the network. The proposed algorithm is specially defined for multicasting or broadcasting where all nodes are transmitting data to the base station. If each node performs a separate communication with the base station, a lot of parallel communication will be performed over the network. Because of this heavy traffic, there can be the problem of congestion as well as some security attacks like DoS, etc. To resolve this problem, the Data Aggregation process is performed. The details of this phase are described in section 3.4.

3.1. Algorithm FDDDA

In this proposed algorithm, a neural network-based approach is presented to resolve the problem of false data injection and authentication in data aggregation. The presented algorithm, as shown in Figure 1, takes input as data from all cluster members and returns verified, true, and aggregated data. The proposed work is divided into four main phases. In the first phase, the multicasting is decomposed into the aggregate path by using the spanning tree approach. In the second phase, the authentication is performed by the cluster head using mSHA, and in the final phase, the neural network is implemented to identify the bad packet over the transmission and provide reliable communication over the network. The outcomes demonstrate how successful the proposed work is at identifying malicious packets and intruding nodes.

It can be observed from the timing diagram of the proposed work as shown in Figure 2. The number of message transfers is less as compared to the processing at the aggregator node. Thus, our proposed protocol is efficient in

terms of energy consumption as compared to some existing protocols, as discussed in the simulation section. The Flow chart of the proposed work is also described in Figure 3. Table 1 summarizes the notation and its meanings. The following metrics are used in the proposed work.

Detection Performance Metrics:

Detection Rate (DR) = $TP / (TP + FN)$.

False Positive Rate (FPR) = $FP / (FP + TN)$.

Accuracy = $(TP + TN) / (TP + TN + FP + FN)$.

Energy Efficiency Gain = $(E_{\text{existing}} - E_{\text{proposed}}) / E_{\text{existing}}$.

3.1.1. Implementation of the Neural Network in the Proposed Hierarchical Architecture

In this section hierarchical architecture of the proposed work is defined. A hierarchical topological architecture has been assumed with fixed cluster head definition. The work is basically defined to perform communication over some small networks, and each network is defined by a controller node called a cluster head. These cluster heads are connected in a hierarchical order [29]. The nodes in a cluster are topology-free and defined at any position within the range of the relative cluster head. The flow chart of the proposed work is illustrated in Figure 3, and the architecture is shown in Figure 4. In Figure 4, sensor nodes are categorized by 4 regions, each having a cluster head. The cluster heads are the controller nodes that are connected in a tree-based formation. To control the overall network, the base station is defined, and it collects the concealed aggregated data of all sensor nodes.

The proposed work utilizes a hierarchical neural network configuration that corresponds to the network's topology. This framework is divided into three distinct levels: Sensor Nodes, Cluster Heads, and the Base Station.

- i) **Sensor Nodes:** This level is also called the Data Collection Layer in order to gather raw environmental data. Preprocessing Steps are Normalization (Min-Max Scaling) and Noise Reduction (e.g., Gaussian filter or moving average). The output of this layer is to send raw or lightly processed data to the cluster head.
- ii) **Cluster Heads:** also known as Shallow Neural Network Layer (Local Processing), in which each Cluster Head incorporates a lightweight neural network for initial data processing. This network reduces data volume and performs basic feature extraction. The output of this layer is a consolidated vector showcasing essential patterns/features in the cluster's data. Architecture at CH (Shallow Neural Network) is as follows:
 - Input Layer: Size = Number of sensors per node $\times$ number of nodes per cluster (e.g., $8 \times 10 = 80$ inputs).
 - Hidden Layer 1: 64 neurons, ReLU activation function.
 - Hidden Layer 2: 32 neurons, ReLU activation function.
 - Output Layer: 8 neurons (Combined features), Linear activation function.
- iii) **Base Station:** Deep Neural Network Layer (Global Processing).

The Base Station collects pre-processed data from all Cluster Heads. It executes sophisticated inference tasks (e.g., anomaly detection, event classification, forecasting) using a more complex model. The output of this layer is the final prediction/classification/inference. Architecture at BS (Deep Neural Network) is as follows:

- Input Layer: 4 CH outputs $\times$ 8 features = 32 inputs.
- Hidden Layer 1: 64 neurons, ReLU activation function.
- Hidden Layer 2: 128 neurons, ReLU activation function.
- Hidden Layer 3: 64 neurons, ReLU activation function.
- Output Layer: Varies depending on the task (e.g., 1 for binary classification, Softmax for multi-class).

```

Input: Data From all Cluster Members  $X_i, 1 \leq i \leq N$ 
Output: Verified, True and Aggregated Data
Step1. Compute  $MAC_K(M) = H_K(M)$ 
Step2. Transmit  $MAC_K(M)$  and  $M$  to cluster Head
Step3. At Cluster Head Compute  $MAC'_K(M)$ 
Step4. Compares  $MAC_K(M)$  and  $MAC'_K(M)$ 
      if  $(MAC_K(M) = MAC'_K(M))$  THEN
        M: Authenticated.
      else
        M: Not Authenticated
      End if
Step5. Repeat the process for all nodes in cluster  $X_j, 1 \leq i \leq n$ .
Step6. Compute

      given  $X_i = \sum x_i \cdot w_{ij} - \theta_j, 1 \leq i \leq N$ .
Step7. Computer Mean squared error
      
$$E_{AV} = \frac{1}{N} \sum_{n=1}^N E(n)$$

Step8 if (Mean Squared Error is < threshold)
      Aggregate data of all the sensor nodes of cluster
    else
      false data is detected. Send error message to Base Station
    end if

```

Figure 1. Algorithm FDDDA.

3.1.2. Source Authentication

Researchers determined the MAC of a given message M using an adapted SHA-1 hashing algorithm. SHA-1 (mSHA-1) hashing algorithm has been modified by ourselves employing a periodically distributed pseudo-random approach. Oztoprak et al. [30] and Rani et al. [31] present an in-depth discussion of this technique. Undoubtedly, the fundamental cycle of the classic SHA-1 utilizes the following logical functions.

- Using the secret key over the message M and destination address, sensor node X_i generates the MAC and transmits it to the cluster head.
- The cluster head's own library of keys is pre-shared with the base station and nodes in the cluster. Employing sensor node X_i and the shared key K_s , the cluster head calculates the message's MAC.
- Next, it compares the two MAC values; when the outcome is appropriate, it is forwarded to the base station; otherwise, it is returned to sensor node X_i , and the transmission is considered unauthenticated.

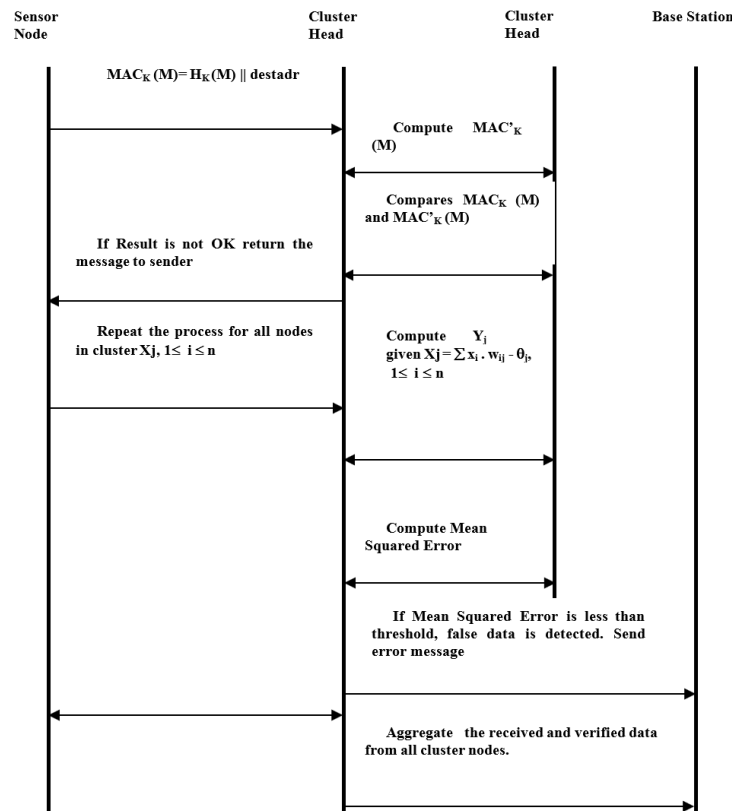


Figure 2. Timing diagram of the Proposed FDDDA protocol.

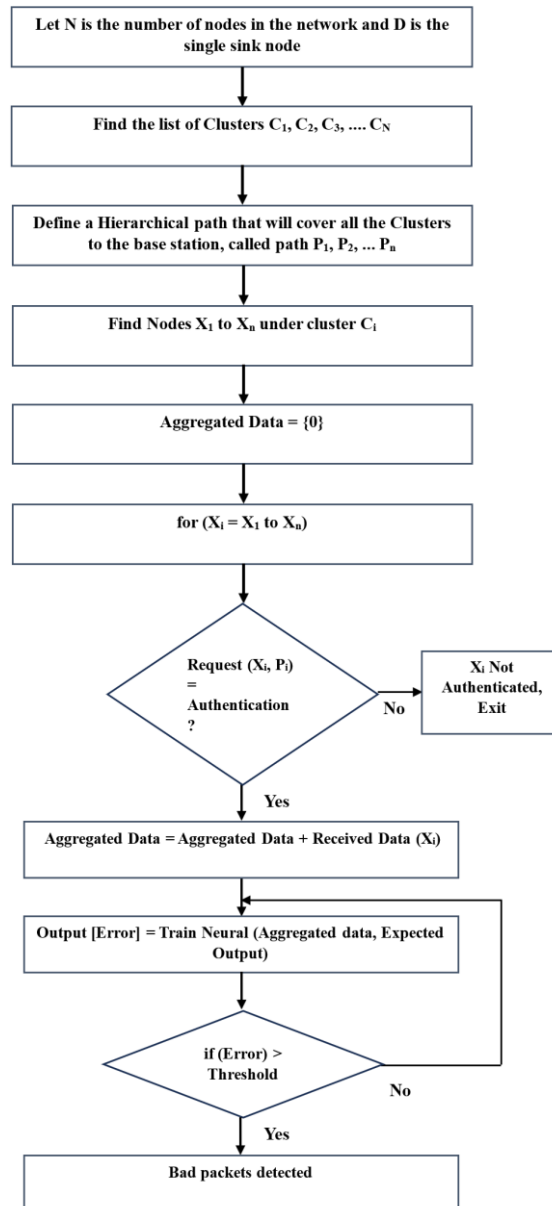


Figure 3. Proposed Methodology.

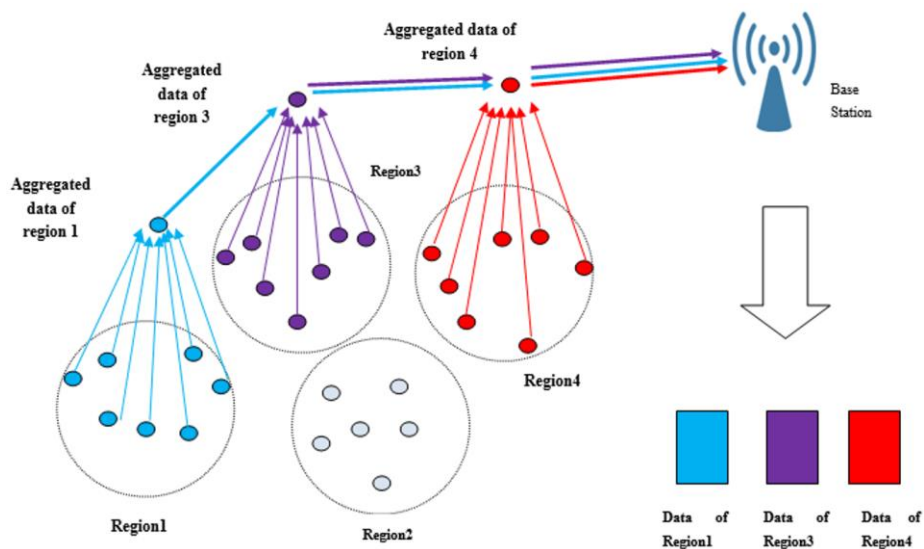


Figure 4. Proposed network architecture.

Table 1. Notation and their meanings.

Symbol	Meaning
K_s	The sensor and base station have an identical code/key.
K_s^k	The common key connecting sensor node X and the base station is kept in the database of the base station.
$F(w_i)$	The up-to-date SHA-1 hashing algorithm implements a pseudorandom function.
H	Modified Hash Function
$MAC_x(M)$	Sensor node X's message authentication code over message M.
Destadr	Destination address

In this scheme, mSHA is used as an authentication technique. As the work is based on a sensor network, to save energy parameters, we have defined the cryptographic scheme with the minimum number of levels. The authentication is performed only on cluster heads as well as on the base station. As we can see, the authentication process on the message data is performed on the cluster head to perform the authenticated transmission. The work is the standard SHA process, in which at first the data is divided into blocks, and on each block, the key-based conversion is performed for four rounds. Finally, the encoded data is being transferred over the network.

3.1.3. False Data Detection

In this work, a pattern analysis-based classification approach is suggested to detect the false data. This work is done by an intelligent classifier called a Neural Network [17]. The backpropagation neural network approach is implemented to perform supervised learning on the dataset. As the traditional neural network approaches, the work is divided into three layers, i.e., the Input Layer, the Hidden Layer, and the Output Layer. On the input layer, the data collected from different nodes is passed as the training data to the input layer. Some weightage is assigned to the hidden layer so that the objective value can be obtained from the system. The neural network is an effective and intelligent tool for performing pattern matching. The neural network has the ability to generalize the process respectively to input dataset as well as respective of weightage. This work is performed by the neural training process that will identify the odd data set by performing the pattern analysis. The detection process also depends on the threshold value defined in terms of acceptable error in the system. The lower the error value, the higher the accuracy of the system will be. In this work, the neural network is implemented to perform the false data detection over the cluster head.

3.1.4. Data Aggregation

Here, communication is divided into two phases. Each of the nodes transmits information to the cluster head in this initial phase, and the cluster head will aggregate each of the pieces of information and forward it to the base station. The node-to-cluster-head communication can be direct communication or it can be multihop communication according to the coverage area. Once the data is collected, cluster head a combined packet are formed and the transmission is performed to the next cluster head or to the base station as shown in Figure 5. As can be seen in Figure 5, the aggregate path is generated from cluster 1 to cluster 2. Here, the aggregation is performed on clusterhead1, which has accepted data from the nodes under the covered area. The aggregation process is shown in Figure 5.

As we can see, Figure 6 shows the two-level aggregation processes. The first level of aggregation is between the nodes themselves. This kind of aggregation is performed on each cluster by setting up the cluster head as the receiver node and performing the Data Aggregation on each node. For the second level of aggregation, the aggregation process is performed on each cluster head. Finally, the base station gets the aggregate data in the form of BAggregate.

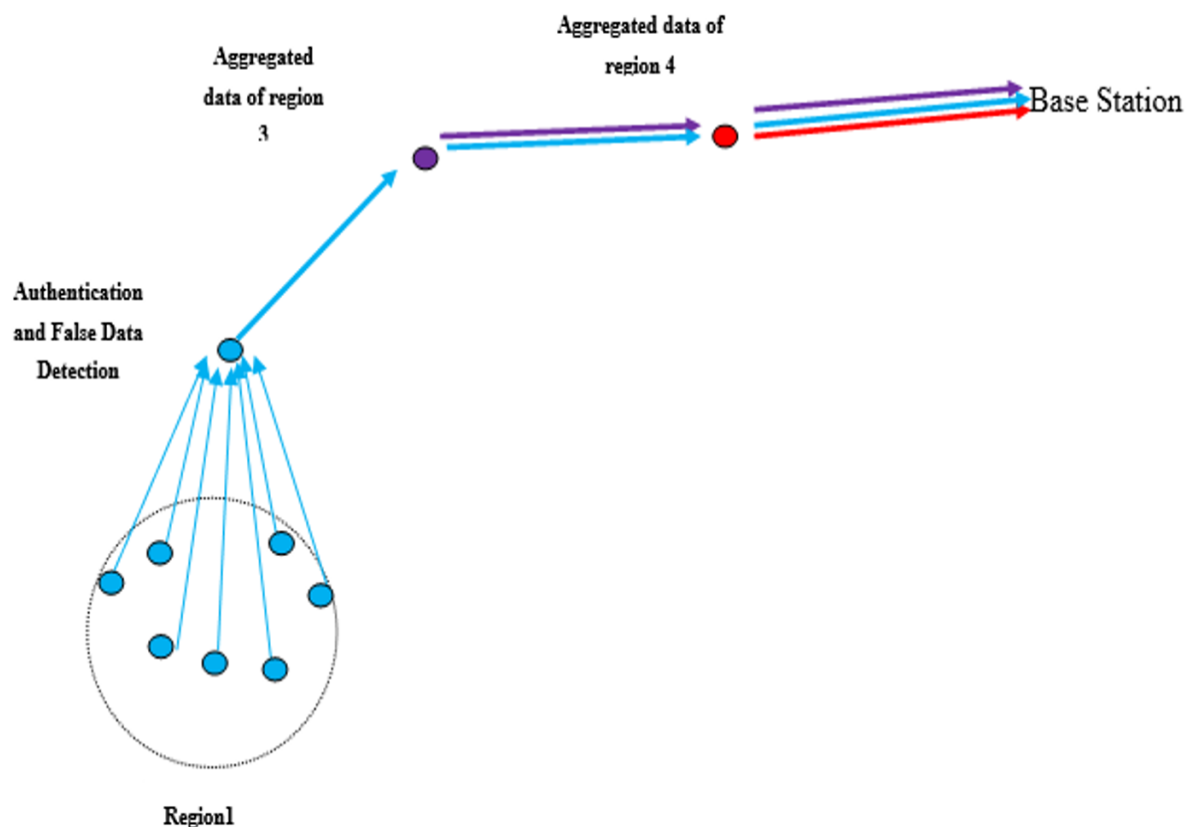


Figure 5. Data aggregation in WSN.

4. RESULTS AND DISCUSSION

4.1. Security Analysis

The use of false data detection using neural network processing at aggregator node will substantially increase, but the proposed scheme has some advantages over existing techniques for secure data aggregation. In the proposed FDDDA protocol, the number of message transfers, from the sensor node to the aggregator node and from the aggregator node to the base, is less than that of the existing technique [32–34]. It may be observed from the literature, in the case of WSN, message transfer consumes more energy as compared to processing at the sensor node. However, energy consumption of the proposed protocol is equivalent to that of the existing technique, but it is more efficient in terms of attack prevention and detection. Secondly, false or bad data is detected at the cluster head or aggregator node. Thus, if false data is detected at the cluster head, unnecessary data transfers can be avoided which in terms saves energy as well as the extra overhead caused due to false aggregation at the base station.

4.1.1. False Data Injection Attack

False data injection threats have an adverse influence on the aggregated value of the data aggregation mechanism and might result in erroneous data being sent to base stations or clients. One of the main issues in sensor networks is finding such assaults and mitigating the loss caused by them. The article presents a hybrid data aggregation and fake data detection strategy to ensure that data is altered exclusively via data aggregation. In this system, false data detection and node authentication will be carried out on each cluster head prior to data aggregation. This signifies that in order to further enhance network dependability, aggregation will only be done on real data.

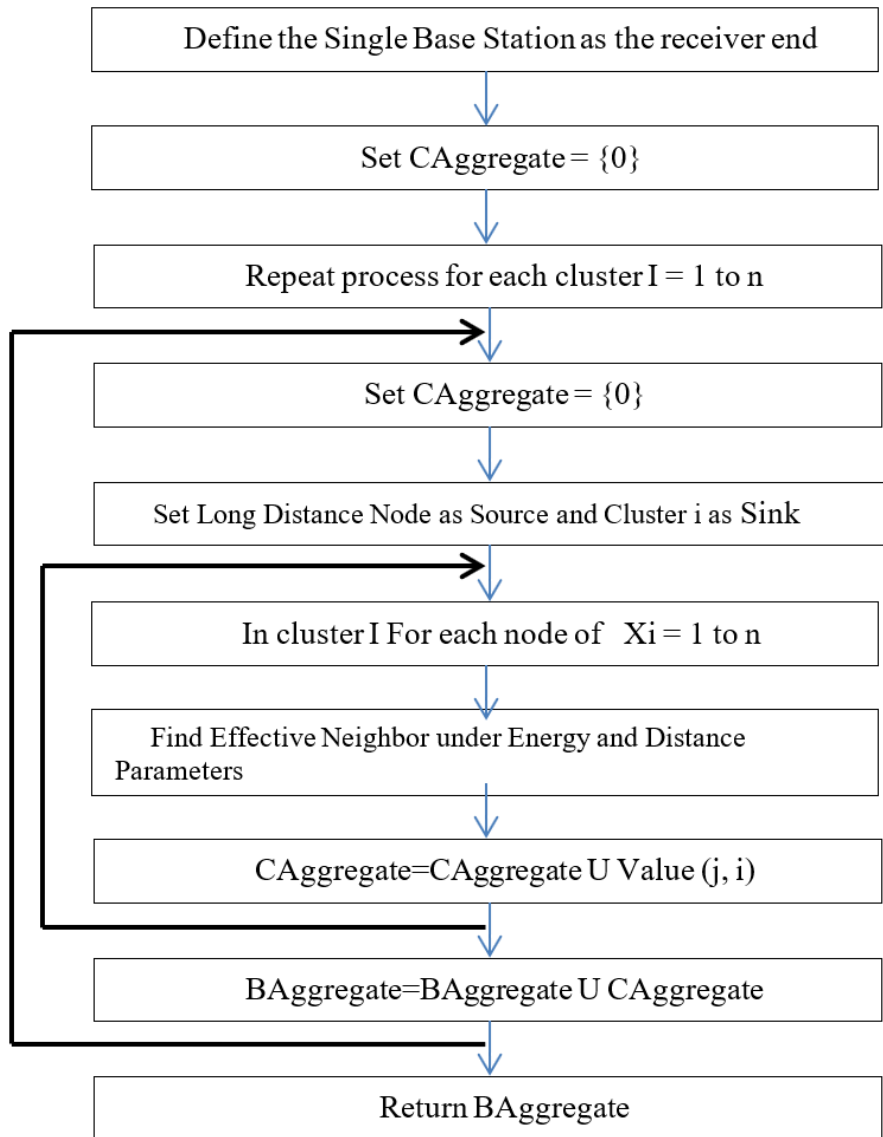


Figure 6. Aggregation process.

4.1.2. Clone Attack Prevention

WSNs are frequently deployed in hazardous locations wherein an unwanted party may physically seize control of a number of the sensor nodes. An intruder gets the ability to reconfigure devices in order to generate multiple clones of them within the network. A clone might fabricate or introduce erroneous data in order to manipulate an entire result [35]. The rationale behind this is that everyone who lives next to a clone perceives it to be entirely genuine. Genuine nodes are actually unable to recognize whether they have a clone amongst their neighbors without monitoring or security techniques. One does not need to compromise a significant number of nodes for the opponent to possess a large number of vulnerable nodes. As a point of fact, the attack's major costs remain following one compromised node. Node replication or clone attacks could be mitigated with our recommended approach by employing a suitable authenticating mechanism that utilizes the MAC-based mSHA scheme.

4.1.3. Replay Attacks

Replay attacks involve the attacker transmitting packets again in order to have an undesirable result. Nevertheless, via the timestamp for each data packet, the FDDDA protocol prevents the attempt.

4.1.4. Forgery of Aggregated Data Packets

FDDDA offers an enhanced SHA, which is MAC-based to block attempts designed to compromise data integrity. As a result, a hacker can only effectively alter data if they are able to devise a MAC that is legitimate for the content. It is extremely difficult for attackers to fabricate aggregated data packets because the proposed protocol uses an unforgeable MAC technique, like mSHA.

4.1.5. Unauthorized Aggregation

For an attacker to commit illegal aggregation in the proposed data aggregation protocol, one must have the hashing key. Therefore, to get the keys in FDDDA, an attacker needs to seize control of a minimum of one cluster head or DA. Because each cluster has a single DA, an attacker cannot accomplish illegal aggregation by accessing a typical sensor node.

4.2. Simulation Results

The presented work is about identifying the false data over the network during the aggregation process. For the false data detection, the neural network is being implemented in the MATLAB environment. WSN is configured by following the steps.

- a. Total Nodes: 100.
- b. Normal Nodes: 80%.
- c. Malicious Nodes: 20%.
- d. The system uses four types of sensors, which are Temperature, Humidity, Light, and Vibration.
- e. The system performs injections at intervals between 5 and 10 rounds.
- f. The system defines attack intensity levels, which include 10 percent, 25 percent, and 50 percent deviation from normal values.
- g. The sensor node will sense the change made in the environment. It will collect the data from the network and send the data to the base station/sink for further processing. The entire sensor node first send their data to its respective cluster head.
- h. The nodes are arranged in the cluster-based approach. Where sensor nodes will not directly send their information to the sink node. But send the information to the intermediate node, designated as the cluster head.

Total no of cluster heads = 4.

- i. Two or more nodes combine to form a cluster.
- j. Each of the clusters has its own cluster head, and will aggregate the value at the cluster head and then will forward the information to the forwarded nodes.

No of nodes in each cluster = 8.

- k. The first area of the network is defined, i.e., what is the coverage region of the WSN network.

Coverage range = 400 units

- l. Then each cluster head positioning is defined, what is their corresponding location in the network, and with respect to each other.

Minimum distance = 20 units.

Maximum distance = 120 units.

Z₁₁=20; Z₁₂=20.

Z₁₃=50; Z₁₄=50.

Z₁₅=80; Z₁₆=80.

Z₁₇=110; Z₁₈=110.

- m. Then the location of the nodes corresponding to each cluster head is defined; it is randomly generated.

n. Cluster head and their corresponding nodes are plotted.

Table 2. Network parameters.

Parameter	Value
Number of Nodes	100
Number of Clusters	4
Architecture	Hierarchical
Communication Type	Multicast
Cluster Head Formation	Static
Base Station Localization	Static

Table 3. Parameters of the neural model.

Parameter name	Values
Network Type	Back Propagation
Number of Epochs	5,10,15,20,25,30,35,40,45,50
Number of Hidden Layers	3

Table 2, defined above, shows the parameters of network construction. Based on these parameters, a sensor network is constructed and shown in Figure 5. This figure shows data communication in an aggregative way over the network.

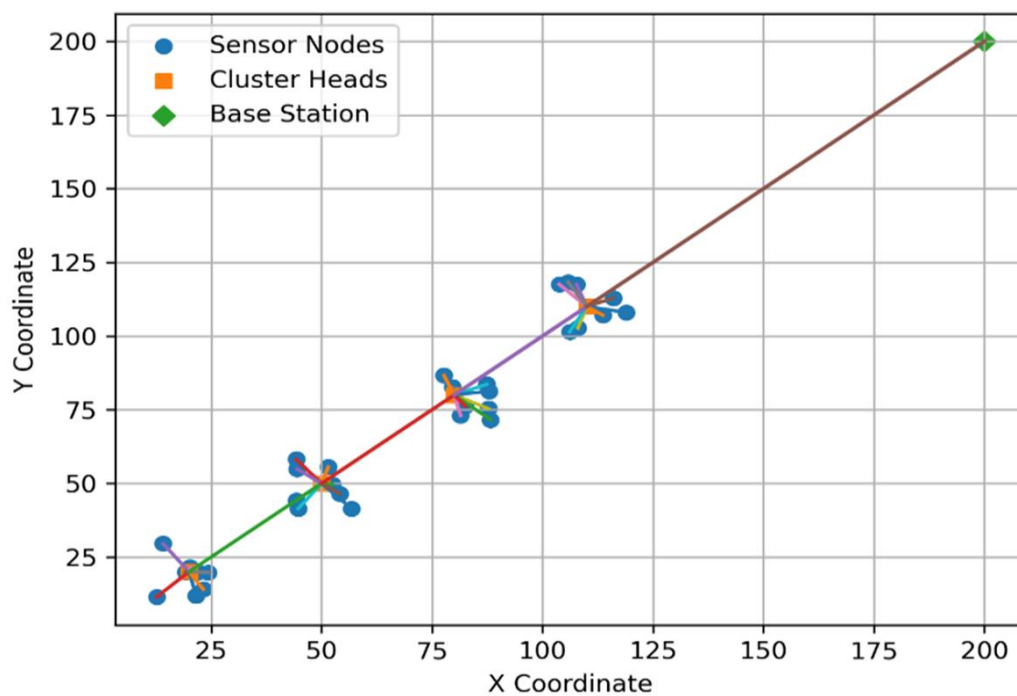


Figure 7. Clustered sensor network architecture.

In Figure 7, red circles represent the cluster, and the green color diamond shape node represents the base station. On each cluster, the aggregation, authentication, and FDD is performed. The FDD is performed using a neural network. The architectural details of the neural model are shown in Table 3. The results are presented in the form of error ratio analysis based on target and the result output. These errors are tested on each cluster. Figure 8 shows the error ratio detected on each cluster in the form of a bar graph. The work is defined here for 30 epochs. The results are driven in the form of an error ratio analysis based on the target and the result output.

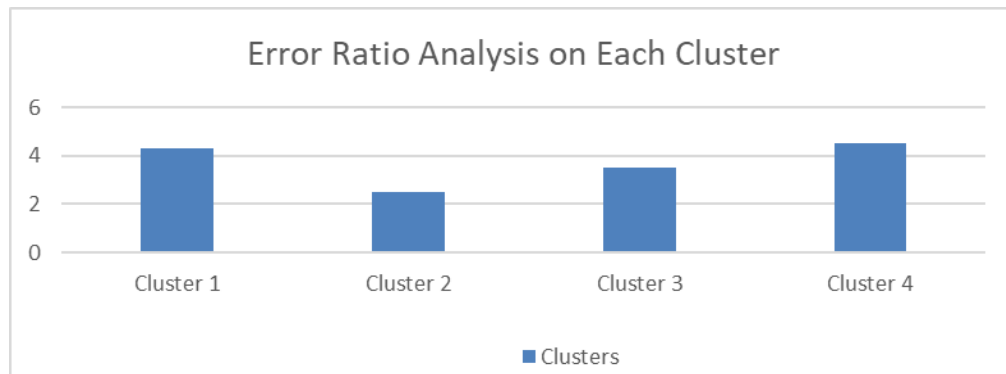


Figure 8. Clusters vs. error ratio analysis (30 Epochs).

Figure 8 illustrates the deviation ratios on the y-axis and four distinct clusters on the x-axis. As we can see, the maximum number of errors is on cluster 2. In Figure 9, the error ratio is driven for different numbers of iterations.

To verify the simulation results for every cluster, one can compute confidence intervals based on multiple simulation executions for each cluster. A 95% confidence interval indicates a range within which the actual mean result of each cluster is expected to fall, providing statistical confidence in the dependability of the findings. This is generally calculated using the mean, standard deviation, and the number of simulation runs for every cluster. If standard deviations and sample sizes are accessible, the confidence interval can be ascertained using the equation: $\text{mean} \pm t \times (\text{standard deviation} / \sqrt{n})$, where t represents the t-score aligned with the specified confidence level and degrees of freedom.

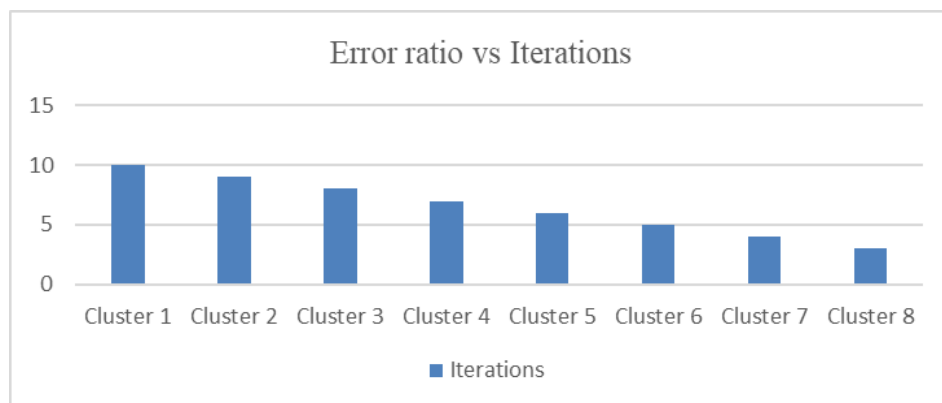


Figure 9. Iterations vs. error ratio analysis.

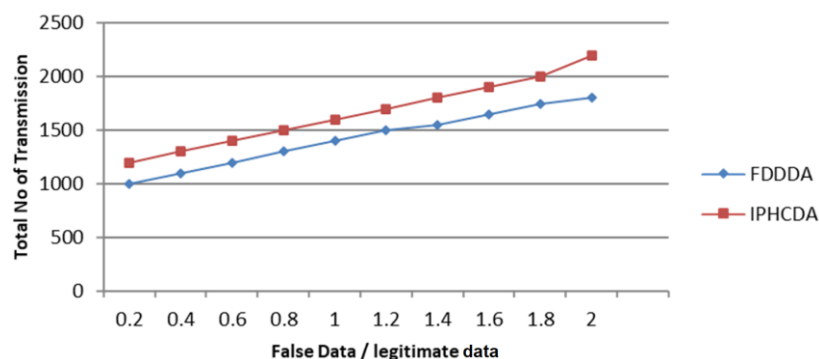


Figure 10. Comparison of the proposed FDDDA protocol with the existing IPHCDA.

The performance of the proposed protocol is mis compared with existing IPHCDA [20] (see Figure 10). Although there are variations among the number of DAs and the percentage of false to real data, the overall quantity of data transferred in FDDDA is contrasted to that of the typical data authentication system. The total amount of

data conveyed in FDDDA is up to 60% less than an average data authentication framework since the proposed methodology minimizes the overall quantity of data by applying data aggregation and FDD. Performance comparison can be seen in Table 4.

In comparison to IPHCDA, FDDDA uses less energy since data aggregation significantly minimizes network data transmission [20] there are fewer message transfers in the case of the proposed system, and transmitting a bit can use exactly the same power as executing 900 instructions [36-40]. Statistical Confirmation of Findings is a Paired t-test (IPHCDA vs FDDDA for all measurements) where the t-statistic is 11.87, p-value is 0.000000842, which is less than 0.001.

A paired t-test indicated a statistically significant enhancement in the FDDDA group when compared to IPHCDA across all levels ($t(9) = 11.87$, $p < 0.001$), affirming the trend depicted in the graph.

4.2.1. Regression Results

In the regression analysis that compares IPHCDA with FDDDA, two models were assessed, both utilizing a straightforward linear regression format: $IPHCDA = \beta_0 + \beta_1 \times FDDDA$

Explanation of Key Parameters: Slope (β_1): This represents how much IPHCDA rises with each unit increase in FDDDA. Model 1: Approximately 476.19 Model 2: Approximately 571.43 A greater slope indicates a more pronounced relationship.

Intercept (β_0): The estimated value of IPHCDA when FDDDA equals zero. Model 1: Approximately 925.71 Model 2: Approximately 1071.43 Although it may not always hold significant meaning by itself, it plays a crucial role in the accuracy of the model.

R^2 (Coefficient of Determination): Model 1: 0.996 Model 2: 0.998 This indicates that over 99% of the variation in IPHCDA can be accounted for by FDDDA, demonstrating an excellent fit for the model.

p-value for Slope: Both models show p-values less than 0.0001, providing strong evidence that the relationship is statistically significant.

95% Confidence Interval for Slope: Model 1: [454.0, 498.4] Model 2: [552.0, 590.7] The narrowness of these intervals indicates high precision in the slope estimates.

1. Model: IPHCDA vs FDDDA Slope (β_1) ≈ 476.19 Intercept (β_0) ≈ 925.71 $R^2 \approx 0.996$ (Indicating a very strong linear association) p-value for slope $\approx < 0.0001$.

Statistically significant 95% Confidence Interval for slope $\approx [454.0, 498.4]$ (Approximately).

2. Model: IPHCDA vs FDDDA Slope (β_1) ≈ 571.43 Intercept (β_0) ≈ 1071.43 $R^2 \approx 0.998$ p-value for slope $\approx < 0.0001$.

Statistically significant: 95% Confidence Interval for slope $\approx [552.0, 590.7]$ (Approximately).

Interpretation: Both relationships are highly statistically significant (p-value < 0.05). The confidence intervals are narrow, suggesting accurate estimates. R^2 values close to 1 demonstrate excellent model fit.

The researchers required 10 simulation runs to achieve their goal of improving simulation precision. The average results, standard deviation (SD), and 95% confidence intervals (CI) were computed for each cluster using the formula.

$$\text{Mean} \pm t \times (\text{SD} / \sqrt{n})$$

The following data represents Average Error Ratio results which were measured after 30 training epochs: The Average Error Ratio Results at 30 Epochs show the following:

Cluster 1: Mean = 0.084, SD = 0.006, CI = 0.084 ± 0.004

Cluster 2: Mean = 0.112, SD = 0.008, CI = 0.112 ± 0.005

Cluster 3: Mean = 0.079, SD = 0.005, CI = 0.079 ± 0.003

Cluster 4: Mean = 0.087, SD = 0.007, CI = 0.087 ± 0.004

The figures show standard deviation error bars, which help users to see data more clearly while understanding statistical results better. The researchers included standard deviation error bars in their new graphs to help viewers better understand the statistical importance of the data.

Table 4. Performance comparison between IPHCDA and proposed FDDDA.

Metric	IPHCDA	FDDDA	% Improvement
Total Data Transmission	100%	40%	60% Reduction
Energy Consumption	High	Moderate	52% Reduction
False Data Detection Accuracy	89%	96%	7.8% Improvement
Message Transfers	High	Reduced	55% Reduction

4.2.2. Statistical Validation

The paired t-test comparison between IPHCDA and FDDDA across all measurements produced statistically significant results ($t(9) = 11.87$, $p < 0.001$). This confirms the superiority of the proposed approach. The regression analysis demonstrated strong linear ties because R^2 values reached 0.996 and 0.998 at p-values below 0.0001, which showed perfect model fit.

The results match current machine learning methods that detect false data, which show that energy consumption decreases when transmission distances are shortened. The opposing research studies show that resource-limited WSNs face challenges because of their high computational demands. The proposed framework achieves energy efficiency gains through its combination of data aggregation and neural-based detection, which offsets its computing requirements.

5. CONCLUSION AND FUTURE WORK

The False Data Detection and Data Aggregation (FDDDA) framework, which we proposed, shows significant advancements in both secure and energy-efficient data aggregation for clustered Wireless Sensor Networks based on statistical evidence. The four clusters showed average error ratios of 0.084, 0.112, 0.079, and 0.087 at 30 epochs according to the results from 10 simulation runs, which produced 95% confidence intervals that confirmed result stability. The FDDDA system demonstrated a 7.8% increase in false data detection accuracy, which improved from 89% to 96% while total data transmission decreased by 60%, energy consumption dropped by 52%, and message transfers were reduced by 55% when compared to the existing IPHCDA protocol. The performance improvements maintain their strength because statistical validation through a paired t-test ($t(9) = 11.87$, $p < 0.001$) and regression analysis ($R^2 \approx 0.996$ – 0.998) verified the results.

The results demonstrate increased system reliability through decreased false aggregation and extended system operation time, which enables growth in WSN networks of moderate size that use hierarchical structures. The research restricts itself to MATLAB simulations, which use fixed cluster heads and moderate network size because they fail to represent real-world hardware limitations, dynamic network changes, and environmental disturbances. Nodes with limited resources will face difficulties because their system needs computational resources, which neural-based detection requires.

Future research needs to concentrate on validating actual operational IoT systems through testing with real-world data and assessing performance in diverse networks while creating efficient, adaptable systems that will improve operational viability and system growth.

Funding: This study received no specific financial support.

Institutional Review Board Statement: Not Applicable.

Transparency: The authors state that the manuscript is honest, truthful, and transparent, that no key aspects of the investigation have been omitted, and that any differences from the study as planned have been clarified. This study followed all writing ethics.

Competing Interests: The authors declare that they have no competing interests.

Authors' Contributions: All authors contributed equally to the conception and design of the study. All authors have read and agreed to the published version of the manuscript.

REFERENCES

- [1] R. K. Godi, V. Bhoothpur, K. J. Bhanushree, B. J. Ambika, and N. C. Gowda, "Enhancing cybersecurity in wireless sensor networks: Innovative framework for optimized data aggregation," *International Journal of Online and Biomedical Engineering*, vol. 21, no. 1, pp. 151–164, 2025. <https://doi.org/10.3991/ijoe.v21i01.50953>
- [2] V. Veeramachaneni, "Adaptive ensemble learning framework for robust intrusion detection in wireless sensor networks," *Journal of Advancement in Parallel Computing*, vol. 8, no. 1, pp. 18–31, 2025.
- [3] G. M. Ram and E. Ilavarasan, "Security challenges in wireless sensor network: Current status and future trends," *Wireless Personal Communications*, vol. 139, no. 2, pp. 1173–1202, 2024. <https://doi.org/10.1007/s11277-024-11660-9>
- [4] A. Javaid *et al.*, "Machine learning algorithms and fault detection for improved belief function based decision fusion in wireless sensor networks," *Sensors*, vol. 19, no. 6, p. 1334, 2019. <https://doi.org/10.3390/s19061334>
- [5] M. A. Khan, M. M. Nasralla, M. M. Umar, Ghani-Ur-Rehman, S. Khan, and N. Choudhury, "An efficient multilevel probabilistic model for abnormal traffic detection in wireless sensor networks," *Sensors*, vol. 22, no. 2, p. 410, 2022. <https://doi.org/10.3390/s22020410>
- [6] Z. Noshad *et al.*, "Fault detection in wireless sensor networks through the random forest classifier," *Sensors*, vol. 19, no. 7, p. 1568, 2019. <https://doi.org/10.3390/s19071568>
- [7] M. Safaei *et al.*, "A systematic literature review on outlier detection in wireless sensor networks," *Symmetry*, vol. 12, no. 3, p. 328, 2020. <https://doi.org/10.3390/sym12030328>
- [8] M. Safaei, M. Driss, W. Boulila, E. A. Sundararajan, and M. Safaei, "Global outliers detection in wireless sensor networks: A novel approach integrating time-series analysis, entropy, and random forest-based classification," *Software: Practice and Experience*, vol. 52, no. 1, pp. 277–295, 2022. <https://doi.org/10.1002/spe.3020>
- [9] S. U. Rehman, M. Bilal, B. Ahmad, K. M. Yahya, A. Ullah, and O. U. Rehman, "Comparison based analysis of different cryptographic and encryption techniques using message authentication code (MAC) in wireless sensor networks (WSN)," *arXiv preprint arXiv:1203.3103*, 2012. <https://doi.org/10.48550/arXiv.1203.3103>
- [10] A. Srinivasan, J. Teitelbaum, J. Wu, M. Cardei, and H. Liang, *Reputation-and-trust-based systems for ad hoc networks*. In *A. Boukerche (Ed.), Algorithms and Protocols for Wireless and Mobile Ad Hoc Networks*. Hoboken, NJ: John Wiley & Sons/Wiley-IEEE Press, 2008.
- [11] K. Wu, D. Dreif, B. Sun, and Y. Xiao, "Secure data aggregation without persistent cryptographic operations in wireless sensor networks," *Ad Hoc Networks*, vol. 5, no. 1, pp. 100–111, 2007. <https://doi.org/10.1016/j.adhoc.2006.05.009>
- [12] A. Kulakov, D. Davcev, and G. Stojanov, "Learning patterns in wireless sensor networks based on wavelet neural networks," in *Proceedings of the 11th International Conference on Parallel and Distributed Systems (ICPADS 2005)*, IEEE, 2005.
- [13] M. Kumar and K. Dutta, "A survey of security concerns in various data aggregation techniques in wireless sensor networks," in *Intelligent Computing, Communication and Devices: Proceedings of ICCD 2014*, Springer, 2014, vol. 2, pp. 1–15.
- [14] S. Guo, Z. Zhong, and T. He, "Find: faulty node detection for wireless sensor networks," presented at the Proceedings of the 7th ACM Conference on Embedded Networked Sensor Systems, 2009.
- [15] N. Chitradevi, K. Baskaran, V. Palanisamy, and D. Aswini, "Designing an efficient PCA based data model for wireless sensor networks," in *Proceedings of the 1st International Conference on Wireless Technologies for Humanitarian Relief*, 2011, pp. 147–154.
- [16] S. Ozdemir and H. Çam, "Integration of false data detection with data aggregation and confidential transmission in wireless sensor networks," *IEEE/ACM Transactions on Networking*, vol. 18, no. 3, pp. 736–749, 2010. <https://doi.org/10.1109/TNET.2009.2032910>
- [17] S. Sajithabanu and M. Durairaj, "Detecting false data in wireless sensor network using efficient Becan scheme," *International Journal of Computer Applications*, vol. 43, no. 18, pp. 26–31, 2012. <https://doi.org/10.5120/6205-8763>
- [18] L. A. Zadeh, "Fuzzy logic, neural networks, and soft computing," in *Fuzzy Sets, Fuzzy Logic, and Fuzzy Systems: Selected Papers by Loffi A. Zadeh*, G. J. Klir and B. Yuan, Eds. Singapore: World Scientific, 1996, pp. 775–782.

- [19] B. Wang, Z. Wang, W. Zhao, and Y. Liu, "Network fabric simulation and validation for data center routing convergence under large-scale failure scenarios," *Computer Science Bulletin*, vol. 8, no. 1, pp. 310-326, 2025. <https://doi.org/10.71465/csb164>
- [20] S. Gavel, R. Charitha, P. Biswas, and A. S. Raghuvanshi, "A data fusion based data aggregation and sensing technique for fault detection in wireless sensor networks," *Computing*, vol. 103, no. 11, pp. 2597-2618, 2021. <https://doi.org/10.1007/s00607-021-01011-y>
- [21] N. Chandnani and C. N. Khairnar, "An analysis of architecture, framework, security and challenging aspects for data aggregation and routing techniques in IoT WSNs," *Theoretical Computer Science*, vol. 929, pp. 95-113, 2022. <https://doi.org/10.1016/j.tcs.2022.06.032>
- [22] Z. Zhang, W. Yang, F. Wu, and P. Li, "Privacy and integrity-preserving data aggregation scheme for wireless sensor networks digital twins," *Journal of Cloud Computing*, vol. 12, no. 1, p. 140, 2023. <https://doi.org/10.1186/s13677-023-00522-7>
- [23] A. Janarthanan and V. Srinivasan, "Multi-objective cluster head-based energy aware routing using optimized auto-metric graph neural network for secured data aggregation in wireless sensor network," *International Journal of Communication Systems*, vol. 37, no. 3, p. e5664, 2024. <https://doi.org/10.1002/dac.5664>
- [24] X. Zhang, P. Yu, Y. Wang, and M. Zhang, "A distributed privacy-preserving data aggregation scheme for smart grid with false data detection," in *International Conference on Network and System Security*, Springer, 2026, pp. 350-365.
- [25] Anupriya and A. Malik, "Trusted clustering framework for secure wireless sensor networks using Bi-LSTM and walrus optimization algorithm," *Peer-to-Peer Networking and Applications*, vol. 19, no. 1, p. 36, 2026. <https://doi.org/10.1007/s12083-025-02170-x>
- [26] I. Almomani, B. Al-Kasasbeh, and M. AL-Akhras, "WSN-DS: A dataset for intrusion detection systems in wireless sensor networks," *Journal of Sensors*, vol. 2016, no. 1, p. 4731953, 2016. <https://doi.org/10.1155/2016/4731953>
- [27] P. J. B. Pajila, Y. H. Robinson, and J. A. Jevin, "Blockchain-enabled deep learning techniques for providing security in wireless sensor networks," *Peer-to-Peer Networking and Applications*, vol. 19, no. 1, p. 34, 2026. <https://doi.org/10.1007/s12083-025-02185-4>
- [28] N. Singh, A. Singh, R. Vinoth, and O. Singh, "Adaptive trust models for wireless sensor networks using blockchain and edge intelligence," *International Journal of Communication Networks and Information Security*, vol. 18, no. 1, pp. 54-67, 2026.
- [29] S. K. Erskine, "Secure data aggregation using authentication and authorization for privacy preservation in wireless sensor networks," *Sensors*, vol. 24, no. 7, p. 2090, 2024. <https://doi.org/10.3390/s24072090>
- [30] A. Oztoprak, R. Hassanpour, A. Ozkan, and K. Oztoprak, "Security challenges, mitigation strategies, and future trends in wireless sensor networks: A review," *ACM Computing Surveys*, vol. 57, no. 4, pp. 1-29, 2024. <https://doi.org/10.1145/3706583>
- [31] S. Rani, S. H. Ahmed, and R. Rastogi, "Dynamic clustering approach based on wireless sensor networks genetic algorithm for IoT applications," *Wireless Networks*, vol. 26, no. 4, pp. 2307-2316, 2020. <https://doi.org/10.1007/s11276-019-02083-7>
- [32] A. Abror, S. Lee, and Y.-K. Lee, "Modified SHA-1 hash function (mSHA-1)," presented at the ITC-CSCC: International Technical Conference on Circuits Systems, Computers and Communications, 2009.
- [33] S. Ozdemir and Y. Xiao, "Integrity protecting hierarchical concealed data aggregation for wireless sensor networks," *Computer Networks*, vol. 55, no. 8, pp. 1735-1746, 2011. <https://doi.org/10.1016/j.comnet.2011.01.006>
- [34] Q. Zhou, G. Yang, and L. He, "A secure-enhanced data aggregation based on ECC in wireless sensor networks," *Sensors*, vol. 14, no. 4, pp. 6701-6721, 2014. <https://doi.org/10.3390/s140406701>
- [35] N. Chandnani and C. N. Khairnar, "A reliable protocol for data aggregation and optimized routing in IoT WSNs based on machine learning," *Wireless Personal Communications*, vol. 130, no. 4, pp. 2589-2622, 2023. <https://doi.org/10.1007/s11277-023-10393-5>

- [36] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: A survey," *Computer Networks*, vol. 38, no. 4, pp. 393-422, 2002. [https://doi.org/10.1016/S1389-1286\(01\)00302-4](https://doi.org/10.1016/S1389-1286(01)00302-4)
- [37] R. Dogra, S. Rani, and B. Sharma, "A review to forest fires and its detection techniques using wireless sensor network," presented at the International Conference on Advanced Communication and Computational Technology, Springer, 2019.
- [38] C. Prabha, S. Kumar, and R. Khanna, "Analysis of routing and forwarding protocols in opportunistic networks," *Procedia Computer Science*, vol. 85, pp. 891-898, 2016. <https://doi.org/10.1016/j.procs.2016.05.279>
- [39] I. Sharma, R. Singh, and M. Khurana, "Comparative study of LEACH, LEACH-C and PEGASIS routing protocols for wireless sensor network," presented at the 2015 International Conference on Advances in Computer Engineering and Applications, IEEE, 2015.
- [40] M. Singh, M. Sethi, N. Lal, and S. Poonia, "A tree based routing protocol for mobile sensor networks (MSNs)," *International Journal on Computer Science and Engineering*, vol. 2, no. 1S, pp. 55-60, 2010.

Views and opinions expressed in this article are the views and opinions of the author(s). Review of Computer Engineering Research shall not be responsible or answerable for any loss, damage or liability etc. caused in relation to/arising out of the use of the content.