Journal of Intelligent Systems and Internet of Things
Vol. 17, No. 01, PP. 304-324, 2025
DOI: https://doi.org/10.54216/JISIoT.170122

Hybrid Optimization based Clustering with CNN-Based De-Authentication for IoT Enabled Heterogeneous Wireless Sensor Networks

Foad Salem Mubarek[1], Akeel A.Thulnoon[2,*], Ahmed Mahdi Jubair[1]

[1]Department of Computer Networks Systems, College of Computer Science and Information Technology, University of Anbar, Ramadi 31001, Anbar, Iraq

[2]Department of Information Technology, College of Computer Science and Information Technology, University of Anbar, Ramadi 31001, Anbar, Iraq

Emails: co.foad.salem@uoanbar.edu.iq; akeelalhadithy@uoanbar.edu.iq; ahmed.mahdi@uoanbar.edu.iq

Abstract

The Internet of Things (IoT) has greatly changed many aspects of human life and is now a vast distributed systems network of interconnected devices that have embedded sensors; however, the battery life of these sensor nodes is limited and requires constant maintenance. Furthermore, IoT networks operating as distributed systems are vulnerable to security threats, like de-authentication and Disassociation Denial-of-Service attacks, which exploit vulnerabilities in Wi-Fi devices. While artificial intelligence, including machine learning, has been integrated into intrusion detection systems to enhance detection of cyberattacks, there is an increasing need for improved accuracy, scalability, efficiency, and IoT-specific security solutions. This paper proposed a novel model, Hybrid Optimization-based Clustering with CNN-Based De-Authentication (HOCCNN), designed to concurrently address both energy conservation and security issues in IoT-enabled heterogeneous wireless sensor networks (WSNs). The HOCCNN adopts a hierarchical clustering technique optimized using the bio-inspired Osprey Optimization Algorithm (OOA) for dynamic and energy-efficient Cluster Head (CH) selection. Additionally, we introduce a CNN model to detect and mitigate De-authentication attacks in HOCCNN by utilizing deep learning techniques and provide a more accurate threat detection solution even in the resource-constrained environment. The performance of HOCCNN was evaluated using MATLAB against existing baseline methods in terms of parameters like packet delivery ratio, network throughput, network lifetime, end-to-end delay, average energy consumption, data accuracy, and data overhead. The model demonstrates superiority over state-of-the-art baselines. Results show significant improvements. 99.1% accuracy in attack detection, 54.18% energy consumption, 6.76 s network lifetime, 0.985 packet delivery ratio, and 53.198 Mb/s throughput. These results prove that HOCCNN is a complete design to achieve scalable, secure, and energy-sustainable HWSNs in IoT.

Received: January 12, 2025 Revised: March 05, 2025 Accepted: April 06, 2025

Keywords: Internet of Things (IoT); Heterogeneous Wireless Sensor Networks; Hybrid Optimization; CNN-Based De-Authentication; Osprey Optimization Algorithm

1. Introduction

Wireless sensor networks (WSNs) have emerged as one of process of acquiring technology and transmitting various types of environmental data in specific monitoring regions [1]. Beyond simple data collection, WSNs allow intelligent data processing and support automated decision-making processes [2]. A WSN generally consists of a large number of sensor nodes which locally conduct message processing and collectively gather and transmit data to a central data collection node called as the sink, The sink is responsible for aggregating, processing, and transmitting this data to external servers for further use [3]. These networks are valued for their ease of deployment, strong reliability, and energy efficiency, security, which has led to their widespread application in sectors such as environmental surveillance, healthcare, industrial monitoring [4].

WSNs constitute a foundational component of the Internet of Things (IoT). In IoT-enabled WSNs, sensor nodes continuously monitor large scale environment with conditions and report events to a central base station (BS) in a distributed system architecture [5]. Furthermore, WSNs has two primary challenges security and energy consumption, which are both inherently interrelated. Enhancing network security often led to increases energy consumption, and vice versa [6]. The requirement of both reducing security and power consumption is certainly a challenge since the situations in which these sensors can be utilized are extremely challenging but this is the challenge which is being catered by the recent research works in this domain [7]. A set of traditional security models, based on the Confidentiality, Integrity and Authentication, are increasingly being revaluated for their suitability. The best-known encryption protocols, key exchange, cryptographic operations between nodes, are also considered as energy-consuming and inappropriate for real world requirements [8].

The classical energy-saving routing schemes, including green routes, often employ hierarchical clustering and special controller nodes to manage data flow and extend network lifetime [9]. One such application is the low-energy adaptive clustering hierarchy (LEACH) protocol [10], which clusters the sensor nodes and allows the local data aggregation to be managed by the reduced energy cost cluster heads. Separate data routing from transmission with the assistance of auxiliary nodes to avoid frequent relaying-induced early node failures. However, these schemes are not without issues. They are significantly dependent on accurate mathematical schemes that are often impossible to establish and are energy-intensive in terms of computations. Moreover, they are often incapable of adapting to various topologies and scales of the network and are prone to congestion and reduced operating efficiency [11]. These drawbacks make more flexible and light schemes more appealing given the highly dynamic nature of the WSN topologies. Hence, alternative ways that are faster and simpler are what are required. for instance, artificial intelligence algorithms are one of the ways that can be utilized for this task. Skills can be developed in a node to communicate with local WSN nodes, identify viruses, scrutinize incoming and outgoing packets, authenticate nodes, and remain available [12].

Machine learning (ML) is one of artificial intelligence major areas, builds predictive models from training data without explicit programs [13]. This works particularly well for WSNs, since uncertainty and dynamics in the environment usually make traditional mathematical modelling problematic. Additionally, the ability to function with minimal human intervention makes ML particularly suitable to decentralized WSN distributed systems architectures [14]. however, there are issues in using ML in WSNs. Sensor nodes' restricted computation and memory make the ML programs complexity difficult to scale, and significant training data are often required to learn and function effectively [15]. From a security perspective, utilizing ML to attain confidentiality and integrity requirements remains problematic [16]. however, there are possible strengths in ML-based programs in congestion control, physical-layer authentication, and error detection [17]. Their capacity to analyse packet dynamics to detect anomalous nodes also make their deployment in making WSNs more secure and robust justified [18].

Despite the increasing significance in ML within the WSN environment, this study highlights the critical role of clustering in enhancing energy efficiency and extending network lifetime [19]. It highlights through the persistent challenge of energy depletion caused from suboptimal Cluster Head selection strategies [20]. Additionally, it addresses the increasing security threats in Internet of Things networks, particularly vulnerabilities to de-authentication and disassociation denial-of-service attacks that exploit weaknesses in Wi-Fi protocols [21]. While ML techniques including deep learning have effective potential in detecting cyberattacks, there remains a pressing gap to optimize their accuracy and computational efficiency in resource-constrained IoT device. To tackle these Grand challenges, in this paper we propose a novel Hybrid Optimization-based Clustering with CNN-Based De-Authentication (HOCCNN), approach aimed at jointly optimizing energy consumption and enhancing security in IoT-enabled heterogeneous WSNs. The main contributions of this article summarized as follows:

- HOCCNN is a novel system model that addresses the energy consumption issues faced by heterogeneous WSNs and offers an effective approach to solving these problems. By considering the energy levels, processing capabilities, and communication requirements of nodes, the HOCCNN model differs from traditional homogeneous models. The selection process for cluster heads in this model is dynamic and optimized, which considers the unique characteristics and mobility of heterogeneous sensors, resulting in increased network robustness and efficiency.
- Optimizing Traffic Loads is critical to maximizing network lifetime and performance. The HOCCNN model uses a sector-specific traffic estimation method that accurately estimates the data load in different areas of the network. The model utilizes dynamic adjustment of the transmission areas after calculating the average traffic load to ensure optimal data routing. A sector-specific approach, along with an intelligent algorithm for calculating traffic loads, allows data packets to be efficiently managed and reduces network congestion.

- Utilizing the natural hunting and feeding behaviour of ospreys, the research develops an 'odor-optimized' OOA to optimize cluster heads using hybrid optimization techniques. The selection process is enhanced by a biologically inspired algorithm that considers both residual energy and the Euclidean distance from the base station. The algorithm's goal is to select cluster heads with optimal energy efficiency by periodically updating the sensor node locations based on their fitness function values.
- The HOCCNN model incorporates multi-hop communication strategies to address scalability concerns in large sensor networks. By selecting intermediate cluster heads based on their RSSI and base station distance, the model ensures efficient data transmission between multiple hops. Energy conservation is achieved by reducing the transmission power needed for long-distance communication, which also enhances network flexibility.

The organization of the article is given below. In section 2, the earlier models are elaborated, and their flaws and identified. In section 3 the Preliminaries model is elaborated. In section 4 the HOCCNNs proposed model is explained in detail. Followed by that in section 5, extensive simulation is carried out and results are analysed. In section 6 the paper is concluded with its future direction of research.

2. Related Work

As highlighted. In [22], the growing spectrum of IoT applications drives the need for better energy performance, with LPWAN technologies like LoRaWAN, DASH7, Sigfox, and NB-IoT enabling long-range, low-cost connectivity. This paper empirically evaluates the energy consumption of these LPWAN technologies, finding LoRaWAN and DASH7 more energy-efficient than Sigfox and NB-IoT. A precision agriculture case study reveals significant drops in battery lifetime in real applications, aiding in selecting the right technology and battery capacity for effective IoT deployment. In [23], the integration of IoT and Wireless Body Area Networks (WBANs) enhances healthcare by enabling efficient remote monitoring. This paper proposes the Energy Harvested and Cooperative-enabled Efficient Routing Protocol (EHCRP) for IoT-WBANs, optimizing routing using parameters like residual energy, hop count, congestion, SNR, and bandwidth. Extensive simulations show that EHCRP significantly improves network lifetime, throughput, and end-to-end delay compared to existing protocols. In [24], in WSNs are vital for IoT and Industry 4.0, but existing approaches often overlook robustness against cascading failures between physical and information domains. This paper introduces a cross-domain agent-based model to analyze system connectivity robustness during malware propagation, detailing agent characteristics and transition rules. Three network topology scenarios are used to verify the model's practicality, discussing their robustness.

In [25], the EH-WSNs enhance IoT by alleviating energy limitations but often overlook energy state and data buffer constraints. This paper proposes a new greedy strategy-based energy-efficient routing protocol, combining an energy evaluation model and a communication range judgment model, to optimize data reception and routing. Simulations demonstrate that this algorithm significantly improves energy consumption, packet delivery ratio, average hop count, and end-to-end delay while maintaining acceptable energy variance. In [26], rapid deployment of WSNs and IoT has expanded their industrial applications, with improvements in Medium Access Control (MAC) protocols being crucial. This work reviews recent WSN MAC protocols, highlighting methods to enhance performance factors like energy consumption, scalability, and clustering intelligence. A comparison table details how these approaches improve network throughput, end-to-end delay, packet drop, and energy consumption. In [27], a hybrid clustering and routing algorithm with threshold-based data collection for heterogeneous WSNs reduces unnecessary transmissions and enhances network stability. The model outperforms other threshold-based energy-efficient protocols like TSEP, TDEEC, LEACH, and TEEN in load balancing and end-to-end delay.

In [28], is essential for short-range, low-rate IoT communications, but enhancing performance requires efficient MAC and routing protocols. This paper proposes a cross-layer protocol combining MAC and routing layers to improve scalability, reliability, and address allocation in mesh networks. Simulations show this approach enhances reliability and reduces end-to-end delay. In [29], enables smart cities and services, with Wireless Sensor Networks (WSNs) crucial for timely data collection but vulnerable to jamming attacks. This work proposes a clustering-based self-healing strategy, Fairness Cooperation with Power Allocation (FCPA), to detect and counteract jamming by clustering nodes, adjusting transmit power, and extending network life through balanced relay usage. Experiments show FCPA significantly outperforms benchmarks, enhancing information transmission and energy efficiency by more than 50%. In [30], researchers introduced TESEES, a zone-based, event-driven protocol designed to enhance energy use in large-scale heterogeneous WSNs. It uses a dynamic thresholding model that adjusts data reporting based on actual events, which helps reduce unnecessary transmissions. As a result, the protocol lowers energy consumption and extends the network's lifespan. Simulations show that TESEES significantly outperforms traditional protocols like SEES in terms of energy efficiency, network longevity, and reduced data volume.

In [31], authors introduced a multi-objective clustering method for energy-efficient wireless sensor networks in precision agriculture. The technique combines the Election-based Aquila Optimizer (EAO) with a CNN to enhance cluster head selection and improve clustering accuracy. This hybrid model outperforms existing approaches across several important metrics, including classification accuracy, throughput, packet delivery ratio, network lifetime, and energy efficiency. In [32], a hybrid WSN-LTE architecture to optimize energy use and enhance QoS in IoT applications. It addresses the challenge of maximizing network lifespan while maintaining multi-factor QoS through two-stage optimization involving clustering and data collection path scheduling. The hybrid approach combines meta-heuristics for clustering and graph theory for path scheduling, resulting in a significant increase in network lifespan and a reduction in packet delivery delay compared to existing methods. Table 1, describes the earlier research summary in detail.

Table 1: Earlier Research Summary

REF. NO	**Algorithm**	**Contributions**	**Limitations**
[22]	Evaluation of LPWAN Technologies (LoRaWAN, DASH7, Sigfox, NB-IoT)	Conducted an empirical evaluation comparing the energy efficiency of major LPWAN technologies.	Empirical comparison; doesn't propose a new protocol or algorithm.
[23]	Energy Harvested and Cooperative-enabled Efficient Routing Protocol (EHCRP) for IoT-WBANs	Proposed the EHCRP routing protocol for IoT-WBANs, incorporating multiple metrics (residual energy, hop count, etc.).	Specific to WBANs; complexity of considering multiple metrics.
[24]	Cross-Domain Agent-Based Model	Developed a cross-domain agent-based model to analyze cascading failures and robustness under malware propagation.	Analytical/modeling approach; simulation-based validation in specific topologies.
[25]	Greedy Strategy-based Energy-Efficient Routing (for EH-WSNs)	Proposed a greedy strategy-based routing protocol for EH-WSNs, incorporating energy and communication range models.	Greedy approach might lead to local optima; simulation-based validation.
[26]	Review of WSN MAC Protocols	Provided a review and comparison of recent WSN MAC protocols and methods for performance enhancement.	Review paper; doesn't propose a novel protocol.
[27]	Hybrid Clustering and Routing Algorithm (Threshold-based)	Introduced a hybrid clustering and routing algorithm using threshold-based data collection for heterogeneous WSNs.	Requires threshold setting; performance depends on threshold accuracy.
[28]	Cross-Layer Protocol (MAC/Routing for IEEE 802.15.4)	Proposed a cross-layer protocol integrating MAC and routing functions for IEEE 802.15.4 mesh networks.	Cross-layer complexity; integration challenges.
[29]	Clustering-Based Self-Healing Strategy (FCPA)	Introduced the Fairness Cooperation with Power Allocation (FCPA) strategy for jamming detection and self-healing.	Specific to jamming attack mitigation; relies on cooperative relaying.

[30]	TESEES Protocol (Zone-based probability)	Proposed the TESEES protocol, a zone-based probability protocol with a novel thresholding model for energy saving.	Zone-based approach; potential overhead in zone management.
[31]	EAO with O-CNN (Election-based Aquila Optimizer & Optimized CNN)	Combined an Election-based Aquila Optimizer with a CNN for multi-objective clustering in WSNs.	Combines metaheuristic and ML; computational complexity.
[32]	Hybrid WSN-LTE Architecture (Meta-heuristics + Steiner tree)	Proposed a hybrid WSN-LTE architecture using meta-heuristics for clustering and Steiner tree for path scheduling.	Hybrid architecture complexity; dependency on LTE infrastructure.

3. Preliminaries

Convolutional neural networks (CNN) are highly effective in image analysis tasks, including face recognition and object recognition. It uses layers like convolution and pooling to extract and preserve important features while avoiding overfitting. In TL, pre-trained models are applied to new tasks on smaller datasets, based on existing knowledge. Using models like Alex Net or Res Net, TL enables patterns in large datasets to be used to solve new challenges without extensive data collection. CNNs and TL are potent tools for both image-related and multidomain applications.

3.1 Convolutional Neural Networks definition

Convolutional Neural Networks (CNNs) are a special type of deep neural networks that have shown extraordinary ability to excel at computer vision-related tasks. The capabilities of CNNs extend to a very wide array of applications, such as face recognition, object detection, image classification, image restoration, image captioning, industrial automation, and even audio recognition. Due his ability the CNNs are like basic tools in contemporary image analysis is that they can automatically extract spatial and hierarchical features from unstructured input data. A basic architecture of CNN's architecture illustrated by Figure 1, consists of various important layers: convolutional layers, pooling (or subsampling) layers, activation (usually ReLU) layers, and fully connected layers. The convolutional layers are central feature extractors that make use of learnable filters (kernels) to create feature maps that capture important spatial patterns from within the input. The maps are then down sampled by using pooling layers that reduce dimensionality, suppress overfitting, and promote efficiency computationally by maintaining only significant features.

Rectification layers add non-linearity to the model, such that it can learn rich relationships within data. The concluding step usually involves flattening feature maps and presenting them to fully connected layers, which do high-level reasoning as well as classification. With regards to conventional machine learning models, CNNs are superior in dealing with large and complex datasets, with less tendency to overfit via localized receptive fields as well as shared weights. Architectural efficiency further allows fast pattern detection as well as classification, such as anomalies or prospective security breaches within sensitive sectors.

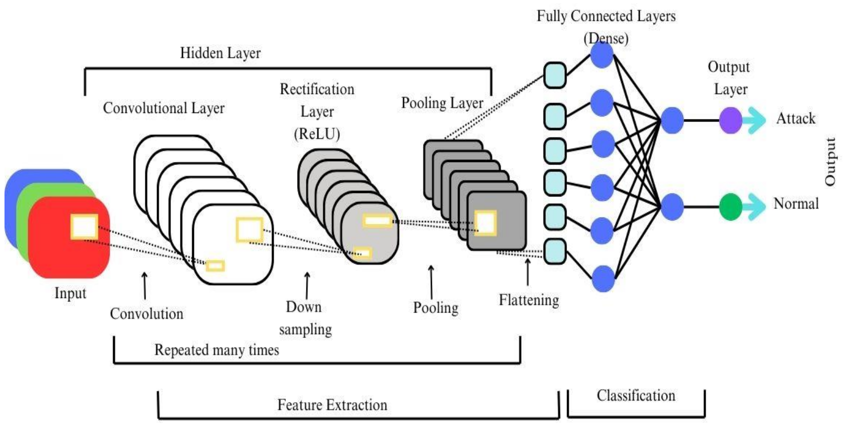

Figure 1. The fundamental structure of a CNN [21].

3.2 Transfer learning definition

Transfer learning (TL) is one of the powerful widely adopted technique that efficiently allows utilization of deep neural network for tasks involving limited datasets. The fundamentals of TL is to exploit structure and learned weights from models trained on large-scale datasets (AlexNet, VGGNet and ResNet) and adapting them to new, often smaller target tasks. As demonstrated in Figure 2, transfer learning adopts the pretrained CNN to guide the knowledge transferred from a source domain to a target domain. By initializing the new task from weights learned from large datasets, such as ImageNet, which includes millions of labelled images of categories and TL avoids training new models from scratch. This not only speeds up learning. However, also leads to better generalization especially when labelled data is limited. Instead of designing a model from scratch, transfer learning enables researchers to build upon established feature representations that are trained on large-scale datasets, leading to a much more efficient approach in solving new and complicated tasks.

4 Proposed HOCCNN Model

A hierarchical and hybrid optimization-based clustering model for heterogeneous WSNs is proposed in the HOCCNN model. Using the new collector node (CH) system, it effectively manages data aggregation while prioritizing energy conservation and network lifetime enhancement. The model uses the OOA to optimally select the cluster head and implements multi-hop data transmission to improve network scalability and efficiency. Intelligent node selection and efficient traffic management strategies further optimize network energy consumption and routing.

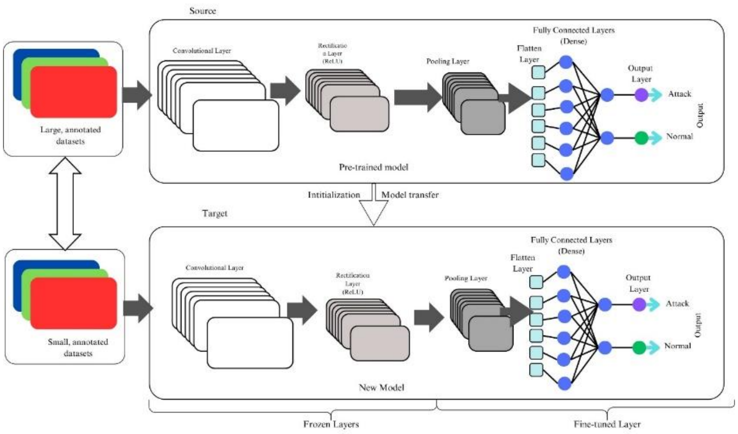

Figure 2. The employment of transfer learning and convolutional neural networks [21].

4.1 System Model

to maintain maximum efficiency within heterogeneous systems, there is a need to adopt an efficient data collection and aggregation scheme specific to hybrid network environments. Such a scheme can make use of hierarchical energy-efficient routing schemes. Unlike previous versions that assumed equal resource provisioning to all sensor nodes, this new system model involves a more realistic approach. The new system employs a cluster head (CH) to aggregate data from member nodes and account for node failures or node energy depletions as well. In this new model, the energy at every node diminishes at a time-varying rate primarily as a function of changes of transmission as well as reception distances. The obtained dynamic leads to elevated energy usage as well as an accompanying overall network lifetime reduction. It considered an initial deployment, where all the nodes have the same energy with E = 100µJoules. The cluster-based routing architecture is further described in Fig. 3 and the communication scheme includes one-hop, two-hop and multi-hop.

After a certain period, t = 1, the state of the network and the energy levels of each node are dependent on their usage. Sensor nodes only transmit their sensing data and consume less energy than other devices that also transmit data from multiple nodes like N8 and N4. At t = 2, the energy levels of each node are observed. From the graph, N4 loses its battery more frequently than other devices due to its high load in the network, while others still have sufficient residual energy. Therefore, it is important to prioritize nodes with enough residual energy for network reconstruction. The main issue is selecting a CH (cluster head) with the highest remaining energy to prolong network life and support. This involves restructuring the network so that the CH carries the maximum number of neighbouring nodes and has enough residual energy for sustained support. The figure 3 shown the clustering architecture diagram.

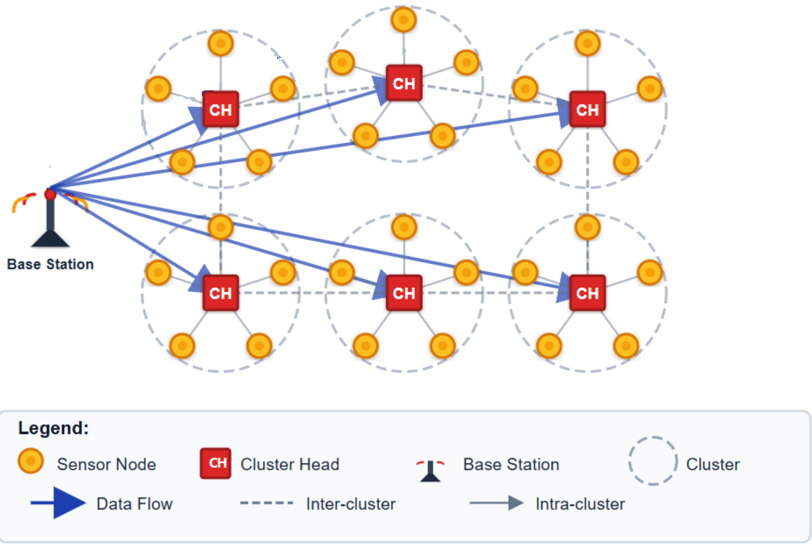

Figure 3. Cluster-based routing involves one-hop, two-hop, and multi-hop routing.

4.2 Energy Model

We define the lifespan of the network as the duration from when all device batteries B_i are fully charged when any of them is completely drained. Our focus is on preventing jamming attacks, so we assume that at the start of the attack, all devices had equal energy levels equivalent to their battery capacity $B_i = B_C \; \forall \, i \in \{1, 2, \cdots, N\}$. IoT nodes use energy for tasks such as collecting and processing information, receiving, and sending data, and maintaining schedules and synchronization. However, to Comparison of different transmission control and cooperative communications strategies All the non-communication-related functions are ignored. The energy for each process is calculated based on its duration and the components included. In particular, the energy consumption of node i in transmission e_{t_i} can be estimated using several important parameters such as the transmit power (P_i), the power amplifier efficiency (η),, the operating power of the transmission circuitry (P_{ct}) and the transmission time. The second depends on both the message length (L) and the transmission rate (R). The energy consumption of transmission can be represented in terms of these parameter to make the energy efficiency of communication strategies in different architecture evaluated and compared accurately in our approach.

$$e_{t_i} = \frac{(\frac{P_i}{\eta} + P_{ct})L}{R} \tag{1}$$

$$e_{t_i} = \frac{P_{cr}L}{R} \tag{2}$$

Likewise, the energy usage for the receiving process is determined by (P_{ct}), which represents the energy consumed by the receiving circuitry. As a result, node i energy consumption after transmitting t_i messages can be calculated. The implementation of the suggested self-healing methods described in the following section involves certain nodes, known as cluster heads, acting as information sinks, causing their energy consumption to rely on the number of messages received r_i. Additionally, some of these strategies involve cooperative communication between nodes, resulting in their energy usage being dependent on the number of messages they relay (c_i) by receiving and retransmitting information. Therefore, The residual energy of node i is affected by a number of factors, such as the selected self-healing strategy, the functional position of the node in its cluster and its communication activity as data sink, it receives more than one message; as relay node, it forwards and cooperates in transmission. When Transmit Power Control is used, the energy expended in forwarding information is distance-sensitive. As a result, any variation of the next-hop relay nodes will reflect the transmission energy cost directly because the longer distance usually requires higher transmission power. This reinforces the necessity for adaptive energy-aware communication policies that take into account topology changes as well as the varying energy consumption of selecting relays.

$$E_{T_i} = \sum_{j=1}^{t_i} e_{ti} = t_i e_{t_i} \tag{3}$$

$$E_{R_i} = \sum_{j=1}^{r_i} e_{ri} = r_i e_{ri} \tag{4}$$

$$E_{C_i} = \sum_{j=1}^{c_i} (e_{r_i + e_{t_i}) = c_i (e_{r_i} + e_{t_i})} \tag{5}$$

4.1 Traffic Model

Effective traffic control is crucial for increasing the lifetime of a WSN. To facilitate this, the network is divided into smaller, manageable sub-regions, known as sectors S. This sector-wise division would improve the estimation of the data load per area by considering the node density, which is represented by ρ. It is known that downstream nodes suffer higher communication cost than upstream nodes, due to the fact that upstream nodes, usually, forward their own generated data to the sink. Using this information, it is possible to estimate the amount of data packets originated in a region. For instance, consider S_n as an n sector at a distance d from sink. The resultant total traffic of sector S_n can be obtained recursively in terms of the traffic contributed by upstream adjacent sector S_{n+1} and the local traffic in sector S_n. Mathematically, this correlation is given by:

$$TF_{jS_n} = N_{S_n} + N_{S_{n+1}} + N_{S_{n-2}} + \cdots + N_{S_1} \tag{6}$$

$$ATF_{S_n} = \frac{TF_{jS_n}}{N_{S_n}} \tag{7}$$

To calculate Equation (6), it needs to find the average traffic load to sector S_n. This may be obtained with the help of the Average Traffic Factor (ATF) as a means to approximate the average data load to a sector, especially those sectors closer to the sink. The ATF allows calculating traffic load to any sector within the network. Algorithm 1 explains the functionality of traffic load, where $T_1 = \frac{(R-d)}{r}$ and $T_2 = \frac{(R-\zeta)}{r}$. It indicates that every node receives its distance from a neighboring node at the beginning of each round so that it may adjust its transmission range. The estimation of load is carried out using Equation (8) and then the suitable SN within the adjusted transmission range is chosen to establish a link with good quality. At each hop, nodes measure their data load and if it is high, *NN* directly forwards the data to *SN*. Otherwise, it is forwarded through neighbouring nodes. It is should mentioning that, nodes in the communication range of the sink will forward the data packets directly without the support of the intermediate nodes.

$$ATF_{jS_n} = \begin{cases} (T_1 + 1) + \frac{T_1(1+T_1)r}{2x}, & if\ x \geq \zeta \\ \frac{1}{2}(T_2 + 2)\zeta^2\theta\rho + \frac{1}{2}T_2 r\zeta\theta\rho(T_2 + 1), & otherwise \end{cases} \tag{8}$$

Algorithm 1: Calculation of Traffic Load and Energy Consumption per Round

Input: Parameters such as network range R, transmission range r between sensors, normal nodes (*NN*), super nodes (*SN*), node density ρ, etc.

Output: For a given node $i \in \{NN\}$, calculate the traffic load $tl_i^{r_0}$ and energy consumption e_i^{r0} for round r_0.

1) Initialize Parameters
2) For each round r_0:
3) For each node $i \in \{NN\}$:
4) Calculate the distance $d_{i,j}$ between node i and sector S_n where n is the number of sectors.
5) If $d_{i,j} \geq r$:
6) Send data from node i to node j. Calculate the traffic load using Equation (8) and energy consumption for transmitting and receiving.
7) Else:
8) Find a S_n in each sector.
9) If $d_{(NN,SN)} \leq r$:
10) Send data from *NN* to *SN*.
11) Calculate the traffic load and energy consumption for receiving data at SN and transmitting data at *NN*.
12) Else:
13) *SN* receives data on its own.
14) Calculate the traffic load and energy consumption for receiving data at SN.
15) While the sink receives data d_0:
16) Calculate the overall energy consumption and lifetime of nodes for round r_o.

4.2 Intelligent Node Selection

The PAN coordinator revaluates the number of slots in CFP (GTS^{new}) to be used by all node GTS to send their data after calculating a new SO and a new BO. It allocates GTS to each $node_{GTS}$ according to a new value of SO parameter by employing S JF to prioritize nodes having fewer GTS requests. This gives less network delay with a cost of small unfairness, since more nodes get to finish their transmission sooner compared to by standard procedure. Similar to standard, the time taken by CAP and by CFP remains a function of SO, but each slot within a CFP has been reduced by half to double its capacity from 7 to 14 within a superframe structure, as stated in ESS. The data capacity within each slot within a CFP CFT^{SLOT} can be calculated by a node by using the below-specified equation.

$$CFP^{SLOT} = 960 \times 2^{SO-3}(bits) \tag{9}$$

4.3 Effective Path Selection

It has been observed that quite a bit of bandwidth becomes wasted during the Contention-Free Period (CFP) in the standard superframe structure. The inefficiency becomes greater with higher number of slots during a CFP, leading to higher levels of bandwidth wastage. The issue has been addressed by the Enhanced Superframe Structure (ESS) by decreasing by half the size of each slot during a CFP, thereby doubling the number of slots and reducing slot-level bandwidth wastage.

However, even with such enhancement, ESS remains suboptimal to accommodate dynamic or time-varying data traffic, which could be changing within every Beacon Interval (BI). The static allocation scheme serves to constrain flexibility in supporting varying traffic needs. To transmit a data payload D_i from node i to the PAN coordinator, the estimated transmission time t_d can be calculated using the following expression:

$$t_d = \frac{D_i}{C} \tag{10}$$

The data rate at which a node communicates is denoted by C. The number of CFP slots needed to transmit D_i data, known as K_i, is then determined. The number of maximum bits that can be sent during each CFP slot, N_{bps}, is also calculated. A larger slot size means a higher capacity for transmitting data in a CFP. In ESS, N_{bps} remains constant due to its fixed SO throughput, but in ESS_{ADS}, it can change depending on the adaptive data traffic. If node *i* needs K_i,slots to send its data D_i to the PAN coordinator, the link utilization U_i for that node is computed using this formula. In this scenario, ts represents the duration in seconds of each CFP slot and is determined. The utilization of the link (U_{CFP}) is calculated for a specific number of nodes (p) that have been allocated CFP slots. Similarly,

The utilization for node i on the link under IEEE 802.15.4, termed as (U_{oi}), is computed to measure the effectiveness of the efficient bandwidth usage in CFP. Here, the index k_o corresponds to the number of CFP slots need to transmit the total data payload, and toto is the duration of one CFP slot in seconds. Additionally, U_{CFP0} the average link utilization of the whole CFP period in case total (q) nodes are assigned CFP slots during a BI. And can be calculated as:

$$K_i = \frac{D_i}{N_{bps}} \tag{11}$$

$$N_{bps} = 15 \times 2^{SO+3} \tag{12}$$

$$U_i = \frac{t_d}{K_i \times t_s} \tag{13}$$

$$t_s = 15 \times e^{-6} \times 2^{SO+5} \tag{14}$$

$$U_{CFP} = \sum_{i=1}^{p} \frac{t_d}{k_i \times t_s} \tag{15}$$

$$U_{o_i} = \frac{t_d}{k_0 \times t_0} \tag{16}$$

$$t_o = 15 \times e^{-6} \times 2^{SO+4} \tag{17}$$

$$U_{CFP} = \sum_{i=1}^{q} \frac{t_i}{k_0 \times t_0} \tag{18}$$

4.4 Cluster Head Selection Process

In the next phase, the distributed method of clustering is employed, utilizing n heterogeneous nodes alongside the centralized process for selecting cluster heads in the first phase, which involves m homogeneous nodes. The primary purpose of introducing heterogeneous nodes is to maintain a balance in the lifespan of the network across both regions. During the selection of cluster heads in the distributed approach, certain preliminary data, such as initial energy levels, remaining energy, node positions, and node statuses within the entire network are shared among all nodes through broadcasting hello packets. This is done by gradually increasing the signal strength to detect neighbouring nodes and form clusters.

In a second hierarchy level, two classes of nodes are used according to the initial energy: normal and advanced nodes. Regular nodes come with a basic amount of energy E_0, and advanced nodes are equipped with an additional α , where α in the range of [0,1] to indicate the fractional portion of E_0 that becomes the surplus. Therefore, every high-level node has the initial energy level of $E_0(1 + \alpha)$.

When the network starts, all advanced nodes have the same high initial energy. However, the network involves in data acquisition, routing, and broadcasting, the energy of each node is consumed along with- functional roles and communication load

To evaluate the average residual energy, we should make a sum of the total remaining energy of normal and advanced nodes. This can be summed and averaged over all the nodes to give an accurate estimate of the energy profile of the network at any-one time

$$E_{nor} = \sum_{i=1}^{a} E_0 \tag{19}$$

$$E_{adv} = \sum_{i=1}^{b} E_0(1 + \alpha_i) \tag{20}$$

Equations (19) and (20) can be used to determine the total energy (E_{Total}) in the second level. i represents the rotating epoch in each time-based round. The average residual energy $\bar{E}(r)$ in an epoch E_i can be computed. T can be obtained by calculating the threshold for selecting CHs, where P_{opt} is the predetermined number of CHs and P_i is the probability of a node being chosen as a CH among the second-level nodes.

$$E_{Total} = E_{nor} + E_{adv} \tag{21}$$

$$\bar{E}(r) = \frac{1}{n}\sum_{i=1}^{n} E_i \tag{22}$$

$$P_i = P_{opt}\,\frac{E_i}{\bar{E}(r)} \tag{23}$$

In a cluster head, G' and G'' denote the sets of eligible normal and advanced-energy nodes, which are candidate CHs. The present round is represented by r, and d is the distance from the BS. A random number for each node is generated s_i and if s_i is less than the fixed threshold value in Equation (24), that node is elected as a CH. Otherwise, it listens for a hello message of selected CHs. The next level zone consists of the heterogeneous nodes having different level of residual energy and the percentages required to be calculated for normal (P_{nor}) and advanced nodes (P_{adv}).

The network's second level of the network consists of heterogeneous nodes with different residual energy. This diversity enables optimised CH election probabilities to be calculated by giving values of (P_{nor}) and (P_{adv}) to be defined separately. The system parameters allow for CH selection, which is both adaptive and energy conscious regarding the type of nodes and the current energy state of the nodes, hence enhancing load balancing and prolonging the network lifetime.

$$T = \begin{cases} \frac{P_{nor}}{1 - P_{nor} \times (r \times mod \frac{1}{P_{nor}}) \times d'}, & if\ n \in G' \\ \frac{P_{adv}}{1 - P_{adv} \times (r \times mod \frac{1}{P_{adv}}) \times d'}, & if\ n \in G'' \\ 0, & otherwise \end{cases} \tag{24}$$

Likewise, the determination of P_i can be carried out for networks with two levels of diversity. In networks with multiple levels of diversity, clusters are created based on the energy levels of the individual nodes. Nodes at higher levels will have greater energy and are more likely to be chosen as cluster heads (CHs). To address this uncertainty, P_i can be computed as the single node probability in a multi-level of heterogeneous network. If p represents the CH target proportion, the total number of CHs in both first and second-level regions can be determined.

$$P_{nor} = \frac{P_{opt}}{1 + \alpha m} \tag{25}$$

$$P_{adv} = \frac{P_{opt}(1 + \alpha)}{1 + \alpha m} \tag{26}$$

$$P_i = \begin{cases} \frac{P_{opt}E_I(r)}{(1+\alpha m)\bar{E}(r)}, & if\ node\ is\ normal \\ \frac{P_{opt}(1+\alpha)E_I(r)}{(1+\alpha m)\bar{E}(r)}, & if\ node\ is\ advance \end{cases} \tag{27}$$

$$P_i = \frac{P_{opt} \times n \times (1+\alpha_i)}{n+\sum_{i=1}^{n} \alpha_i} \tag{28}$$

$$Total_{CHs} = \sum_{i=1}^{m\times p} CH_{S_{level\,1}} + \sum_{i=1}^{n\times p} CH_{S_{level\,2}} \tag{29}$$

4.5 Multi-hop Communication

Our suggested model is expanded to incorporate the use of multi-hoping during data transmission, which allows for better network scalability. This approach involves selecting intermediate Cluster Heads (CHs) based on their distance from each other and their received signal strength indication (RSSI), using a threshold determined by the network administrator. Firstly, all CHs broadcast their RSSI and location within a particular range set by the network administrator to be transmitted. Remote CHs then use this information to calculate the multi-hoping criteria (MHC), which is determined by the RSSI and mutual distance between intermediate CHs and the base station. the equation is used to calculate MHC:

$$MHC = \frac{RSSI_{ICHs}}{Dis_{ICHs} - BS} \tag{30}$$

The RSSI value of intermediate CHs, $RSSI_{ICHs}$, and the mutual distance between intermediate CHs and BS, $Dis_{ICHs} - BS$, play a crucial role. In addition to multi-hoping aid, the actual data transmission of sensor nodes based on a threshold helps conserve valuable energy resources.

4.6 Hybrid Optimization-based Clustering

This work presents a new approach called HOCCNN (Hybrid Osprey-Cluster Convolutional Neural Network) based on the Osprey Optimization Algorithm framework. The main goal of this algorithm is to improve cluster head (CH) selection in WSNs while solving important issues such as network lifetime and end-to-end delay. The proposed approach is divided into two phases: the first stage employs OOA method to determine optimal CHs, while the second phase clusters sensor nodes by associating with respective selected CH according to its distance Euclidean measures. This two-stage approach can effectively enhance the performance and effectivity of WSN.

The OOA-guided CH selection is inspired by nature, using the foraging of ospreys, or water hawks. Ospreys are birds of prey that weigh anywhere between 0.9 and 2.1 kilograms with body lengths of 50–66 cm and 127–180 cm of wingspan. Their hunting technique includes soaring around 40 meters above water bodies, spotting the fish prey normally weighing as much as 2 kilograms, and performing sharp dives with the aim of catching the fish using their talons. Once secured, the prey is transported to a safe location for consumption.

The Osprey Optimization Algorithm mimics the intelligent hunting behavior to handle complex optimization problems. The goal-driven adaptive strategy of the osprey constitutes a reference for OOA design in order to properly explore and exploit solutions space with CH selection. Through the incorporation of this biologically-inspired technique into WSN clustering architecture, HOCCNN can provide low-energy and minimal-delay solutions in dynamic resource-constrained sensor network scenarios.

$$O = \begin{bmatrix} O_1 \\ \cdot \\ \cdot \\ O_i \\ \cdot \\ \cdot \\ O_n \end{bmatrix}_{n\times m} = \begin{bmatrix} O_{1,1} & \dots & o_{1,j} & \cdots & o_{1,m} \\ & & \cdot & & \\ & & \cdot & & \\ O_{i,1} & \dots & o_{i,j} & \dots & o_{i,m} \\ & & \cdot & & \\ & & \cdot & & \\ O_{n,1} & \dots & o_{n,j} & \dots & o_{n,m} \end{bmatrix}_{n\times m} \tag{31}$$

$$O_{i,j} = LB_j + r_{i,j}.(upb_j - lob_j) \qquad i = 1,2 \dots, n, \quad j = 1,2, \dots.., \ m, \tag{32}$$

The Osprey Optimization Algorithm is a metaheuristic method based on populations whose purpose is to use a set of osprey agents that continuously seek the best solution of a problem. It is applied through several iterations such that each osprey is a candidate solution and is part of the group work with the aim of arriving at the global optimum

In this method, every osprey is mapped in concepts to each sensor node and mathematically formulated as a point in an n-dimensional search space. The osprey population is represented as a matrix as indicated in Equation (31), where all the search space points of the individuals stored

The network consists of n sensor nodes (or ospreys) uniformly randomly initialized within the search area through Equation (32). The variables n and m denote the number of ospreys (population size) and the number of problem decision variables (problem dimensions), respectively. The initialization of each position is controlled using the variable $r_{i,j}$ uniformly randomly generated within the range [0, 1]. The variables upb_j and lob_j denote the upper and lower bounds of the problem variable in the jth position, respectively, while ensuring that the initialization of the osprey position is uniformly distributed inside the feasible solution space.

$$F = \begin{bmatrix} F_1 \\ \cdot \\ \cdot \\ F_i \\ \cdot \\ \cdot \\ F_n \end{bmatrix}_{n\times 1} \tag{33}$$

The objective function uses Equation (33) to generate a set of values (osprey) by evaluating the problem. These estimated values are presented in a vector, where F stands for a set of values for the objective function is a numerical measure employed for evaluating the goodness of each candidate solution in the Osprey Optimization Algorithm (OOA). The fitness of the ith osprey in the iteration is described as F_i, and it denotes the objective function value of the analogous position of an individual in the search space. The algorithm identifies the best and the worst objective values of each iteration of the population for guiding the search process towards the best solutions.

The performance of each osprey is evaluated and then modified as necessary. The algorithm is motivated by the hunting behaviour of osprey attacks while searching for a prey during its search process. In the wild, ospreys can see fish swimming up to a metre below surface level and target very specific dives to catch them. Such a principle is mimicked in the beginning stage of OOA, where to imitate such kind of attack mechanism and enlarge search space as well as avoid falling into local optimums, population updated.

At this stage, the position of every osprey in search space is rooted based on object values at could be predefined targets like fish under water. It can be expressed as follows in a mathematical form from Equation (34) using the current optimal O_{best}. A new candidate position is created by randomly selecting a target location of attack and found the corresponding desired point using (35). If this new position possesses a better objective function value than the current one, then update as follows (using Equation 36) for osprey's movement. This updating rule of dynamic scale improves the exploring power of this algorithm and also balances well with exploitation, which account for its effectiveness in tackling complex optimization problems.

$$F_{pos_i} = \{O_w | w \in \{1,2..,n\} \wedge FH_w < FH_i\} \cup \{O_{best}\}, \tag{34}$$

$$o_{i,j}^{pos1} = 0_{i,j} + ren_{i,j}.\left(SF_{i,j} - R_{i,j}.y_{i,j}\right), \tag{35a}$$

$$o_{i,j}^{pos1} = \begin{cases} o_{i,j}^{pos1}, lob_j \le o_{i,j}^{pos1} \le upb_{j;} \\ lob_j, o_{i,j}^{pos1} < lob_j; \\ upb_j, o_{i,j}^{pos1} > upb_j; \end{cases} \tag{35b}$$

$$O_i = \begin{cases} o_{i,j}^{pos1}, FH_i^{pos1} < FH_i; \\ O_i, else, \end{cases} \tag{36}$$

In the initial phase of OOA, o_i^{pos1} represents the efficient location of the $i-th$ osprey, while $o_{i,j}^{pos1}$ indicates the jth dimension. FH_i^{pos1} refers to the objective function value, and SF_i represents the chosen fish for the $i-th$ osprey. The jth dimension of SFi is denoted by $SF_{i,j}$. The range [0,1] is represented by $ran_{i,j}$, as a random interval, and $R_{i,j}$ indicates a random number between 1 and 2.

During the Exploitation Phase, the selects osprey a secure position to consume its prey after successfully hunting. By the second phase of OOA, the population is updated by mimicking the natural habits of ospreys. As the osprey moves to a safe spot to eat, its position in the search area changes, with enhancing OOA exploit ability for local searches and generate improved solutions. Equation (37) is utilized in the design phase of OOA to simulate the osprey's actions and control a random location for each member of the population to feed on its catch. The previous osprey's position can adjust according to the superior objective function value as shown in Equation (38).

$$o_{i,j}^{pos2} = o_{i,j} + \frac{lob_j + ran_{i,j}.(upb_j - low_j)}{k}, 1 = 1,2,3, \dots, n, \;\; j = 1,2, \dots m, \;\; k = 1,2, \dots T, \tag{37a}$$

$$x_{i,j}^{pos2} = \begin{cases} o_{i,j}^{pos2}, lob_j \leq o_{i,j}^{pos2} \leq upb_j, \\ lob_j, o_{i,j}^{pos2} < lob_j, \\ upb_j, o_{i,j}^{pos2} > upb_j, \end{cases} \tag{37b}$$

$$O_i = \begin{cases} o_{i,j}^{pos2}, FH_i^{pos2} < FH_i; \\ O_i, else, \end{cases} \tag{38}$$

In the following step of Osprey Optimization (OOA), the current location of the $i - th$ osprey is illustrated as o_i^{pos2}, and $o_{i,j}^{pos2}$ denotes the location at the $j - th$ dimension. The corresponding objective function at this location is presented as $FH_{i,j}^{pos2}$. The search procedure engages stochastic variation with a random variable $ran_{i,j}$, uniformly produced from [0, 1]. The algorithm's progress is tracked using an iterative counter k and K represents the maximum iteration.

The fitness function also regulates the movement direction of the osprey, like finding the location of fish during natural hunting. The fitness analysis for the proposed HOCCNN framework takes into consideration major parameters such as the Residual Energy Ratio (RER) and distance from one node to the Base Station (BS) or Sink node. They significantly aid the determination of energy-efficient and topologically appropriate candidate solutions for the selection of Cluster Head (CH).

The Residual Energy Ratio (RER) of a sensor node shows how much energy is left in it after operation. It is calculated based on the difference between the current remaining energy and original total available energy of each node. This measure gives us information about the energy sustainability of a node, and it can be computed as in Eq. (39). While RER and distance are incorporated into the objective function of HOCCNN, which guarantees that selected cluster heads is energy-wise optimal as well as spatially proper to lengthen network life time and lower communication overhead.

$$RER(O_i) = \frac{Energy_{avail}}{Energy_{intial}} \tag{39}$$

Initial energy denotes the energy assigned to the node during network initialization, and energy efficiency denotes the node's usable energy at a certain time. For the case of choosing the cluster head (CH), there is the distance metric applied to get the best node by measuring how close one is to the sink node. Specifically, the Euclidean distance from one of the sensor nodes O_i to the sink node is computed using Equation (40) as guidance for this selection. The fitness function of how suitable an osprey's position in the search space is, is provided by Equation (41). During each iteration, the current position of an osprey is updated and its corresponding fitness value is compared with the one already saved. In the event the new position corresponds to a better (i.e., lower-cost or higher-benefit-based)-fitness value, it becomes the new updated position and the corresponding node becomes this iteration's Cluster Head.

This iterative selection scheme makes CHs energy-aware and topologically adaptive resulting in extended network lifetime, minimizing the communication overhead. The complete procedure for the CH selection mechanism is as depicted in Algorithm 2, which provides the entire pseudo code of our optimization-based approach.

$$dis(O_i, sink) = \sqrt{\sum_{i=1}^{n}(sink - O_i)^2} \tag{40}$$

$$O_{i(fitness)} = 0.5 \times (1 - RER(O_i) + 0.5 \times (1 - dis(O_i)) \tag{41}$$

Algorithm 2: Selection Algorithm for Optimal CH Node using HOCCNN

Input: Number of nodes in the network, denoted by $'n'$, and the total number of iterations, denoted by $'T'$

Output: Optimal position for osprey to act as CH node

1. Randomly initialize the network population
2. Calculate the objective function
3. For each iteration t, from 1 to T:
4. For each node i, from 1 to n:

5. //exploration phase
6. Update the fish position for each member of OOA using Equation (31).
7. Randomly determine a scaling factor (SF) using the ith osprey.
8. Compute the new position for the osprey using Equation (35a).
9. Verify that the boundary condition is satisfied using Equation (35b).
10. Update the position of the ith osprey using Equation (33).
11. //exploitation phase
12. Compute the new position for the osprey using Equation (37a).
13. Verify that the boundary condition is satisfied for this new position using Equation (37b).
14. Update the position of the osprey using Equation (35).
15. Evaluate the fitness function using Equation (38).
16. If an osprey reaches an optimal position in the network, then:
17. Set that osprey as the best candidate for CH.
18. Else:
19. Go back to step 1.
20. END
21. END
22. Return candidate CH.

After selecting a CH, the sensor nodes $'n'$ is arranged into clusters within the network. This selection process is vital for maintaining energy efficiency and prolonging the network's lifespan. The use of Euclidean distance plays a significant role in this clustering process, as it ensures effectiveness within the network. Each node calculates its distance from potential CHs within the network range, and together, all nodes make an intelligent decision based on proximity. This method is beneficial for situations where energy consumption and node communication need to be optimized, as shown in Equation (42) where O_i and O_j represent two distinct nodes within the network area.

$$dis(O_i\ , O_j) = \sqrt{\sum_{i=1}^{n}(O_j - O_i\ )^2}, \tag{42}$$

5 Result Experimentation

The implementation of the proposed HOCCNN model is carried out in the MATLAB software and the parameters which are considered for the analysis are accuracy, precision, recall, F1 score, energy consumption, lifetime, delivery ratio and throughput. The obtained results of the proposed HOCCNN model are compared with the earlier baseline models like TESEES [26], AOEBOA [27] and MOMSDC [28].

5.1 Comparison of Accuracy %: The accuracy of predictive models is a widely used method in ML. It also determines the proportion of cases that have been correctly classified. Below is the mathematical formula for calculating accuracy.

$$Accuracy = \frac{TP-TN}{TP+TN+FP+FN} \tag{43}$$

Based on the above equation, the term TP (True Positives) represents the number of cases that were accurately identified as positive. A list of cases that are classified as accurate negatives is given by TN (True Negatives) FP (false positives) refers to the number of cases that were detected as false positives, also known as false alarms. The term FN (false negatives) is employed to denote the number of cases that were not classified as positive numbers, also known as missed detection. Figure 4 shows the graphical experimental accuracy calculation of the methods used in the proposed HOCCNN model.

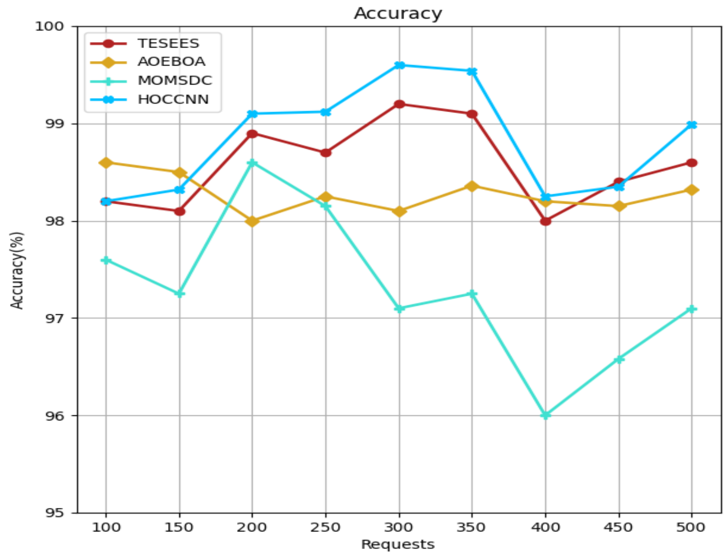

Figure 4. Accuracy Calculation

5.2 Comparison of Precision %: In machine learning, accuracy is an important indicator when evaluating a prediction. This is used to determine how accurately the model can detect positive cases. It can be mathematically expressed as the ratio given in Equation (44).

$$Precision = \frac{TP}{TP+FP} \quad (44)$$

From the above equation, the term TP refers to correctly identifying cases as positive, while FP refers to incorrectly identifying cases as positive (when they are negative). Figure 5 shows the precision graphical experimental calculation used in the proposed HOCCNN model.

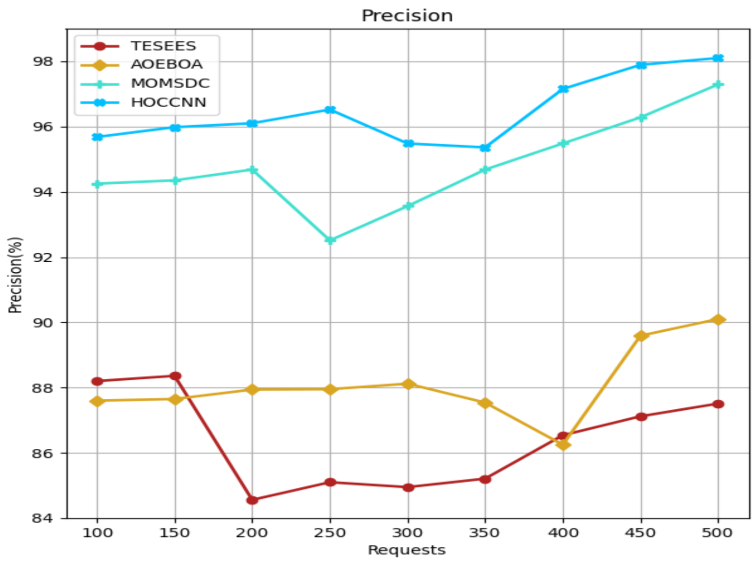

Figure 5. Precision Calculation

5.3 Comparison of Recall %: The so-called sensitivity or true positive rate is another important measure used to evaluate assignments based on ML predictions. It evaluates positive cases from all true positive cases to accurately identify them. Mathematically, the recall can be calculated as followed by equation 45.

$$Recall = \frac{TP}{TP+FN} \quad (45)$$

Based on the above equation, the terms TP refer to cases that are accurately predicted as positive, while FN refers to those. which are falsely predicted as negative (even though they are positive). The recall process is mainly used to achieve the maximum accuracy of the positive prediction process. Achieving the maximum recovery score leads to the correct classification of data-positive cases. Figure 6 shows the graphical experimental recall calculation of the methods used in the proposed HOCCNN model.

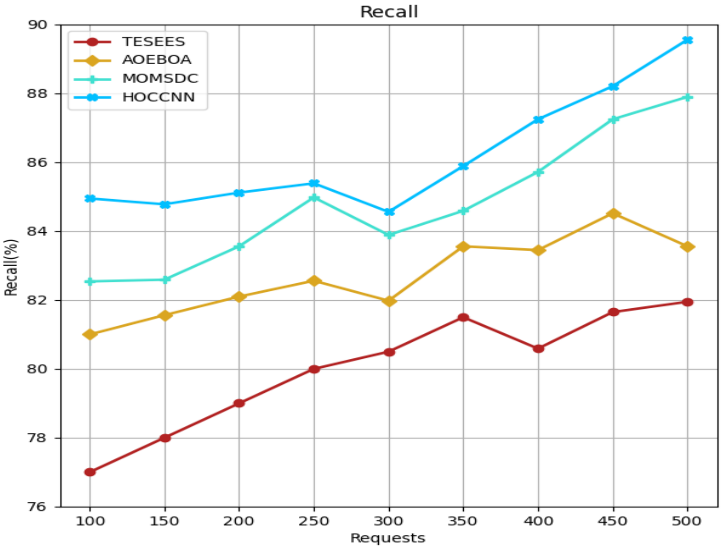

Figure 6. Recall Calculation

5.4 Comparison of F1-Score %: One of the most frequently used measures for evaluating machine learning predictions is the F1 measure. It aims to find a compromise between precision and recall, providing a single score that takes both metrics into account. The measure F1, which is also expressed mathematically in Equation 46, is based on the values of the harmonic mean of precision and memory.

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{46}$$

F1 score, which emphasizes smaller values, is more robust against the imbalance between precision and recall, and the proposed HOCCNN achieved the maximum F1 score of up to 91.56. Figure 7 is shown below the graphical experimental of F1-Socre calculation of the methods used in the proposed HOCCNN model.

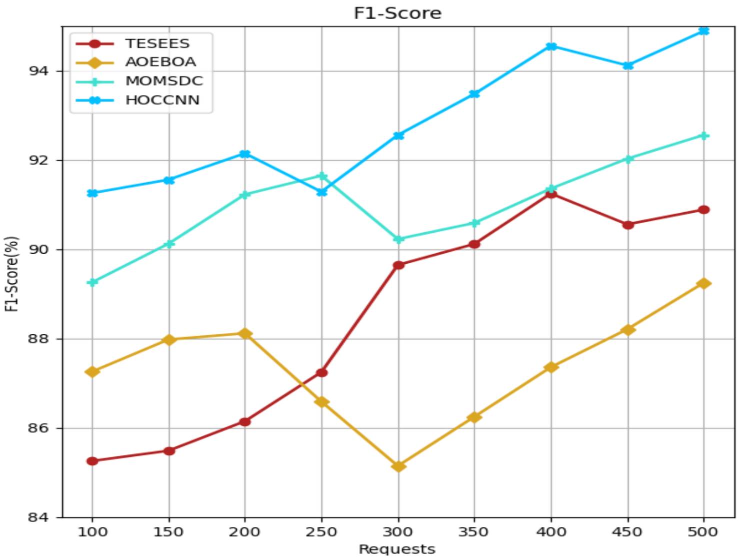

Figure 7. F1-Score Calculation

5.5 Comparison of Energy Consumption: The total amount of energy used by the network is determined by adding up the energy usage of each node for transmitting packets. This calculation is carried out using the formula provided in Equation (47). In the figure 8 the calculation of energy consumption is illustrated and it gets compared with the others.

$$Energy_{con} = \frac{Node\ consumed\ energy\ on\ each\ data\ transfer}{Network\ total\ energy\ consumption} \tag{47}$$

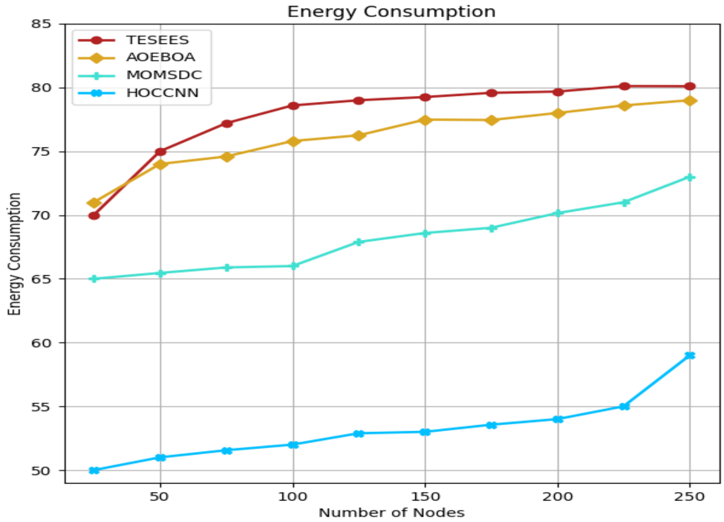

Figure 8. Energy Consumption Calculation

With the presence of hybrid optimization-based clustering process the energy consumption of the proposed HOCCNN is lower than the other methods like TESEES, AOEBOA and MOMSDC.

5.6 Comparison of Network Lifetime: The lifespan of a network, which indicates how long it can effectively perform its designated tasks, is known as its network lifetime. The calculation for the network lifetime can be found in equation (48) and the graphical output of network lifetime is shown in figure 9.

$$Network_{Lifetime} = \frac{Network\ time\ utilization}{Total\ simulation\ time} \quad (48)$$

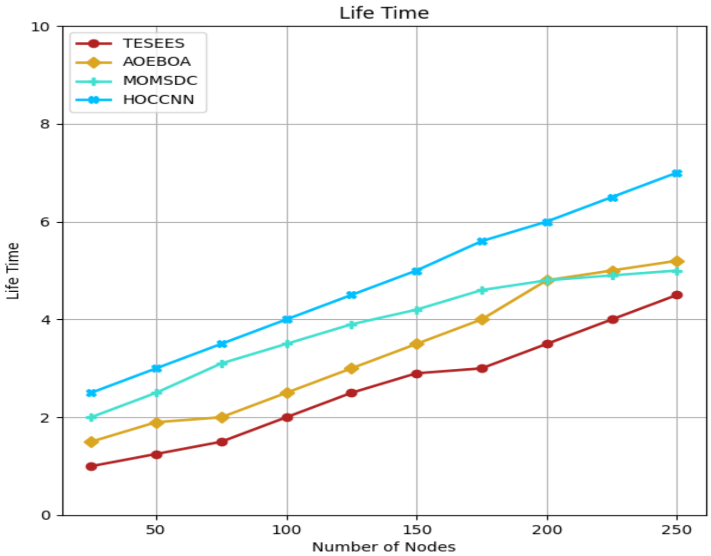

Figure 9. Network Lifetime Calculation

Which is the presence of efficient node selection process and clustering model the efficiency of each node which are present in the HOCCNN is improvised and that helps to increase the overall lifetime of the nodes which are present in the network.

5.7 Comparison of Packet Delivery Ratio: The packet delivery ratio, also referred to as the delivery percentage, measures the proportion of packets that are successfully received out of all packets that were sent. This ratio can be determined using the equation provided in Equation (49) and the graphical output of packet delivery ratio is illustrated in figure 10.

$$PDR_{ratio} = \frac{Total\ successful\ received\ packets}{Total\ transmitted\ packets} \quad (49)$$

Improved traffic model is incorporated with the HOCCNN and that greatly increased the success rate of data transmission at the time of communication between each node in the network.

5.8 Comparison of Network Throughput: Throughput, also referred to as data transfer rate, is the speed at which information is transmitted through a network. It can be determined by utilizing the equation (50) provided in the given formula. In the figure 11 the calculation of network throughput is shown.

$$Network_{thr} = \frac{Successful\ packets \times packet\ size}{Time\ taken\ to\ transmit\ the\ data} \tag{50}$$

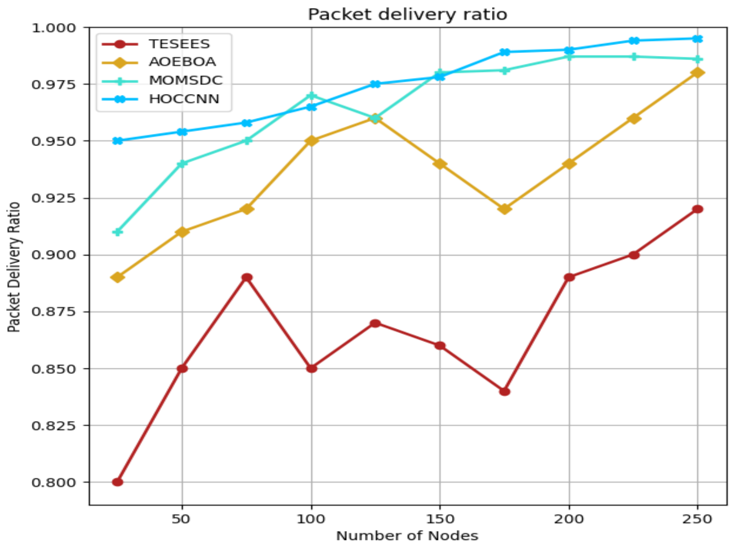

Figure 10. Packet Delivery Ratio Calculation

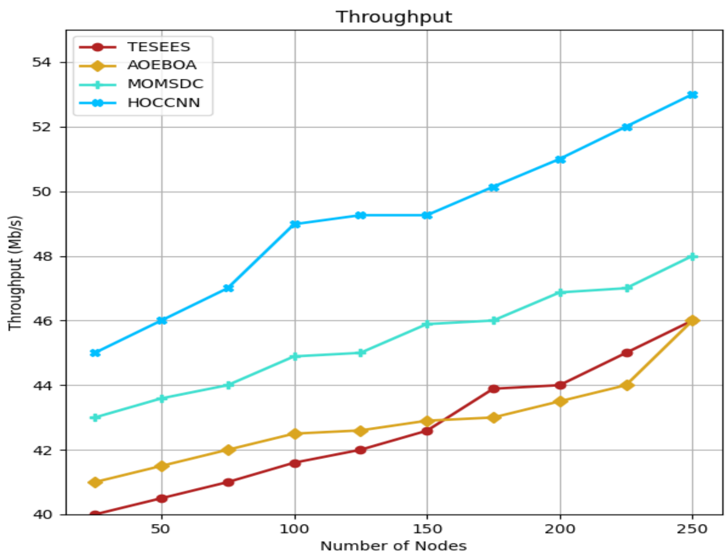

Figure 11. Throughput Calculation

Both the clustering process and effective traffic model improve the communication quality that greatly reduces the routing overhead at the time of data transmission which helps to increase the throughput of the network. In table 2 the overall comparative value analysis is given.

Table 2: Comparative Analysis

Algorithms	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)	Energy Consumed (%)	Network Lifetime (s)	PDR (%)	Throughput (Mb/Sec)
TESEES	98.4	85.6	80.1	89.13	78.45	2.51	0.85	41.764
AOEBOA	98.2	87.5	83.5	88.55	77.28	3.33	0.95	43.826
MOMSDC	97.5	94.2	87.3	90.59	68.98	4.56	0.965	47.367
HOCCNN	99.1	95.6	88.9	91.56	54.18	6.76	0.985	53.198

6 Conclusion

A HOCCNN model is a powerful approach to enhance the performance of heterogeneous WSNs by employing hierarchical clustering and hybrid optimization techniques. The model employs the OOA to determine the most suitable CH and enables multi-hop data transmission, which effectively addresses both energy conservation and network scalability. Due to the fluctuating nature of heterogeneous WSNs, factors like residual energy and node mobility are considered in the system model. Energy and traffic models offer a comprehensive model that allows for both energy consumption and data traffic management, while also ensuring the network's efficient operation. Intelligent node selection and efficient path selection further optimize the network by favouring nodes with sufficient residual energy and adjusting transmission areas based on traffic load. HOCCNN is a hybrid optimization clustering technique that utilizes nature-inspired algorithms to optimize CH selection and network performance. Multi-hop communication not only increases network coverage but also saves energy by optimizing data transmission paths. The HOCCNN model is an important factor in the heterogeneous WSN field because it enhances network lifetime, energy efficiency, and scalability.

Funding: "This research received no external funding"

Conflicts of Interest: "The authors declare no conflict of interest."

References

[1] M. Trigka and E. Dritsas, "Wireless Sensor Networks: From Fundamentals and Applications to Innovations and Future Trends," *IEEE Access*, 2025, doi: 10.1109/access.2025.3572328.

[2] W. Osamy, A. M. Khedr, A. Salim, A. I. AlAli, and A. A. El-Sawy, "Recent studies utilizing artificial intelligence techniques for solving data collection, aggregation and dissemination challenges in wireless sensor networks: A review," *Electron.*, vol. 11, no. 3, p. 313, 2022, doi: 10.3390/electronics11030313.

[3] A. Abba Ari *et al.*, "Data collection in IoT networks: Architecture, solutions, protocols and challenges," *IET Wirel. Sensor Syst.*, vol. 14, no. 4, pp. 85–110, 2024, doi: 10.1049/wss2.12080.

[4] M. Jubair *et al.*, "Optimization of clustering in wireless sensor networks: Techniques and protocols," *Appl. Sci.*, vol. 11, no. 23, p. 11448, 2021, doi: 10.3390/app112311448.

[5] M. Jubair, R. Hassan, A. H. M. Aman, and H. Sallehudin, "Social class particle swarm optimization for variable-length Wireless Sensor Network Deployment," *Appl. Soft Comput.*, vol. 113, p. 107926, 2021, doi: 10.1016/j.asoc.2021.107926.

[6] D. B. Rawat and S. R. Reddy, "Software defined networking architecture, security and energy efficiency: A survey," *IEEE Commun. Surveys Tuts.*, vol. 19, no. 1, pp. 325–346, 2016, doi: 10.1109/comst.2016.2618874.

[7] N. Chandnani and C. N. Khairnar, "An analysis of architecture, framework, security and challenging aspects for data aggregation and routing techniques in IoT WSNs," *Theor. Comput. Sci.*, vol. 929, pp. 95–113, 2022, doi: 10.1016/j.tcs.2022.06.032.

[8] P. Rosa, A. Souto, and J. Cecílio, "Light-SAE: A lightweight authentication protocol for large-scale IoT environments made with constrained devices," *IEEE Trans. Netw. Service Manag.*, vol. 20, no. 3, pp. 2428–2441, 2023, doi: 10.1109/tnsm.2023.3275011.

[9] S. He, Q. Li, M. Khishe, A. Salih Mohammed, H. Mohammadi, and M. Mohammadi, "The optimization of nodes clustering and multi-hop routing protocol using hierarchical chimp optimization for sustainable energy efficient underwater wireless sensor networks," *Wirel. Netw.*, vol. 30, no. 1, pp. 233–252, 2024, doi: 10.1007/s11276-023-03464-9.

[10] S. Ghazy, G. Kaddoum, and S. Singh, "Low-latency low-energy adaptive clustering hierarchy protocols for underwater acoustic networks," *IEEE Access*, vol. 11, pp. 50578–50594, 2023, doi: 10.1109/access.2023.3277395.

[11] A. Alimi *et al.*, *Network-on-Chip Topologies: Potentials, Technical Challenges, Recent Advances and Research Direction*. London, U.K.: IntechOpen, 2021, doi: 10.5772/intechopen.97262.

[12] S. Lata, S. Mehfuz, and S. Urooj, "Secure and reliable WSN for Internet of Things: Challenges and enabling technologies," *IEEE Access*, vol. 9, pp. 161103–161128, 2021, doi: 10.1109/access.2021.3131367.

[13] R. Gupta, D. Srivastava, M. Sahu, S. Tiwari, R. K. Ambasta, and P. Kumar, "Artificial intelligence to deep learning: Machine intelligence approach for drug discovery," *Mol. Divers.*, vol. 25, no. 3, pp. 1315–1360, 2021, doi: 10.1007/s11030-021-10217-3.

[14] Hemalatha Reddy, J. Sinjini, M. Priyadharshini, V. Indumathi, V. Bhavani, and T. Jahan, "Optimizing energy efficiency in wireless sensor networks using machine learning techniques," in *Proc. Int. Conf. Inf. Manag. Eng.*, 2023, pp. 599–610, doi: 10.1007/978-981-97-9262-7_52.

[15] T. Kim, L. F. Vecchietti, K. Choi, S. Lee, and D. Har, "Machine learning for advanced wireless sensor networks: A review," *IEEE Sensors J.*, vol. 21, no. 11, pp. 12379–12397, 2020, doi: 10.1109/jsen.2020.3035846.

[16] N. Waheed, X. He, M. Ikram, M. Usman, S. S. Hashmi, and M. Usman, "Security and privacy in IoT using machine learning and blockchain: Threats and countermeasures," *ACM Comput. Surv.*, vol. 53, no. 6, pp. 1–37, 2020, doi: 10.1145/3417987.

[17] T. C. Akinci, G. Erdemir, A. T. Zengin, S. Seker, and A. I. Idriss, "Machine learning-based error correction codes and communication protocols for power line communication: An overview," *IEEE Access*, vol. 11, pp. 124760–124781, 2023, doi: 10.1109/access.2023.3330690.

[18] M. Beko and S. Tomic, "Toward secure localization in randomly deployed wireless networks," *IEEE Internet Things J.*, vol. 8, no. 24, pp. 17436–17448, 2021, doi: 10.1109/jiot.2021.3078216.

[19] Surenther, K. Sridhar, and M. K. Roberts, "Enhancing data transmission efficiency in wireless sensor networks through machine learning-enabled energy optimization: A grouping model approach," *Ain Shams Eng. J.*, vol. 15, no. 4, p. 102644, 2024, doi: 10.1016/j.asej.2024.102644.

[20] N. R. Saadallah, S. A. Alabady, and F. Al-Turjman, "Energy-efficient cluster head selection via genetic algorithm," *Al-Rafidain Eng. J.*, vol. 29, no. 1, pp. 12–25, 2024, doi: 10.33899/rengj.2023.143955.01293.

[21] S. K. Gebresilassie, J. Rafferty, L. Chen, Z. Cui, and M. Abu-Tair, "Transfer and CNN-based de-authentication (disassociation) DoS attack detection in IoT Wi-Fi networks," *Electronics*, vol. 12, no. 17, p. 3731, 2023, doi: 10.3390/electronics12173731.

[22] R. K. Singh, P. P. Puluckul, R. Berkvens, and M. Weyn, "Energy consumption analysis of LPWAN technologies and lifetime estimation for IoT application," *Sensors*, vol. 20, no. 17, p. 4794, 2020, doi: 10.3390/s20174794.

[23] M. D. Khan *et al.*, "Energy harvested and cooperative enabled efficient routing protocol (EHCRP) for IoT-WBAN," *Sensors*, vol. 20, no. 21, p. 6267, 2020, doi: 10.3390/s20216267.

[24] Xu, M. Lu, H. Zhang, and C. Pan, "A novel multi-agent model for robustness with component failure and malware propagation in wireless sensor networks," *Sensors*, vol. 21, no. 14, p. 4873, 2021, doi: 10.3390/s21144873.

[25] S. Hao, Y. Hong, and Y. He, "An energy-efficient routing algorithm based on greedy strategy for energy harvesting wireless sensor networks," *Sensors*, vol. 22, no. 4, p. 1645, 2022, doi: 10.3390/s22041645.

[26] S. Sadeq, R. Hassan, H. Sallehudin, A. H. M. Aman, and A. H. Ibrahim, "Conceptual framework for future WSN-MAC protocol to achieve energy consumption enhancement," *Sensors*, vol. 22, no. 6, p. 2129, 2022, doi: 10.3390/s22062129.

[27] M. Bilal, E. U. Munir, and F. K. Alarfaj, "Hybrid clustering and routing algorithm with threshold-based data collection for heterogeneous wireless sensor networks," *Sensors*, vol. 22, no. 15, p. 5471, 2022, doi: 10.3390/s22155471.

[28] J.-W. Kim, J. Kim, and J. Lee, "Cross-layer MAC/routing protocol for reliability improvement of the Internet of Things," *Sensors*, vol. 22, no. 23, p. 9429, 2022, doi: 10.3390/s22239429.

[29] N. López-Vilos, C. Valencia-Cordero, R. D. Souza, and S. Montejo-Sánchez, "Clustering-based energy-efficient self-healing strategy for WSNs under jamming attacks," *Sensors*, vol. 23, no. 15, p. 6894, 2023, doi: 10.3390/s23156894.

[30] S. H. Abdul-Qawy *et al.*, "An enhanced energy efficient protocol for large-scale IoT-based heterogeneous WSNs," *Sci. Afr.*, vol. 21, p. e01807, 2023, doi: 10.1016/j.sciaf.2023.e01807.

[31] V. Pandiyaraju, S. Ganapathy, N. Mohith, and A. Kannan, "An optimal energy utilization model for precision agriculture in WSNs using multi-objective clustering and deep learning," *J. King Saud Univ.-Comput. Inf. Sci.*, vol. 35, no. 10, p. 101803, 2023, doi: 10.1016/j.jksuci.2023.101803.

[32] S. Mohan and S. Panda, "Multi-factor optimized mobile sink data collection framework for hybrid WSN-LTE assisted IoT network," *Heliyon*, vol. 10, no. 5, 2024, doi: 10.1016/j.heliyon.2024.e25998.