

Article

Energy-Efficient Secure Routing for a Sustainable Heterogeneous IoT Network Management

Ashok Thangavelu ^{1,*} and Prabakaran Rajendran ²

¹ Department of Biomedical Engineering, Kongunadu College of Engineering and Technology, Namakkal City 621215, India

² Department of Electrical and Electronics Engineering, Bharathidasan Institute of Technology, Anna University, Tiruchirappalli City 620024, India; hiprabakaran@gmail.com

* Correspondence: ashokt.ece@gmail.com

Abstract: The Heterogeneous Internet of Things (H-IoT) is considered as the upcoming industrial and academic revolution in the technological world, having billions of things and devices connected to the Internet. This H-IoT has a major issue of energy consumption during data transmission which leads to low scalability. Additionally, anomalies in the data create a serious threat to energy in H-IoT. To overcome these issues, a novel approach has been proposed in this study termed as the Energy-Efficient Memetic Clustering Method (EEMCM), which combines the Parallelized Memetic Algorithm (PMA) with the AlexNet architecture to improve anomaly detection efficiency in IoT WSNs. Initially, cluster formation and CH selection are carried out using PMA. This is followed by routing path generation, and the data are prepared for high-level feature extraction. The extracted features are classified to identify anomalies. For anomaly detection, high-level features were collected that contain data relevant to the model given as input into the AlexNet architecture, which detects anomalies and identifies normal or potential attacks within the IoT WSNs. The proposed EEMCM model has been implemented in the MATLAB platform and obtained an accuracy of 99.11%. As a result, the overall performance of the network is improved.

Keywords: heterogeneous IoT; wireless sensor network; cluster formation; anomaly detection; Parallelized Memetic Algorithm (PMA); AlexNet



Citation: Thangavelu, A.; Rajendran, P. Energy-Efficient Secure Routing for a Sustainable Heterogeneous IoT Network Management. *Sustainability* **2024**, *16*, 4756. <https://doi.org/10.3390/su16114756>

Received: 24 April 2024

Revised: 21 May 2024

Accepted: 30 May 2024

Published: 3 June 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

1. Introduction

Heterogeneous Internet of Things (H-IoT) networks, which are distinguished by a variety of IoT devices, are the next evolution of networking infrastructures [1]. H-IoT is based on the development of multi-networking heterogeneous models, and effective communication tools for industrial as well as personal applications have been developed [2]. Since most end devices, such as sensors, run on batteries, it becomes imperative to look at energy consumption. Understanding how much energy is used in wireless transmission is crucial for estimating power usage [3]. An IoT-based WSN [4] constitutes various specialized sensors involved in various applications [5] like animal monitoring, environmental sensing, disaster management [6], habitat monitoring, intelligent transportation, healthcare, transport, armed forces surveillance, and weapon control [7,8]. As a sensor's lifespan depends on its batteries [9], during harsh conditions in which they operate or are impossible to replace or recharge, an issue arises and threatens the WSN's incorporation into the IoT. As a result, an extended network lifetime is seen as a significant obstacle for WSN-based IoT. Thus, a clustering technique is employed in the WSN to increase energy usage and network lifetime [10].

By decreasing spare data transfer between the nodes, and with reliance only on local information, clustering is a well-liked method for lowering energy use [11]. Rare clustering protocols [12] like Power-Efficient Gathering in Sensor Information Systems (PEGASIS), Low-Energy Adaptive Clustering Hierarchy (LEACH), Hybrid Energy Efficient Distributed

(HEED) clustering, Threshold-Sensitive Energy Efficient Sensor Network (TEEN), Adaptive Threshold-Sensitive Energy Efficient Sensor Network (APTEEN), and base-station controlled dynamic clustering protocol were used, while the clustering protocols present poor results [13]. CH plays a critical role in optimizing energy utilization. Based on a few criteria, different clusters of sensor nodes in a WSN are created [14]. Numerous methods used optimization algorithms to suggest CH selection. Some of them are Particle Swarm Optimization (PSO), Artificial Bee Colony (ABC), and Whale Optimization Algorithm, which failed, due to a low number of parameter selections [15]. However, the weaknesses of IoT systems, such as limited resources, heterogeneous devices, and a lack of standardization, make them prone to dangerous faults and attacks [16,17].

Denial of Service, Ransomware, Botnet, SCADA/Trojan Horse, Packet Sniffing, and Man-in-the-Middle are some of the attacks that have had an impact on IoT [18]. For certain of the identified attacks, malicious disruptions of the data flow inside the IoT network must be avoided to decrease energy consumption. Early detection of intrusions helps stop attacks before they have a chance to do permanent harm [19]. More specifically, attack classification requires powerful processing units, such as GPUs and massive storage, which are needed for Machine Learning (ML) and Deep Learning (DL) [20]. Still, these attacks impede the creation of novel applications and intelligent systems. When compared to the state-of-the-art, the proposed research has rectified premature convergence, improved network lifetime with less energy usage, and obstructed attacks during communication [21,22].

The remarkable contributions of this work are listed below:

- Previous works focused less on threading attacks and data leakage in Heterogenous IoT. Hence, to overcome this limitation of research, a new security-based routing approach has been proposed.
- Memetic Algorithm (MA) for the selection of CH and feature extraction has been proposed to overcome the energy consumption, scalability and dimensional reduction issues.
- An Energy-Efficient Memetic Clustering Method (EEMCM) is proposed to tackle challenges like limited node energy, outlier detection, threats, data leaks, and redundancy, to reduce energy consumption.

The remainder of the document is arranged as follows. Section 2 describes the existing research. Section 3 describes the Energy-Efficient Memetic Clustering Method (EEMCM). Section 4 describes the results and presents a discussion about EEMCM. Section 5 discusses the outcome of the work and Section 6 closes the analysis with a conclusion.

2. Literature Survey

Alghamdi et al., 2020 [23] created a clustering model by taking into account four key factors such as energy, delay, distance, and security, for Cluster Head (CH) selection. Additionally, to choose the CHs, the analysis suggested a hybrid algorithm called Firefly replace the position update in Dragonfly, which combined the ideas of the two methods. The WSN paradigm focused on lifespan improvement and structural data flow, while fewer parameters were taken for the CH selection.

To choose the CHs in the IoT-WSN, Raslan et al., 2021 [24] provided an Improved Sunflower Optimization Algorithm (ISFO) algorithm. It was integrated by the Lévy flight operator with the Sunflower Optimization Algorithm (SFO). The Lévy flight operator was able to assist the SFO in avoiding early convergence and escaping from being trapped in the local minima. But then, the remaining population was stuck in the intermediate area if the overall best solution reached local minima.

Pour & Javidan et al., 2021 [25] suggested a clustering algorithm that considered the distance between nodes and sink, network density, and residual energy of nodes before selecting CHs. There was a significant decrease in power consumption as the number of rounds increased. Additionally, it focused each round's calculation on a variable range that was established by network density, though it was high computation.

A clustering method with four steps of processing—cluster formation, splitting and merging, CH selection, and data transformation—based on Internet of Things devices was

introduced by Poluru & Ramasamy et al., 2020 [26]. By the k-mean clustering approach, the network was clustered. To address every optimization issue, the Cyclic Rider Optimization Algorithm (C-ROA) was introduced. The performance of energy harvesting was not fully covered here.

Sankar et al., 2023 [27] suggested a Sandpiper Optimization Algorithm (SOA) that takes residual energy and distance into account when forming clusters and selecting the right CH node to increase network lifetime. The sink was positioned in the middle of the network region, and the simulation continued with various network sizes. Security was not taken into consideration in this work.

Bakshi et al., 2021 [28] presented a technique for adaptive CH selection based on Glow-worm Swarm Optimization (GSO). The network as a whole was divided into an optimal number of clusters by the method. The approach ensured minimized overlapping of the clusters using a voting mechanism that minimizes the communication overhead. Due to the limited step-size, GSO requires a trade-off between accuracy and convergence speed.

WSN clustering based on the memetic algorithm (MemA) was suggested by Ahmad et al., 2021 [29] to use local exploration strategies needed to reduce the likelihood of early convergence. MemA was used to dynamically balance the load among clusters to obtain optimal clusters in WSN-IoT. Finding a Cluster Head Set (CH-set) as soon as feasible when a requirement arises was the aim of the analysis. In the next generation, the Wireless Mobile Nodes (WMNs) with the highest weight value were chosen rather than newcomers. For a memetic algorithm search, there was frequently a maximum computational budget that was used.

To lower the energy consumption of WSN and increase both the stability period and network lifetime, an Energy-Efficient Two-Stage routing Protocol (EETSP) was presented by Dwivedi et al., 2021 [30]. By using this protocol, the CHs and Secondary Cluster Heads (SCHs) lower the network's energy consumption while also greatly increasing the number of packets transmitted to the Base Station (BS). The BS's processing power, energy supply, and computing capability were all constrained. Using symmetric key techniques, Hybrid Secure Multipath Optimized Routing (HSMOR) was presented by Kala et al., 2021 [31] with multi-variation tuples. The adversaries in the WSN were discovered and prevented by the suggested strategy. With the use of a sophisticated symmetric key methodology, it also organized an authentication and encryption model to select the sensor hubs. Performance metrics considered here were fewer, and better analysis could have been performed for this work.

For WSN-based IoT networks with energy constraints, Fixed-Parameter Tractable Approximation Clustering (FPTAC), an energy-aware data distribution mechanism was developed by Yarinezhad & Sabaei et al., 2021 [32]. FPTAC presented an approximation algorithm needed to solve LBCP with an approximation ratio of 1.2. FPTAC was outfitted with a routing mechanism to ascertain the pathways within the network, thereby completing the protocol for WSN-based IoT networks. Routing algorithms have an energy-hole problem because they ignore the nodes' positions and current energies.

3. Proposed Methodology

This paper proposes an EEMCM based on efficient cluster formation and CH selection and integrates the AlexNet architecture for anomaly detection to address energy consumption and security challenges in H-IoT WSNs, as shown in Figure 1.

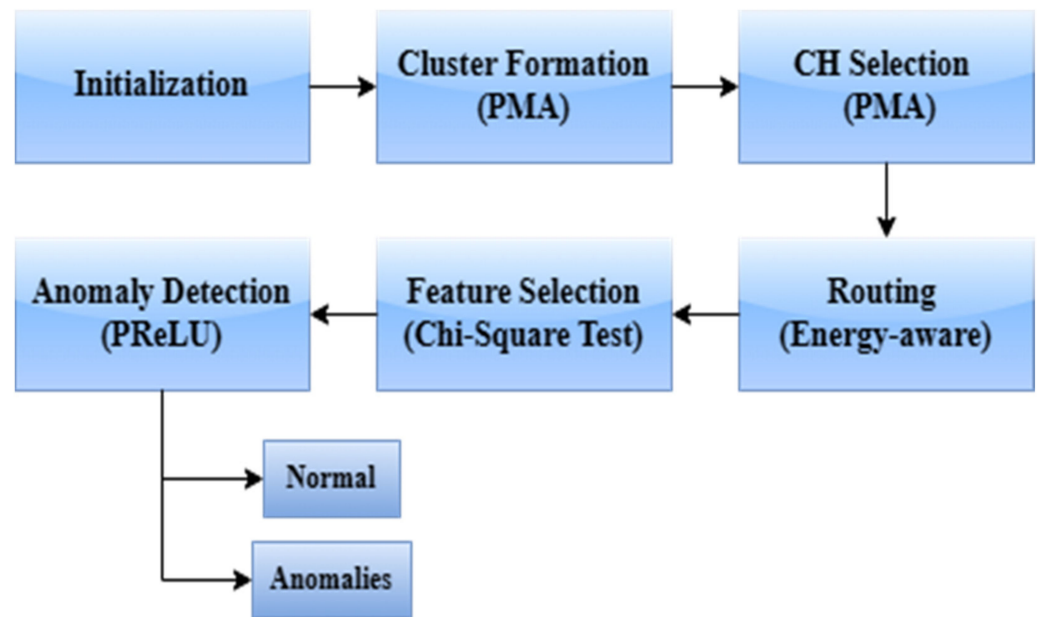


Figure 1. Block Diagram for Proposed Methodology.

3.1. Initialization

During the Initialization phase, m number of sensor nodes are deployed. Each IoT node performs self-configuration tasks, by the integration of communication parameters in the network connections like network topology, node density, and environmental conditions.

$$\text{Initial sensor nodes } (S_i) = \{S_{i1}, S_{i2}, \dots, S_{in}\} \quad (1)$$

In Equation (1), S_i represents the initialization of m number of sensor nodes in the network as S_{in} .

$$\text{Parameter Setup } (P_i) = \{P_{i1}, P_{i2}, \dots, P_{in}\} \quad (2)$$

In Equation (2), P_i represents the communication parameters such as network topology, node density, and environment data to set up in the sensor node S_i .

$$\text{Network Connections } (A) = \begin{bmatrix} a_{11} & a_{12} & \dots & a_{1n} \\ a_{21} & a_{22} & \dots & a_{2n} \\ \dots & \dots & \dots & \dots \\ a_{n1} & a_{n2} & \dots & a_{nn} \end{bmatrix} \quad (3)$$

In Equation (3), A represents the adjacency matrix of the network, where a_{ij} represents the connection status between the sensor nodes S_i and S_j . Each entry a_{ij} can take on a value of 1 if a connection between sensor nodes S_i and S_j exists and 0 if there is no connection.

3.2. Cluster Formation

In this phase, m , the sensor nodes are deployed in the network and formed as a cluster. For this grouping, the methodology utilizes the Memetic Algorithm (MA) for cluster formation in H-IoT. MA combines the strengths of evolutionary algorithms with local and global search techniques for efficient exploration of solution spaces while refining candidate solutions towards optimality. However, MA may struggle to handle multimodal optimization issues in which the search space is complex. Large-scale multimodal optimization problems also provide scalability challenges, resulting in longer convergence periods. To address these issues, data parallelism techniques are implemented to distribute the computational workload across multiple processors into the MA as Parallelized-MA (PMA) for cluster formation. PMA involves dividing data into smaller subsets and distributing them across multiple processing units. Each unit independently computes on its assigned subset,

enabling parallel execution. So, the data points can be processed independently, allowing for efficient cluster formation over huge data. By doing this, PMA significantly reduces the convergence time and enhances its scalability in dealing with large-scale multimodal optimization problems.

Step 1: The MA fitness function is defined, which evaluates the fitness function and measures how well each cluster configuration satisfies the constraints of the network, such as efficient data routing, balanced energy consumption, and effective communication.

$$MA(F(x)) = \sum_{i=1}^n f_i(S_i) \quad (4)$$

In Equation (4), $F(x)$ is the highest fitness value of the candidate solution x , which corresponds to a particular arrangement of sensor nodes into clusters. It calculates the highest fitness of a candidate solution x by summing the fitness values $f_i(S_i)$ derived from parameters such as node connectivity, energy levels, and proximity to other nodes of individual nodes (S_i).

$$Local\ Search(x) = \lambda(x, Neighbour(x)) \quad (5)$$

In Equation (5), the local search operator of $\lambda(x)$, explores the neighbour of a candidate solution x . A global search also needs to be performed.

$$Global\ Search(x) = \lambda(population) \quad (6)$$

In Equation (6), the global search operator, $\lambda(population)$ explores the entire population of candidate solutions. The operator λ is applied to the entire population, selecting common elements among all candidate solutions in the population.

$$T(S_i, S_j) = \begin{cases} S_i & \text{if } f(S_i) > f(S_j) \\ S_j & \text{if } f(S_i) \leq f(S_j) \end{cases} \quad (7)$$

Equation (7) represents the selection operator used in the Memetic Algorithm. It compares the fitness values of the candidate S_i , S_j and selects the one with the higher fitness for further evolution.

$$C(S_i, S_j) = \begin{cases} S_i^K & \text{with probability } \frac{1}{2} \\ S_j^K & \text{with probability } \frac{1}{2} \end{cases} \quad (8)$$

Equation (8) represents the uniform crossover operator 'C' used in the Memetic Algorithm. It selects genes from either of the parent solutions, S_i or S_j , with an equal probability of creating offspring solutions.

$$p = (S_i, P_i, A) \quad (9)$$

In Equation (9), p represents the PMA for cluster formation. PMA distributes the computational workload across multiple processing units, enabling parallel execution of cluster formation tasks to improve scalability.

3.3. CH Selection

Throughout this stage, after the cluster formation, the approach leverages the abilities of the PMA for CH selection in H-IoT. This method assigns a probability to each sensor node based on its fitness value for individuals to determine the probability of selection to serve as a CH. Subsequently, in PMA, the cluster formation of m sensor nodes data is divided into multitasks, and each task is assigned a fitness percentage that represents its completion efficiency. The task with the highest efficiency is often determined by completion time

and is designated as the CH. Integrating the fitness percentage method with PMA ensures efficient CH selection from the parallel processing capabilities of PMA.

$$P_i = \frac{f_i}{\sum_{j=1}^n f_j} \quad (10)$$

In Equation (10), P_i represents the probability of selecting the sensor node i as a CH. $\sum_{j=1}^n f_j$ represents the sum of fitness values of all sensor nodes in the network.

$$CH_i = \begin{cases} 1 & \text{with probability } P_i \\ 0 & \text{otherwise} \end{cases} \quad (11)$$

In Equation (11), CH_i denoting the binary variable indicates whether sensor node i is selected as a CH (1) or not (0) based on the probability.

$$E_i = \frac{\text{Num of completed tasks}}{\text{Total Num of task}} \quad (12)$$

In Equation (12), E_i represents the task completion efficiency of sensor node i out of the total number of tasks assigned in the parallel processing in the PMA.

$$F.P_i = \frac{E_i}{\sum_{j=1}^n E_j} \quad (13)$$

In Equation (13), $F.P_i$ represents the fitness percentage of sensor node i in the task completion efficiency.

$$CH_i = \begin{cases} 1 & \text{if } E_i = \text{Max}(E) \\ 0 & \text{otherwise} \end{cases} \quad (14)$$

In Equation (14), $\text{Max}(E)$ represents the maximum task completion efficiency among all sensor nodes in the network.

$$CH.PMA = \sum_{i=1}^n P_i.PMA(S_i, P_i, A) \quad (15)$$

Equation (15) signifies that the CH selection with PMA is determined by the summation of the fitness probability P_i of each sensor node multiplied by its corresponding output from the PMA. This integration ensures that CHs are selected efficiently based on both fitness probability and the parallel processing capabilities of PMA.

3.4. Routing

After the cluster formation and selection, the data are transmitted to the base station (BS) or sink. For this purpose, selecting the shortest path is necessary, and hence an energy-aware routing method for routing data to the BS in an H-WSN has been presented. This method balances energy consumption in the network and prevents network congestion. It considers both the distance to the BS and the energy levels of sensor nodes with potential routes when calculating the path cost. Sensor nodes evaluate multiple paths and select the one with the lowest energy-aware cost, ensuring efficient data transmission while conserving energy and providing a network lifetime. By integrating energy-aware routing techniques into the methodology, optimizing the data delivery to the BS in H-WSNs, the mitigation of energy constraint issues and improvement in network performance are achieved.

$$\text{Energy aware Routing}(\text{PathCost}_{ij}) = \text{Distance}_{ij} * \text{Energy}_{ij} \quad (16)$$

Equation (16), $PathCost_{ij}$, represents the cost of the path from sensor node i to j . $Distance_{ij}$ indicates the distance between the sensor node i and j and $Energy_{ij}$ indicates the energy level of the sensor node i to j .

$$Potential\ Routes = \{r_1, r_2, \dots, r_m\} \quad (17)$$

In Equation (17), Potential Routes represents the set of potential routes from the sensor node to the BS and r_m indicates the total number of potential routes.

$$Cost(r_k) = \sum_{i=1}^n PathCost_{ij} \quad (18)$$

In Equation (18), $Cost(r_k)$ represents the energy-aware cost with the route r_k , by summing the path costs of all sensor nodes along the route and $\sum_{i=1}^n PathCost_{ij}$ indicates the summation of the path cost of all sensor nodes i along the route r_k , of both distance and energy. To select the optimal path:

$$Optimal\ path = \underset{r_k \in PotentialRoutes}{\operatorname{argmin}} Cost(r_k) \quad (19)$$

Equation (19) represents the selected optimal routing path among the potential routes and $\underset{r_k \in PotentialRoutes}{\operatorname{argmin}} Cost(r_k)$ denotes the process of selecting the route r_k with the minimum energy-aware cost among all potential routes.

In the selection of an optimal routing path in H-WSNs by using the energy-aware routing method, efficient data transmission is ensured while minimizing the energy and improving the network performance.

3.5. Anomaly Detection

To improve the network performance and reduce energy consumption, our model aims to build an efficient chi-squared test for feature selection and AlexNet for anomaly detection. In H-WSN, the sensor node collects the data from their environment. The sensor nodes transmit the collected data wirelessly to a data sink for further analysis and processing. Moreover, the collected data are input into our AlexNet models, enabling effective anomaly detection and improving the performance of the IoT network.

3.5.1. Feature Selection

The chi-squared test is a filter-based method and is a well-known feature selection technique. The benefit of this feature selection method is that it is network efficient. A chi-squared test is used in statistics to test the independence of two events. The data are assigned with two variables, observed count OC and expected count EC . Chi-squared measures how the expected count of EC and observed count of OC deviates from each other. The high-level feature is detected by the chi-squared test, which is utilized to identify the most relevant features from the sensor data for anomaly detection. By selecting a subset of innovative features, the chi-squared test reduces the dimension of the data by improving the network efficiency.

$$S_{i,c} = \sum \frac{OC_i - EC_i}{EC_i} \quad (20)$$

In Equation (20), c represents the degree of freedom, OC is the observed count and EC is the expected count. The relationship is between the independent category feature (predictor) and dependent category feature (response). In our feature selection method, we aim to select the feature with a high dependence value on the utilization of the chi-squared test on the response, and the features are selected and fed to the AlexNet model for training, as shown in Figure 2.

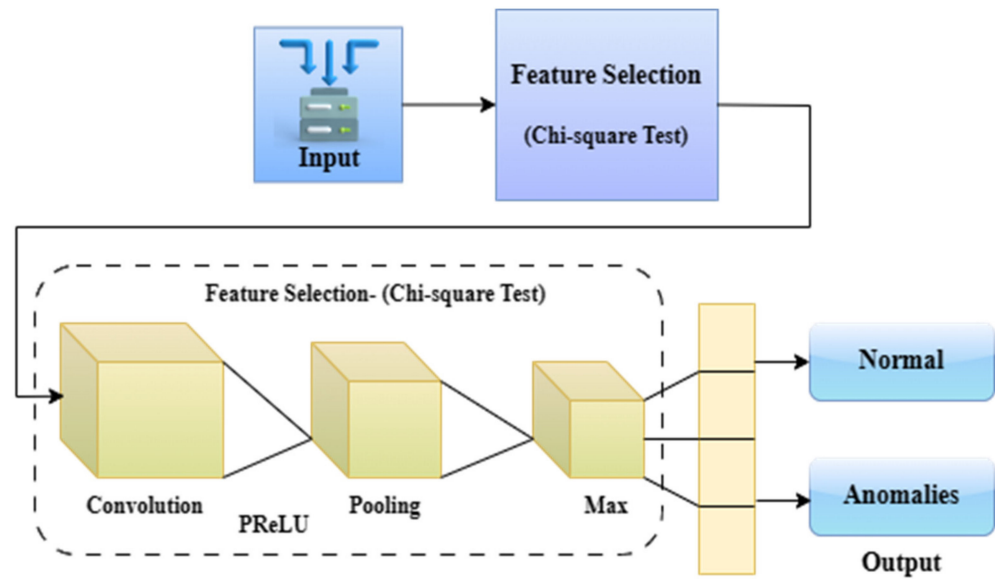


Figure 2. Proposed AlexNet Architecture.

3.5.2. Classification

One well-known DL architecture, AlexNet, is used mainly to identify irregularities in data transfers. In particular, dangers like Mirai and Bash lite attacks might try to enter systems while data are being transferred. The strong capabilities of AlexNet play a crucial role in spotting and stopping these types of intrusion attempts, protecting the security and integrity of the data transfer procedure. To minimize spatial dimensions and preserve the spatial relationships between nearby features, AlexNet makes use of overlapping pooling layers. Fully connected layers, pooling layers, and convolutional layers make up the AlexNet module. These layers work together as a sequential model to extract features. AlexNet contains input, hidden, and output layers. The input layer starts with the convolutional layers that apply an activation function such as the Parametric Rectified Linear Unit (PReLU). This activation function was employed in this investigation to handle the non-linear data, which effectively addresses anomaly detection and helps to avoid overfitting, providing better unseen data and reducing the risk in training time. After the input layer, the hidden layer has pooling layers that scale down the data to reduce the feature dimensionality; finally, fully connected layers learn the data and use them for classification. The use of AlexNet, a state-of-the-art architecture for any anomaly detection task may be highly advantageous for the computer vision fields of ML and intelligent processing problems. During this operation, the AlexNet model analyses incoming sensor data and the devices significantly identified from its normal patterns as anomalies. The classified operation is given by

$$O_{i,j,k} = \sum_{l=1}^{D_m} \sum_{l=1}^F \sum_{n=1}^F I_{i+m-1,j+n-1,l} \cdot K_{m,n,l,k} + b_k \quad (21)$$

In Equation (21), $O_{i,j,k}$ represents the output of the feature map, and I the pixel value of the input image at the location $i + m - 1, j + n - 1, l$.

$$PReLU = \begin{cases} f_x \cdot x & \text{if } x < 0 \\ x & \text{if } x \geq 0 \end{cases} \quad (22)$$

Equation (22) represents the PReLU activation function in the input layer, where x is the input to the activation function and f_x is the learnable parameter that adjusts the slope

of the negative part of the function. K is the weight of the kernel at the position (m, n) in channel l for filter k and b_k is the bias team of the filter k .

$$a_i = b(S_i) = \begin{cases} f_x S_i, & S_i < 0 \\ S_i, & S_i \geq 0, \end{cases} \quad (23)$$

In Equation (23), f_x represents a real number, S_i is the input value, and b is the PReLU Activation function.

The gradients of PReLU are

$$\frac{da_i}{dS_i} = b'(S_i) = \begin{cases} f_x, & S_i < 0 \\ 1, & S_i \geq 0, \end{cases} \quad (24)$$

$$\frac{da_i}{df_x} = \begin{cases} S_i, & S_i < 0 \\ 0, & S_i \geq 0, \end{cases} \quad (25)$$

In Equations (24) and (25) all of the slopes on the left have a starting value of 0. By leveraging the complementary strengths of the AlexNet for anomaly detection, this hybrid model aims to achieve efficient anomaly detection in WSNs to improve network performance and reduce energy consumption.

4. Results and Discussions

EECMCM has been implemented by MATLAB 2023a, with Windows 10 OS, Intel core i7 processor, and 8GB RAM, and their results are shown in the upcoming section.

4.1. Dataset Description

This dataset fills the gap in the public botnet datasets, particularly in the IoT domain. It suggests actual traffic data collected from nine commercial IoT devices that have been verified to be infected with BASHLITE and Mirai.

Dataset link: <https://www.kaggle.com/datasets/mkashifn/nbaiot-dataset> (accessed on 23 April 2024).

Use of formal techniques in this study:

Formal techniques used in this study are the Energy-Efficient Cluster Head Selection Mechanism for Livestock Industry using Artificial Rabbits Optimization (EECHS-ARO) [33], the Energy-Efficient Cluster Head Selection in Wireless Sensor Networks Using an Improved Grey Wolf Optimization (EECHIGWO) [22], the Osprey Optimization Algorithm based on Energy-Efficient Cluster Head Selection (SWARAM) [15] and Hybrid Snake Whale Optimization (HSWO) [34]. However, previously, there were a number of models that focused on minimization of energy consumption in WSN. These works excelled in results. Hence, these works are taken as formal techniques for comparison.

4.2. Performance Analysis of EECMCM

Operational efficiency is maximized and energy consumption is reduced by optimizing network configuration parameters like node density and network topology. The system performs better and uses its resources more efficiently because of optimized configuration, which guarantees more efficient network operations.

The graph in Figure 3a depicts the random node deployment of heterogeneous nodes in a WSN with 50 nodes and Figure 3b depicts the random node deployment of heterogeneous nodes in a WSN with 100 nodes. The placement of some nodes closer to one another and some farther apart results in a significant change in node density from that of random node deployment.

Figure 4a shows optimal path determination in WSN with 50 nodes. Figure 4b shows optimal path determination in WSN with 100 nodes. Optimally secure determination reduces routing overhead to offer fully secure data transfer. For this, energy-aware routing

is considered. This technique effectively prevents network congestion and balances energy consumption within the network.

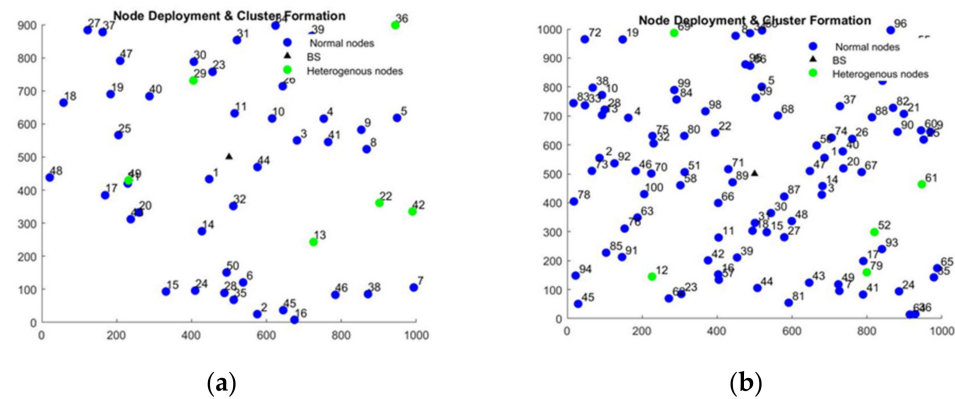


Figure 3. Random node deployment of heterogeneous nodes in a WSN: (a) with 50 nodes (b) with 100 nodes.

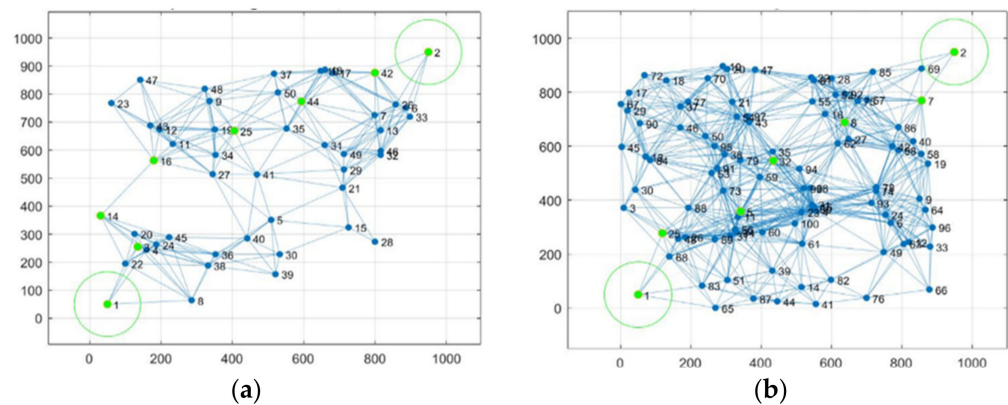


Figure 4. Optimal path determination in WSN. (a) 50 nodes (b) 100 nodes. The green dots represent the cluster heads, blue dots represent the cluster members and the green circles represent the source and destination.

From Table 1, it is seen that using the dataset the attacks are injected to the model and this model evaluated the attacks. Based on these statistics, the table shows the training time and execution time of each attack based on its features, which is tabulated.

Table 1. Attack type statistics of proposed model.

Attack Type	Packets	Training Time [min]	Execution Time [min]
Recon	1,697,851	33.3	18.9
Man in middle	2,504,267	14.2	20.1
DoS	2,771,276	18.7	34.1
Botnet (Mirai)	764,137	52	66.9

4.3. Comparative Analysis of Proposed Algorithm with Existing Algorithms

4.3.1. Comparative Analysis of Proposed MA with Existing Algorithms

The simulation results for the WSN's performance, taking into account dead nodes, are shown in Figure 5a, with existing techniques like EECHS-ARO [33], HSWO [34] and SWARAM. When a node's energy ran out throughout the simulation, that node counted as a dead node. The network lifetime, which was defined as the amount of time until the network's final node died, was intended to be measured by the simulation. The

quantity of alive nodes in the network indicates how many nodes are there. A large number of active nodes in the network improves network performance. The suggested methodology's alive node performance is evaluated using various node counts and existing algorithms. Figure 5b shows the comparison of alive nodes. The performance evaluation of packets transmitted by the proposed methodology with existing algorithms is presented in Figure 5c. Figure 5d illustrates how the total number of packets received by the base station (BS) is increased by reducing the energy consumption of the nodes during data packet transmission.

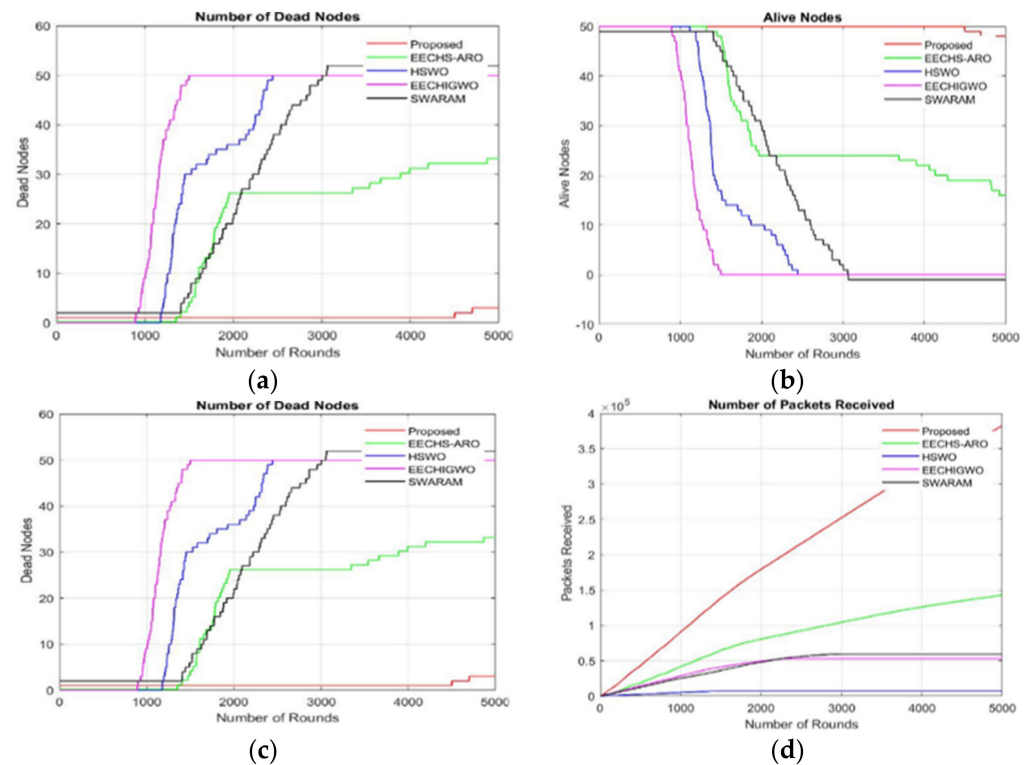


Figure 5. Comparison of (a) dead nodes, (b) alive nodes, (c) packets transmitted and (d) packets received in the network.

The simulation results for communication overhead are graphically shown in Figure 6a for those proposed with existing techniques like EECHS-ARO, HSWO, EECHIGWO, and SWARAM, and Table 2 provides evidence that the suggested method performs better than the existing methods. The existing method has attained a value of 0 for 50 nodes and 5, 14, 11.5, 10.5 and 7.5 for 300 nodes during simulation. For nodes 50, 100, 150, 200, 250, and 300 the proposed method attained 0, 0, 0, 1, 3, and 5, respectively. The proposed value is lower compared with the existing one. Thus, the proposed was better in communication overhead.

EC is shown graphically in Figure 6b and the tabular form is viewed in Table 3; it was found to be lower compared to existing approaches for EECHS-ARO, HSWO, EECHIGWO, and SWARAM with 0.17, 0.15, 0.13, 0.1 for 500 nodes, while 0.8, 0.1, 0.12, 0.15, 0.18, 0.2, 0.22 and 0.25 are the values of rounds 500, 1000, 1500, 2000, 2500, 3000, 3500, 4000 proposed, which is lower compared with existing approaches. Thus, the proposed approach was better for EC.

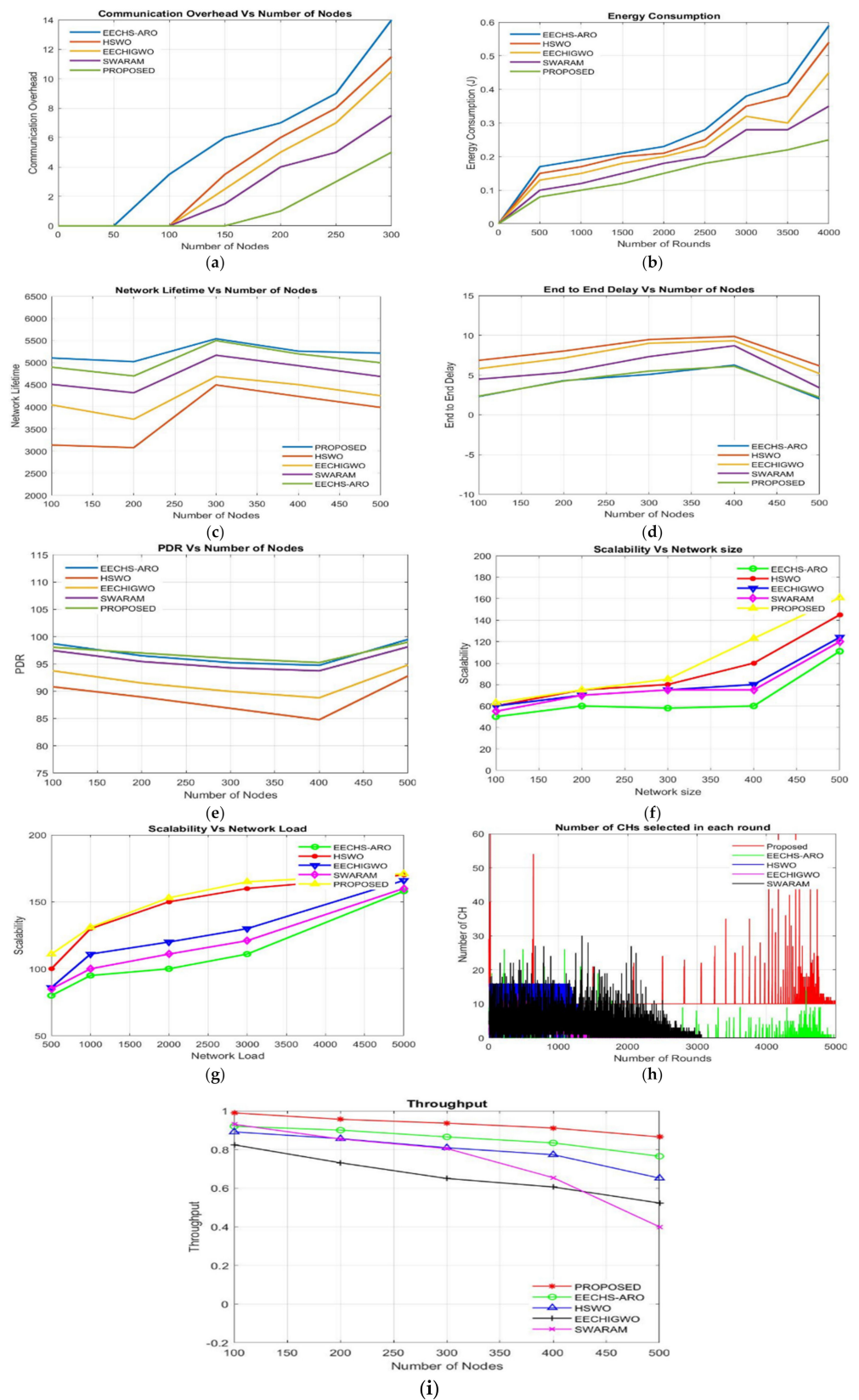


Figure 6. (a) Communication Overhead; (b) Energy Consumption; (c) Network Lifetime; (d) End—to—End Delay; (e) PDR; (f) Scalability vs. Network size; (g) Scalability vs. Network Load; (h) CH Selection; (i) Throughput.

Table 2. Comparative Analysis of Communication Overhead.

Methods	Communication Overhead					
	50	100	150	200	250	300
EECHS-ARO	0	3.5	6	7	9	14
HSWO	0	0	3.5	6	8	11.5
EECHIGWO	0	0	2.5	5	7	10.5
SWARAM	0	0	1.5	4	5	7.5
PROPOSED	0	0	0	1	3	5

Table 3. Comparative analysis of Energy Consumption (EC).

Methods	Energy Consumption (EC)							
	500	1000	1500	2000	2500	3000	3500	4000
EECHS-ARO	0.17	0.19	0.21	0.23	0.28	0.38	0.42	0.59
HSWO	0.15	0.17	0.2	0.21	0.25	0.35	0.38	0.54
EECHIGWO	0.13	0.15	0.18	0.2	0.23	0.32	0.3	0.45
SWARAM	0.1	0.12	0.15	0.18	0.2	0.28	0.28	0.35
PROPOSED	0.8	0.1	0.12	0.15	0.18	0.2	0.22	0.25

The network lifetime of the suggested strategy and the use of the previous models are shown in Figure 6c, where the x-axis presents the no. of nodes and the y-axis represents the network lifetime. This is viewed in Table 4: 4900, 3147, 4048, and 4512 are the values of EECHS-ARO, HSWO, EECHIGWO, and SWARAM, where the proposed approach attained 5107 for 100 nodes. The proposed approach attained 5024, 5542, 5261 and 5217 for 200, 300, 400 and 500 nodes, respectively. The proposed approach attained a higher network lifetime than the existing approaches.

Table 4. Comparative analysis of Network Lifetime.

Methods	Network Lifetime				
	100	200	300	400	500
EECHS-ARO	4900	4700	5500	5200	5000
HSWO	3147	3081	4498	4239	3990
EECHIGWO	4048	3723	4689	4505	4254
SWARAM	4512	4321	5169	4933	4689
PROPOSED	5107	5024	5542	5261	5217

The end-to-end delays of the proposed- and existing-scheme comparison are displayed in Figure 6d and in Table 5. The optimality factor is satisfied by the suggested scheme's constant attempt to deliver the packet to the best intermediate node. Conversely, existing methods require a significant amount of time to locate the destination node. Additionally, the destination node takes a while to respond, so the average end-to-end delay in the associated scheme is longer than that in the suggested model. The values 2.3, 6.85, 5.8, and 4.48 are those of EECHS-ARO, HSWO, EECHIGWO, and SWARAM and the proposed approach obtained 2.35ms for 100 nodes.

Table 5. Comparative analysis of end-to-end delay.

Methods	End-to-End Delay				
	100	200	300	400	500
EECHS-ARO	2.3	4.3	5.08	6.25	2.01
HSWO	6.85	8.02	9.47	9.86	6.15
EECHIGWO	5.8	7.13	9	9.31	5.18
SWARAM	4.48	5.33	7.32	8.69	3.39
PROPOSED	2.35	4.25	5.5	6.1	2.2

The PDR of the proposed approach is greater, with a value of 98.05. It is compared with existing techniques like EECHS-ARO, HSWO, EECHIGWO, and SWARAM, which obtained 98.72, 90.81, 93.74, and 97.45 for 100 nodes. The proposed approach obtained 97, 96, 95.25, 99 for 200, 300, 400, and 500 nodes, respectively. The PDRs are graphically shown in Figure 6e and the values are displayed in tabular form in Table 6.

Table 6. Comparative analysis of PDR.

Methods	PDR				
	100	200	300	400	500
EECHS-ARO	98.72	96.5	95.25	94.75	99.5
HSWO	90.81	88.95	86.87	84.78	92.81
EECHIGWO	93.74	91.5	89.96	88.8	94.81
SWARAM	97.45	95.44	94.28	93.74	98.14
PROPOSED	98.05	97	96	95.25	99

The scalability vs. network size of the proposed approach is lower, with a value of 24. It is compared with existing techniques like EECHS-ARO, HSWO, EECHIGWO, and SWARAM, which obtained 43, 50, 68, and 85 for 500 nodes. The values 24, 20, 19, and 15 are the proposed values for 1000, 2000, 3000, and 5000 nodes, respectively. The scalability vs. network size is graphically shown in Figure 6f and the values are displayed in Table 7.

Table 7. Comparative analysis of scalability with network size.

Methods	Scalability with Network Size				
	500	1000	2000	3000	5000
EECHS-ARO	43	42	41	24	20
HSWO	50	43	36	30	20
EECHIGWO	68	60	50	40	38
SWARAM	85	70	60	55	55
PROPOSED	24	24	20	19	15

The scalability vs. network load of the proposed approach is higher, with a value of 171. It is compared with existing techniques like EECHS-ARO, HSWO, EECHIGWO, and SWARAM, which obtained 170, 166, 160, and 158 for 5000 nodes. The scalability vs. network size is graphically shown in Figure 6g and the respective values are shown in Table 8. Figure 6h shows the comparison of the cluster head count produced by the existing approach and suggested in each iteration. The throughput graph is displayed in Figure 6i for the proposed and existing approaches.

Table 8. Comparative analysis of Scalability with Network Load.

Methods	Scalability with Network Load				
	500	1000	2000	3000	5000
EECHS-ARO	100	130	150	160	170
HSWO	86	111	120	130	166
EECHIGWO	85	100	111	121	160
SWARAM	80	95	100	111	158
PROPOSED	111	131	153	165	171

Table 9 shows the description of the packets delivered to the sink. For 100 nodes, the existing methods delivered 10,000, 16,000, 20,000 and 25,000 packets with EECHS-ARO, HSWO, EECHIGWO, and SWARAM, respectively, and the proposed method received 28,000 packets.

Table 9. Comparative analysis of packets delivered to sink.

Methods	Packets Delivered to Sink						
	100	250	500	1000	1250	2250	4000
EECHS-ARO	10,000	15,000	22,000	34,000	32,000	60,000	79,000
HSWO	16,000	25,000	30,000	45,000	58,000	80,000	130,000
EECHIGWO	20,000	30,000	35,000	54,000	61,000	105,000	147,000
SWARAM	25,000	35,000	40,000	64,000	70,000	110,000	157,000
PROPOSED	28,000	38,000	42,000	65,000	70,100	120,000	158,000

Table 10 shows the description of the scalability of density. For node 500, existing methods attained 60, 60, 55 and 50 for EECHS-ARO, HSWO, EECHIGWO, and SWARAM, and the proposed method obtained 63.

Table 10. Comparative analysis of scalability of density.

Methods	Scalability of Density				
	500	1000	2000	3000	5000
EECHS-ARO	60	75	80	100	145
HSWO	60	70	75	80	124
EECHIGWO	55	70	75	75	120
SWARAM	50	60	58	60	111
PROPOSED	63	75	85	123	161

Figure 7 shows average energy effect of the proposed and existing algorithms. Average energy effect is measured to show the superiority of the proposed model in energy consumption. Those proposed, CHOCR [35], IMD-EACBR [36], IDR-DRL-DC [37] and DTC-BR [38], obtained average energy effect of 90, 92, 95, 95 and 90 (%), respectively, thus proving that the proposed model obtained a better average energy effect.

Figure 8 shows the packet loss ratio compared to recent existing works. From the figure, it is clear that the proposed model attained better packet loss, which was due to security measures taken against cyber security attacks. Those proposed, CHOCR [35], IMD-EACBR [36], IDR-DRL-DC [37] and DTC-BR [38], obtained a loss ratio of 12, 15, 17, 21 and 22%, respectively. Thus, overall, the proposed model excels more than previous ones.

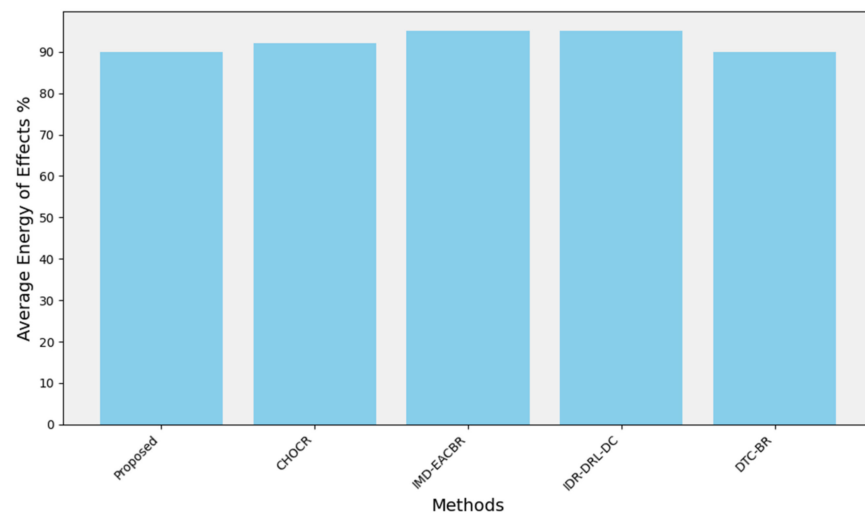


Figure 7. Average energy-effect comparison.

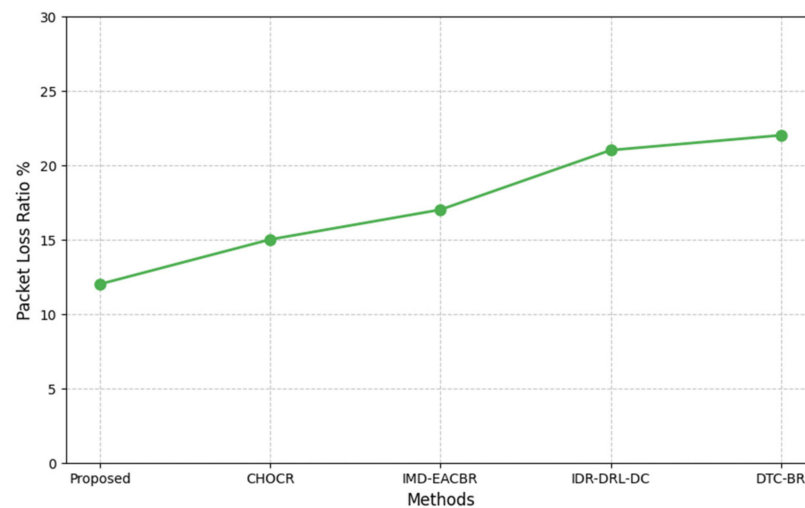


Figure 8. Packet loss-ratio comparison.

Figure 9 shows the robustness of the proposed EECM model. From the figure, the various attacks in the dataset are identified within the minimum time, which has been clearly mentioned. Attacks such as OS Scan, Active Wiretap, ARP MitM, Video Injection, Fuzzing, SSDP Flood, SYN DoS, SSL Renegotiation and Mirai have been focused on in this work. The proposed model obtained 98.05, 98.01, 98.06, 98, 98.01, 98.05, 98.05, 98.05 and 98.01 (%), respectively, where exiting models attained a very low percentage.

Figure 10 shows the computational complexity of the model in terms of time $O(N^2)$. From this, it has been proven that the proposed model attained a minimum time complexity of 18.9%, while others such as CHOQR [35], IMD-EACBR [36], IDR-DRL-DC [37] and DTC-BR [38] attained 22, 24, 22 and 23 (%), respectively. Thus, the proposed model was proven to be effective.

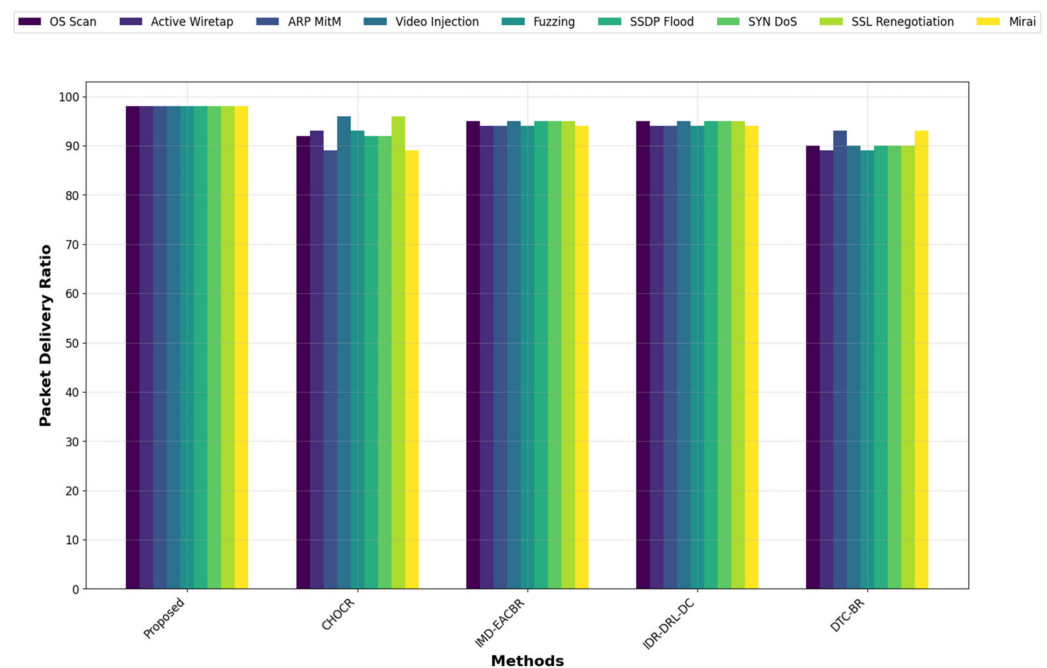


Figure 9. Packet delivery ratio for the comparison of various attacks.

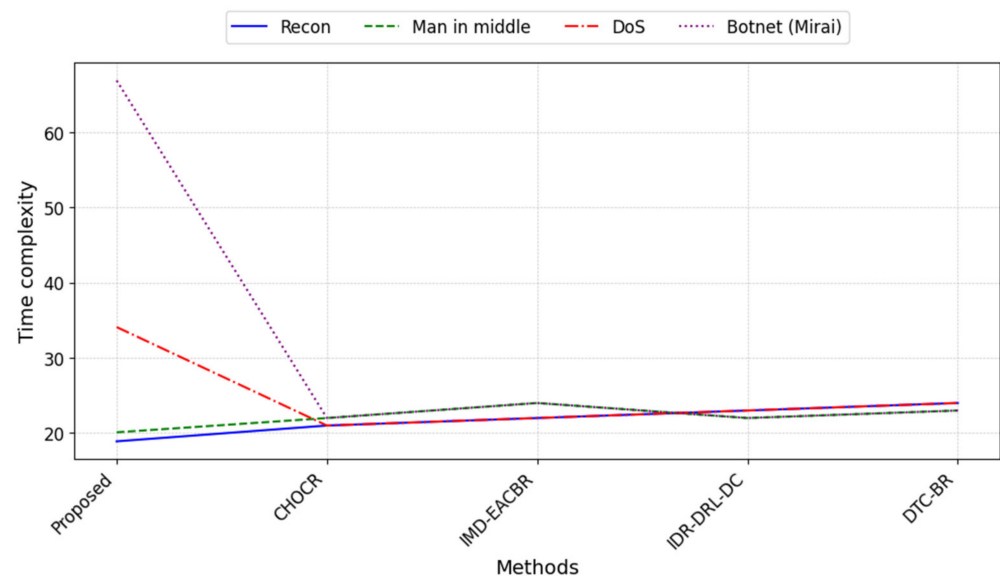


Figure 10. Time-complexity comparison graph.

4.3.2. Comparative Analysis of AlexNet with Existing Algorithm

Empirical findings are obtained from comparing machine learning models, according to accuracy. A range of assessment metrics employed to determine the effectiveness are shown in Figure 11. The percentage of correctly classified data instances relative to the total number of data instances is known as accuracy. A high accuracy of 97.45% is achieved by the CNN, and the 96.81% achieved by the RNN is not far behind. An accuracy of 95.33% is achieved by the ANN. The proposed has achieved 99.11%. From the analysis, CNN obtained 96.56%, KNN obtained 96.4%, RNN obtained 94.77% and the proposed model obtained 99%. Recall and F1 score metrics provide details on recall and precision while capturing all positive instances. With an F1 score of 97.55% and a recall of 97.44%, CNN performs admirably. KNN and RNN have comparable F1 scores and recall rates of approximately 95%. With a recall of 98.23% and an F1 score of 98.89%, the proposed notably beats the existing in both recall and precision.

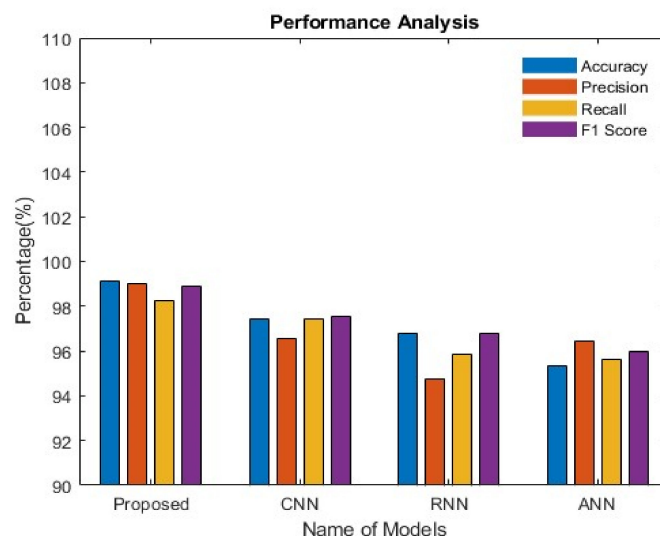


Figure 11. Performance analysis of accuracy, precision, recall and F1-score.

5. Discussion

In this study, energy consumption is the main goal of the proposed model, and security is also considered. A large number of active nodes in the network improves network performance. The suggested methodology's live node performance is evaluated using various node counts and existing algorithms. The existing communication overhead attained a value of 0 for 50 nodes and 5, 14, 11.5, 10.5, and 7.5 for 300 nodes during simulation. For nodes 50, 100, 150, 200, 250, and 300 the proposed methodology attain 0, 0, 0, 1, 3, and 5, respectively. Energy consumption for EECHS-ARO, HSWO, EECHIGWO, and SWARAM is 0.17, 0.15, 0.13, and 0.1 for 500 nodes; 0.8, 0.1, 0.12, 0.15, 0.18, 0.2, 0.22 and 0.25 are the values of rounds 500, 1000, 1500, 2000, 2500, 3000, 3500, and 4000 for the proposed model, which shows the supremacy of the proposed model. The average energy effect is measured to show the superiority of the proposed model for energy consumption. For the proposed CHOCR [35], IMD-EACBR [36], IDR-DRL-DC [37] and DTC-BR [38] obtained average energy effect of 90, 92, 95, 95 and 90 (%), respectively. Thus, the energy consumption of the model has been minimized effectively.

6. Conclusions

In the world of technology, H-IoT is thought to be the next big thing, bringing billions of objects and gadgets online for use and analysis. An innovative method known as EEMCM was proposed to address the scalability issues. It enhances the efficiency of anomaly detection in IoT WSNs by combining the MA with the AlexNet architecture. MA was first used for cluster formation and CH selection. Routing path creation was the next step, after which the data were ready for high-level feature extraction. The extracted features were then categorized to find anomalies. The AlexNet architecture gathered high-level features containing model-relevant data as input to identify anomalies and to distinguish between legitimate and possible attacks within the IoT WSNs. The accuracy, precision, recall, and F1-score of the proposed model is 99.11%, 99%, 98.23% and 98.89%. Overall, the EEMCM obtained superior results compared to existing algorithms. In future work, the performance of the EEMCM can be tested in a real-time environment by considering various other performance parameters such as mobility, security, fault tolerance, and load balancing. In addition, we can develop hybrid algorithms to choose the optimal CH.

Author Contributions: Both authors contributed to the conception of the problem setting and overall design of the work. This version was revised and improved by both authors, who also read and approved the final manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: No funding was received for conducting this study.

Institutional Review Board Statement: The research is original and all the figures and tables are created by the authors of this manuscript.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data sets are available in the Kaggle public repository and can be accessed publicly. The implementation of the proposed method is posted for public access in the github link <https://github.com/ashokttedcc/HT1.git> (accessed on 23 April 2024).

Conflicts of Interest: The authors declare that they have no conflict of interest.

References

1. Dao, N.-N.; Phan, T.V.; Sa'ad, U.; Kim, J.; Bauschert, T.; Do, D.-T.; Cho, S. Securing Heterogeneous IoT with Intelligent DDoS Attack Behavior Learning. *IEEE Syst. J.* **2021**, *16*, 1974–1983. [\[CrossRef\]](#)
2. Mukherjee, A.; Jain, D.K.; Yang, L. On-Demand Efficient Clustering for Next Generation IoT Applications: A Hybrid NN Approach. *IEEE Sensors J.* **2021**, *21*, 25457–25464. [\[CrossRef\]](#)
3. Singh, R.K.; Puluckul, P.P.; Berkvens, R.; Weyn, M. Energy consumption analysis of LPWAN technologies and lifetime estimation for IoT application. *Sensors* **2020**, *20*, 4794. [\[CrossRef\]](#)
4. Subramani, N.; Perumal, S.K.; Kallimani, J.S.; Ulaganathan, S.; Bhargava, S.; Meekanizi, S. Controlling energy aware clustering and multihop routing protocol for IoT assisted wireless sensor networks. *Concurr. Comput. Pract. Exp.* **2022**, *34*, e7106. [\[CrossRef\]](#)
5. Chaurasiya, S.K.; Mondal, S.; Biswas, A.; Nayyar, A.; Shah, M.A.; Banerjee, R. An Energy-Efficient Hybrid Clustering Technique (EEHCT) for IoT-Based Multilevel Heterogeneous Wireless Sensor Networks. *IEEE Access* **2023**, *11*, 25941–25958. [\[CrossRef\]](#)
6. Jaiswal, K.; Anand, V. A Grey-Wolf based Optimized Clustering approach to improve QoS in wireless sensor networks for IoT applications. *Peer Peer Netw. Appl.* **2021**, *14*, 1943–1962. [\[CrossRef\]](#)
7. Sahoo, L.; Sen, S.S.; Tiwary, K.; Moslem, S.; Senapati, T. Improvement of Wireless Sensor Network Lifetime via Intelligent Clustering Under Uncertainty. *IEEE Access* **2024**, *12*, 25018–25033. [\[CrossRef\]](#)
8. Elashry, S.S.; Abohamama, A.S.; Abdul-Kader, H.M.; Rashad, M.Z.; Ali, A.F. A Chaotic Reptile Search Algorithm for Energy Consumption Optimization in Wireless Sensor Networks. *IEEE Access* **2024**, *12*, 38999–39015. [\[CrossRef\]](#)
9. Bilal, M.; Munir, E.U.; Alarfaj, F.K. Hybrid Clustering and Routing Algorithm with Threshold-Based Data Collection for Heterogeneous Wireless Sensor Networks. *Sensors* **2022**, *22*, 5471. [\[CrossRef\]](#)
10. Hassan, A.A.-H.; Shah, W.M.; Habeb, A.-H.H.; Othman, M.F.I.; Al-Mhiquani, M.N. An Improved Energy-Efficient Clustering Protocol to Prolong the Lifetime of the WSN-Based IoT. *IEEE Access* **2020**, *8*, 200500–200517. [\[CrossRef\]](#)
11. Jubair, A.M.; Hassan, R.; Aman, A.H.M.; Sallehudin, H.; Al-Mekhlafi, Z.G.; Mohammed, B.A.; Alsaffar, M.S. Optimization of clustering in Wireless sensor networks: Techniques and protocols. *Appl. Sci.* **2021**, *11*, 11448. [\[CrossRef\]](#)
12. Yan, X.; Huang, C.; Gan, J.; Wu, X. Game Theory-Based Energy-Efficient Clustering Algorithm for Wireless Sensor Networks. *Sensors* **2022**, *22*, 478. [\[CrossRef\]](#) [\[PubMed\]](#)
13. Sharma, R.; Vashisht, V.; Singh, U. Metaheuristics-based energy efficient clustering in WSNs: Challenges and research contributions. *IET Wirel. Sens. Syst.* **2020**, *10*, 253–264. [\[CrossRef\]](#)
14. Khalaf, O.I.; Romero, C.A.T.; Hassan, S.; Iqbal, M.T. Mitigating Hotspot Issues in Heterogeneous Wireless Sensor Networks. *J. Sensors* **2022**, *2022*, 7909472. [\[CrossRef\]](#)
15. Somula, R.; Cho, Y.; Mohanta, B.K. SWARAM: Osprey Optimization Algorithm-Based Energy-Efficient Cluster Head Selection for Wireless Sensor Network-Based Internet of Things. *Sensors* **2024**, *24*, 521. [\[CrossRef\]](#) [\[PubMed\]](#)
16. Krichen, M. A Survey on Formal Verification and Validation Techniques for Internet of Things. *Appl. Sci.* **2023**, *13*, 8122. [\[CrossRef\]](#)
17. Hofer-Schmitz, K.; Stojanović, B. Towards formal verification of IoT protocols: A Review. *Comput. Networks* **2020**, *174*, 107233. [\[CrossRef\]](#)
18. Rizvi, S.; Orr, R.; Cox, A.; Ashokkumar, P.; Rizvi, M.R. Identifying the attack surface for IoT network. *Internet Things* **2020**, *9*, 100162. [\[CrossRef\]](#)
19. Ionnou, C. Vasos Vassiliou IEEE Computer Society, Institute of Electrical and Electronics Engineers. In Proceedings of the 15th Annual International Conference on Distributed Computing in Sensor Systems, Santorini Island, Greece, 29–31 May 2019.
20. Almurisi, N.; Tadisetty, S. Cloud-based virtualization environment for IoT-based WSN: Solutions, approaches and challenges. *J. Ambient. Intell. Humaniz Comput.* **2022**, *13*, 4681–4703. [\[CrossRef\]](#)
21. Shukla, A.; Tripathi, S. A multi-tier based clustering framework for scalable and energy efficient WSN-assisted IoT network. *Wirel. Networks* **2020**, *26*, 3471–3493. [\[CrossRef\]](#)

22. Rami Reddy, M.; Ravi Chandra, M.L.; Venkatramana, P.; Dilli, R. Energy-Efficient Cluster Head Selection in Wireless Sensor Networks Using an Improved Grey Wolf Optimization Algorithm. *Computers* **2023**, *12*, 35. [\[CrossRef\]](#)
23. Alghamdi, T.A. Energy efficient protocol in wireless sensor network: Optimized cluster head selection model. *Telecommun. Syst.* **2020**, *74*, 331–345. [\[CrossRef\]](#)
24. Raslan, A.F.; Ali, A.F.; Darwish, A.; El-Sherbiny, H.M. An Improved Sunflower Optimization Algorithm for Cluster Head Selection in the Internet of Things. *IEEE Access* **2021**, *9*, 156171–156186. [\[CrossRef\]](#)
25. Pour, S.E.; Javidan, R. A new energy aware cluster head selection for leach in wireless sensor networks. *IET Wirel. Sens. Syst.* **2021**, *11*, 45–53. [\[CrossRef\]](#)
26. Poluru, R.K.; Ramasamy, L.K. Optimal cluster head selection using modified rider assisted clustering for IoT. *IET Commun.* **2020**, *14*, 2189–2201. [\[CrossRef\]](#)
27. Sankar, S.; Ramasubbareddy, S.; Dhanaraj, R.K.; Balusamy, B.; Gupta, P.; Ibrahim, W.; Verma, R. Cluster Head Selection for the Internet of Things Using a Sandpiper Opti-mization Algorithm (SOA). *J. Sens.* **2023**, *2023*, 3507600. [\[CrossRef\]](#)
28. Bakshi, M.; Chowdhury, C.; Maulik, U. Energy-efficient cluster head selection algorithm for IoT using modified glow-worm swarm optimization. *J. Supercomput.* **2021**, *77*, 6457–6475. [\[CrossRef\]](#)
29. Ahmad, M.; Shah, B.; Ullah, A.; Moreira, F.; Alfandi, O.; Ali, G.; Hameed, A. Optimal Clustering in Wireless Sensor Networks for the Internet of Things Based on Memetic Algorithm: MemeWSN. *Wirel. Commun. Mob. Comput.* **2021**, *2021*, 8875950. [\[CrossRef\]](#)
30. Dwivedi, A.K.; Mehra, P.S.; Pal, O.; Doja, M.N.; Alam, B. EETSP: Energy-efficient two-stage routing protocol for wireless sensor network-assisted Internet of Things. *Int. J. Commun. Syst.* **2021**, *34*, e4965. [\[CrossRef\]](#)
31. Kala, I.; Karthik, S.; Srihari, K. Advanced hybrid secure multipath optimized routing in Internet of Things (IoT)-based WSN. *Int. J. Commun. Syst.* **2021**, *34*, e4782. [\[CrossRef\]](#)
32. Yarinezhad, R.; Sabaei, M. An optimal cluster-based routing algorithm for lifetime maximization of Internet of Things. *J. Parallel Distrib. Comput.* **2021**, *156*, 7–24. [\[CrossRef\]](#)
33. Ramalingam, R.B.S.; Basheer, S.; Balasubramanian, P.; Rashid, M.; Jayaraman, G. EECHS-ARO: Energy-efficient cluster head selection mechanism for livestock industry using artificial rabbits optimization and wireless sensor networks. *Electron. Res. Arch.* **2023**, *31*, 3123–3144. [\[CrossRef\]](#)
34. Samiayya, D.; Radhika, S.; Chandrasekar, A. An optimal model for enhancing network lifetime and cluster head selection using hybrid snake whale optimization. *Peer-to-Peer Netw. Appl.* **2023**, *16*, 1959–1974. [\[CrossRef\]](#)
35. Lakshmana, K.; Subramani, N.; Alotaibi, Y.; Alghamdi, S.; Khalafand, O.I.; Nanda, A.K. Improved Metaheuristic-Driven Energy-Aware Cluster-Based Routing Scheme for IoT-Assisted Wireless Sensor Networks. *Sustainability* **2022**, *14*, 7712. [\[CrossRef\]](#)
36. Srivastava, A.; Paulus, R. Coverage hole aware optimal cluster based routing for wireless sensor network assisted IoT using hybrid deep recurrent neural network. *Concurr. Comput. Pract. Exp.* **2022**, *35*, 1. [\[CrossRef\]](#)
37. Sebastin, S.S.; Prabhu, V.; Parthasarathy, V. An intelligent data routing strategy based on deep reinforcement learning for IoT enabled WSNs. *E-Prime-Adv. Electr. Eng. Electron. Energy* **2023**, *6*, 100298. [\[CrossRef\]](#)
38. Al-Sadoon, M.E.; Jedidi, A.; Al-Raweshidy, H. Dual-Tier Cluster-Based Routing in Mobile Wireless Sensor Network for IoT Application. *IEEE Access* **2023**, *11*, 4079–4094. [\[CrossRef\]](#)

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.