FMDB *Transactions on Sustainable Intelligent Networks*

Oppositional Pufferfish Optimization Algorithm-Based Cluster Head Selection and Congestion Trust-Aware Energy-Efficient Clustering Routing Protocol in IoT-WSN

T. Kanimozhi[1,*], S. Belina V. J. Sara[2]

[1]Department of Computer Science, Faculty of Science and Humanities, SRM Institute of Science and Technology, Ramapuram, Chennai, Tamil Nadu, India.
[2]Department of Computer Applications, Faculty of Science and Humanities, SRM Institute of Science and Technology, Kattankulathur, Chengalpattu, Tamil Nadu, India.
kanimozhimcaa@gmail.com[1], sbelinav@srmist.edu.in[2]

Abstract: In the current era, the Internet of Things (IoT) has become an essential technology for interlinking physical devices with the Internet. Wireless Sensor Networks (WSNs) are often employed to monitor large-scale, diverse environments. However, the constrained energy supply of IoT-WSN sensor nodes creates substantial challenges to sustaining network operations. Clustering schemes offer a practical solution for optimising energy consumption and prolonging network lifespan. Nevertheless, IoT-WSN data collection often suffers from congestion, which leads to packet loss, reduced reliability, and degraded throughput. The Congestion Trust-Aware Energy-Efficient Clustering Routing (CTAEECR) protocol promises to improve network trust management, congestion control, energy efficiency, and lifespan. The network region is hierarchically split into clusters using Weighted Possibilistic Fuzzy C-Means (WPFCM) during setup to reduce packet overhead. Each cluster's Main Cluster Head (MCH) and optional Secondary Cluster Head (SCH) rotate between nodes to balance energy consumption during data transmission. The Oppositional Puffer Fish Optimisation Algorithm (OPOA) optimises cluster head selection and rotation using pufferfish defenses and predator–prey interactions. Exponential-decay Trust-aware routing employs DWIQL. Traffic-based routing takes packets off busy paths. The protocol manages data and congestion for reliable communication. MATLABR2021b contrasts CTAEECR and clustering-based routing. Performance measures include throughput, average energy use, latency, network longevity, PDR, and PLR. Simulations demonstrate CTAEECR surpasses benchmark protocols in efficiency, reliability, and network sustainability.

Keywords: Internet of Things (IoT); Main Cluster Head; Packet Loss Ratio; Secondary Cluster Head; Energy Efficient Clustering; Packet Delivery Ratio; Direct Trust; Indirect Trust.

Received on: 28/02/2025, **Revised on:** 07/05/2025, **Accepted on:** 17/07/2025, **Published on:** 03/01/2026

Journal Homepage: https://www.fmdbpub.com/user/journals/details/FTSIN

DOI: https://doi.org/10.69888/FTSIN.2026.000603

Cite as: T. Kanimozhi and S. B. V. J. Sara, "Oppositional Pufferfish Optimization Algorithm-Based Cluster Head Selection and Congestion Trust-Aware Energy-Efficient Clustering Routing Protocol in IoT-WSN," *FMDB Transactions on Sustainable Intelligent Networks*, vol. 3, no. 1, pp. 26–44, 2026.

1. Introduction

[*]Corresponding author.

The Internet of Things (IoT) is an important source of technology solutions for a wide range of applications. WSN, this reduces the cost of adopting new technologies. By leveraging smart sensor node networks with Internet connectivity, this integration not only reduces costs but also enhances convenience in daily life [2]. WSN is a low-cost, well-established technology that has been applied across multiple domains, including intelligent transportation systems, military surveillance, environmental monitoring, and industrial control [1]. It provides extensive physical data that can be repurposed for diverse applications. Therefore, combining IoT with WSN does not require a significant paradigm shift. The benefits of WSN-based IoT include low cost and ease of implementation. However, the primary challenge in WSNs is network longevity [3]. To address this, energy efficiency must be incorporated into network routing protocols, thereby extending network lifetime and enabling wireless nodes to operate for longer periods without battery replacement in WSN-based IoT environments [4]. A novel optimisation method has been proposed to achieve optimised network longevity and energy efficiency. The clustering technique is an effective method for enhancing energy efficiency and prolonging network lifetime. Clustering protocols partition the network into multiple clusters, with each cluster managed by a designated Cluster Head (CH). This approach lowers energy consumption and enhances network longevity by reducing the need for frequent long-distance transmissions [5]; [6]. The CH collects information from the sensor nodes within its cluster and transmits it to the receiver. Various data-aggregation algorithms are used within clustering protocols to process this data and forward it to the Base Station (BS) as meaningful information.

The CH plays a critical role in maintaining the energy savings achieved by clustering approaches. The cluster's Member Nodes (MNs) send their sensed data to the CH, which then forwards the aggregated information to the BS via single- or multi-hop transmission. The clustering structure remains a significant challenge in WSNs, as it can reduce network lifetime by leading to inefficient energy use [7]; [8]. Furthermore, data aggregation and route discovery are often affected by an inadequate clustering structure [9]. Consequently, the overall lifetime of a WSN is heavily influenced by the efficiency of its clustering design. A common issue is the inaccurate assessment of the distance to the CH. Many clustering algorithms that aim to create balanced, non-overlapping clusters struggle to determine the optimal number of clusters when relying on existing models. Another limitation arises in CH selection: most distributed approaches ignore routing information during selection. Since sensor nodes are typically deployed in unprotected or hostile environments, their routing protocols are highly vulnerable to attacks. These attacks can be broadly categorised into internal and external threats [10]. To defend against external attacks, security mechanisms based on identity verification and cryptography have been proposed, enabling IoT-WSNs to operate robustly and securely. However, such protection methods are ineffective against internal attacks, as they depend on node cooperation and complete trust among network nodes [11]; [12]. Moreover, these mechanisms require substantial memory and involve complex computations, thereby increasing energy overhead. Consequently, trust mechanisms are effective in addressing internal attacks on WSNs [13]. Trust-based security methods predict a node's future behaviour based on its past actions [14].

However, traditional mechanisms still face several limitations, including their inability to defend against diverse attack types, increased network bandwidth usage, higher data loss rates, collisions, and elevated energy consumption. These challenges remain significant concerns in IoT-WSNs. To address congestion, a congestion management system has been developed to monitor and regulate the number of packets entering the network, ensuring traffic levels remain within the desired threshold. Effective decision-making techniques are essential in this context to inform choices about current congestion conditions. Moreover, the successful reception of packets relies heavily on the presence of an optimal congestion management mechanism [15]; [16]. In this paper, the Congestion Trust-Aware Energy-Efficient Clustering Routing (CTAEECR) protocol is proposed to enhance network longevity, energy efficiency, congestion management, and trust. The setup step is performed only in the first round, during which the network area is partitioned into levels and sectors, thus minimising protocol overhead. Main Cluster Head (MCH) and Secondary Cluster Head (SCH) alternate among cluster nodes at the beginning of each cycle, ensuring balanced energy consumption by distributing the CH load across all nodes. For CH selection and rotation, the Oppositional Puffer Fish Optimisation Algorithm (OPOA) is employed. Protocol trust is further reinforced through Direct Trust (DRT), Indirect Trust (IDRT), and Witness Trust (WT) mechanisms. Additionally, a Dual Reward Improved Q-Learning (DWIQL) method with an exponential decay time factor is introduced for trust-aware routing. Simulation results in MATLAB R2021 b demonstrate that the CTAEECR protocol outperforms existing clustering-based routing protocols.

2. Literature Review

Farsi et al. [17] proposed a Congestion-aware Clustering and Routing (CCR) protocol to address the congestion problem in WSNs. By appropriately selecting the Primary Cluster Head (PCH) and Secondary Cluster Head (SCH), the CCR protocol aims to reduce end-to-end delay and extend network lifetime. The setup phase is performed only during the initial round, whereas the transmission phase operates in two layers (intra-cluster communication and inter-cluster routing). CCR protocol improves throughput, packet delivery ratio, end-to-end latency, and overall network longevity. Yan and Qi [18] proposed a multi-attribute method for routing decisions using the Congestion-Aware Routing Algorithm (CARA), which detects congestion at nodes and in their surrounding areas. In CARA, the selection of the optimal next-hop node is performed using a multi-attribute decision-making strategy that considers four factors: node load, forwarding rate, remaining cache capacity, and average remaining cache

during forwarding. To prevent packet loss and alleviate congestion, the protocol avoids forwarding additional packets to nodes when the load factor is low. By incorporating congestion awareness into routing, the CARA algorithm improves network transmission throughput while adapting to congestion conditions in both sensor nodes and their environments. Compared with other congestion control algorithms, CARA achieves lower energy consumption and reduced network congestion. Sangeetha et al. [19] proposed Congestion Aware Routing using Fuzzy Ruleset (CARF) to manage excess traffic conditions in WSNs. CARF identifies non-localised node paths, integrates them with existing localised node paths, and applies fuzzy rule-based prediction to select more reliable, congestion-free routes to the sink node.

The CARF model consists of two phases: (1) Congestion Detection, and (2) Multiple Path Identification using non-localised node placement. Data packets are then routed to the sink node in a congestion-mitigated manner. In this approach, the unknown coordinates of a sensor node are estimated using a non-localised node positioning algorithm that generates additional packet transmission channels on top of the existing ones. Non-localised nodes are located through a geometric method based on the Point in Which Side (PIWS) hop count. Furthermore, the Enhanced Fuzzy-based Congestion Mitigation (ECFM) algorithm is introduced to estimate node congestion levels using fuzzy rule sets. The CARF model was implemented and evaluated using the Network Simulator 2 (NS-2). Chanak and Banerjee [20] proposed a distributed traffic-aware congestion control system for IoT-enabled WSNs. Nodes are organised into multiple tiers to transmit sensing data. The proposed design focuses on constructing a cross-layer architecture that coordinates functions across multiple layers to improve communication reliability and efficiency in IoT systems. Its main components are: (i) a middleware mechanism for bridging the wireless personal area network application subnet with the Internet backbone, (ii) a congestion-free and stable routing scheme at the network layer, (iii) efficient access regulation and power adaptation in the MAC sublayer, and (iv) a lightweight yet robust transport-layer control strategy. Tumula et al. [21] introduced an Opportunistic Energy-efficient Dynamic Self-configuration Routing (OEDSR) protocol to reduce energy consumption and improve Quality of Service (QoS) in IoT applications. In OEDSR, the residual energy and mobility factors of sensor nodes are obtained via a graph-theoretic routing-tree model, which is then used to determine the optimal path to the BS.

Sensor node clustering is performed using the Steiner tree algorithm, while a hierarchical tree architecture and dynamic cluster formation help identify the ideal path, minimising connections and reducing energy consumption. OEDSR leverages network properties, communication patterns, and node heterogeneity to optimise energy usage. Finally, OEDSR is evaluated against existing routing protocols in terms of throughput, latency, and Packet Delivery Ratio (PDR). Sharma et al. [22] proposed a Multi-Level Hierarchical Secure and Optimal Routing (ML-HSOR) protocol to enhance network performance and extend network lifespan. In the clustering process, a Markov model with an adaptive weighting mechanism is employed to select the most suitable node to act as the CH. To detect malicious nodes, the protocol incorporates multi-level trust evaluation and authentication. To ensure secure data transmission, the Polarity Learning-based Chimp Optimisation Algorithm (PL-COA) is applied after encrypting the aggregated message and timestamp, enabling selection of the optimal transmission path. The proposed system reduces energy consumption while achieving higher PDR, throughput, detection ratio, and lower latency than other existing methods. Shende and Sonavane [23] proposed an Energy and Trust-aware Multicast Routing (ETR) protocol called CrowWhale-ETR. In this approach, the Crow Search Algorithm (CSA) is combined with the Whale Optimisation Algorithm (WOA), with the objective function designed based on node energy and trust factors. Using this hybrid optimisation (CWOA), node trust and energy are evaluated to determine optimal routes in the WSN model. The selected secure paths are then used for data transmission, thereby improving network security. During each transmission, the energy and trust values of individual nodes are updated, with simulations conducted on 50 nodes for analysis.

The proposed approach achieves minimal delay, a maximum detection rate, higher residual energy, and increased throughput under both normal and attack scenarios. Additionally, TBSEER is highlighted as a mechanism to improve data transmission reliability while reducing routing overhead. Hu et al. [24] developed a Trust-Based Secure and Energy-Efficient Routing (TBSEER) protocol for WSNs. In TBSEER, the comprehensive trust value is computed from adaptive direct trust, indirect trust, and energy trust, making the protocol resilient against attacks. A volatilisation factor and an adaptive punishment mechanism are incorporated to detect rogue nodes more quickly. To further reduce energy consumption (EC) caused by repeated computations, nodes calculate only direct trust values, while the sink node determines indirect trust values. In addition, Cluster Heads (CHs) mitigate wormhole attacks by selecting the most secure multi-hop paths based on the comprehensive trust value. Average speed, Packet Loss Rate (PLR), Average End-to-End Delay (E2ED), and energy consumption are evaluated under various attack scenarios using MATLAB.

Hassan et al. [25] proposed an Improved Energy-Efficient Clustering Protocol (IEECP) to extend the lifetime of WSN-based IoT systems. Firstly, the protocol determines the optimal number of clusters to form overlapping balanced clusters. Secondly, the Modified Fuzzy C-Means (M-FCM) algorithm is introduced to create balanced static clusters, thereby minimising and equalising the energy consumption of sensor nodes. Thirdly, the CH Selection-Rotation Algorithm (CHSRA) is introduced for CH selection and rotation. Furthermore, a dynamic threshold for CH rotation is defined, ensuring more balanced energy consumption across successive cluster heads within the cluster. IEECP reduces and balances node energy consumption, making

it well-suited for networks that require extended lifetimes through improved clustering. IEECP outperforms existing protocols in terms of both energy efficiency and network longevity. Raslan et al. [26] proposed an Improved Sunflower Optimisation (ISFO) algorithm for Cluster Head (CH) selection in IoT-WSNs. In ISFO, the Lévy flight operator is integrated with the SFO algorithm to balance intensification and diversification processes, thereby avoiding local minima. The ISFO model aims to achieve optimal energy consumption control, thereby extending the lifespan of IoT-WSN networks.

Rizwanullah et al. [27] proposed an efficient routing model that adaptively selects the optimal path for data transfer between IoT devices to prolong the lifespan of WSNs. The Hybrid Muddy Soil Fish Optimisation-based Energy-Aware Routing Scheme (HMSFO-EARS) is developed for optimal path selection during transmission. To enhance route selection, HMSFO-EARS integrates the Adaptive β-Hill Climbing (ABHC) technique with the MSFO algorithm, thereby maximising network lifetime and reducing energy consumption. The proposed method achieves higher throughput, lower latency, reduced energy usage, and extended network lifetime. Srivastava and Paulus [28] investigated multi-objective optimisation for joint energy- and longevity-aware clustering (ELR-C) based routing in WSN-IoT. The Multi-Objective Chaotic Slime Mould (MCSM) algorithm is applied to the routing protocol to attain optimal clustering and enhance overall energy efficiency. An Improved Butterfly Optimisation (IBO) method based on trust degree is employed to optimise design parameters for CH selection. Furthermore, a Cat Hunting-based Feed-Forward Neural Network (CH-FFNN) is developed to facilitate multi-hop routing between CH and sink nodes, thereby improving energy efficiency and network lifetime. The ELR-C routing protocol achieves superior performance compared to existing methods in terms of energy utilisation and network sustainability.

3. Proposed Methodology

In this paper, a Congestion Trust-Aware Energy-Efficient Clustering Routing (CTAEECR) protocol is proposed that incorporates trust while addressing congestion and energy efficiency in the IoT-WSN architecture. The Weighted Possibilistic Fuzzy C-Means (WPFCM) method is employed to construct clusters. To optimise CH selection during packet transmission, the Oppositional Pufferfish Optimisation Algorithm (OPOA) is combined with a rotation mechanism. The Dual Reward Improved Q-Learning (DWIQL) model updates nodes' trust levels. To mitigate congestion, the proposed mechanisms reduce energy consumption, balance load distribution, and enhance network longevity, stability, reliability, and scalability. Figure 1 illustrates the overall workflow of the proposed model.

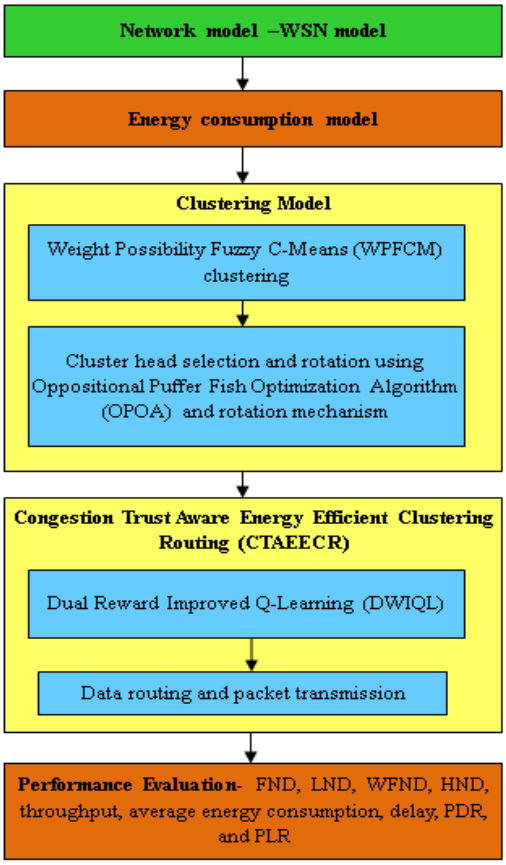

Figure 1: Overall flow of proposed model

3.1. Energy Consumption Model

The energy usage of sensor nodes is evaluated using a radio energy consumption model [29]. Based on the transmission distance, either the free-space model or the multi-path fading model is applied in the energy consumption model. Accordingly, the energy consumption of a transmitter (TX) node for sending a message of size L bits over a distance (dis) is given by equations (1-2):

$$\mathrm{En_{TX}}(\mathrm{l},\mathrm{dis}) = \begin{cases} \mathrm{En_{elec}} * \mathrm{l} + \mathrm{En_{AD}} * \mathrm{l} + \varepsilon_{\mathrm{fs}} * \mathrm{L} * \mathrm{dis}^2, \mathrm{dis} \le \mathrm{dis}_0 \\ \mathrm{En_{elec}} * \mathrm{l} + \mathrm{En_{AD}} * \mathrm{l} + \varepsilon_{\mathrm{amp}} * \mathrm{L} * \mathrm{dis}^4, \mathrm{dis} > \mathrm{dis}_0 \end{cases} \quad (1)$$

$$dis_0 = \sqrt{\frac{\varepsilon_{fs}}{\varepsilon_{amp}}} \quad (2)$$

Where $\varepsilon_{\mathrm{fs}}$ and $\varepsilon_{\mathrm{amp}}$ is denoted as the energy consumption for free-space propagation and multipath propagation, respectively. dis_0 is the distance threshold between the TX and receiver (RX), and $\mathrm{En_{elec}}$ and $\mathrm{En_{AD}}$ is denoted as the energy consumption of electronics and data aggregation by equation (3):

$$En_{RX}(L) = En_{elec} * L \quad (3)$$

Where $\mathrm{En_{RX}}(\mathrm{L})$ is denoted as the energy consumed by the receiver-side node. When data is received from a member node (MN), aggregated by the MN, and sent to the BS, each CH consumes energy. The sensing data is sent to the BS via multi-hop communication outside the sensing area. Equation (4) receives the sensor data using multi-hop communication [30]:

$$En_{CH} = LEn_{elec}\left(\frac{N}{K} - 1\right) + LEn_{AD}\frac{N}{K} + LEn_{elec} + L_{\varepsilon f_s} dis_{BS}^2 \quad (4)$$

Where the symbol $\mathrm{dis_{BS}^2}$denotes the distance between the CH and the subsequent hop. Thus, the energy consumption of each MN is given by equation (5) [3]:

$$En_{non-CH} = LEn_{elec} + L_{\varepsilon f_s} dis_{CH}^2 \quad (5)$$

Where the symbol indicates the distance from MN-CH $\mathrm{dis_{CH}^2}$. Equation (6) is a description of the energy of each cluster:

$$En_{cluster} = En_{CH} + \left(\frac{N}{K} - 1\right) En_{non-CH} \cong E_{CH} + \left(\frac{N}{K}\right) E_{non-CH} \quad (6)$$

Equation (7): total energy consumption ($\mathrm{En_{total}}$) for K clusters:

$$En_{total} = NLEn_{elec} + NL_{\varepsilon f_s}\left(\frac{1.262}{2\pi}\frac{M^2}{K}\right) + NLEn_{elec} + NLEn_{AD} + KL_{\varepsilon f_s} d_{BS}^2 \quad (7)$$

Equation (8) determines the number of clusters:

$$K = \sqrt{\frac{1.262N}{2\pi}}\frac{M}{d_{BS}} \quad (8)$$

WPFCM is recommended for creating balanced clusters.

3.2. Cluster Formation and Setup

Possibility of Weight Using the shortest intra-cluster distance and energy efficiency, clusters are formed using the fuzzy C-Means (WPFCM) algorithm. Only in cases where the target function optimises the distance between cluster centres while minimising the distance between clusters and nodes inside clusters. It is described by equation (9) [30]:

$$\mathrm{J_{PFCM}}(\mathrm{t},\mathrm{u},\mathrm{v}) = \sum_{\mathrm{i}=1}^{\mathrm{c}}\sum_{\mathrm{j}=1}^{\mathrm{n}}\left(\mathrm{au_{ij}^m} + \mathrm{bt_{ij}^q}\right)\mathrm{D_{ij}^2} + \sum_{\mathrm{i}=1}^{\mathrm{c}}\sum_{\mathrm{j}=1}^{\mathrm{n}}\left(1 - \mathrm{t_{ij}}\right)^{\mathrm{q}} \quad (9)$$

Where $\mathrm{m},\mathrm{q} > 1, \eta > 0, 0 < \mathrm{u_{ij}}, \mathrm{t_{ij}} < 1, 1 \le \mathrm{i} \le \mathrm{c}, \; \&1 \le \mathrm{j} \le \mathrm{n}$. $\mathrm{D_{ij}^2} = \|\mathrm{x_j} - \mathrm{v_i}\|^2$symbolised as the Euclidean distance, while a and b stand for the constants. Equation (10) is used to calculate γ_{ij}:

$$\gamma_{ij} = \exp\left(\frac{\sum_{j=1}^{n}\|x_j-\bar{x}\|^2\times c}{-\|x_j-\bar{x}\|^2\times n}\right) \quad (10)$$

The following is an explanation of WPFCM's objective purpose:

$$J_{WPFCM}(U,T,V) = \sum_{i=1}^{c}\sum_{j=1}^{n}\left[\left(1-\gamma_{ij}\right)u_{ij}^{m}+\gamma_{ij}t_{ij}^{q}\right]D_{ij}^{2}+\sum_{i=1}^{c}\sum_{j=1}^{n}\left(1-t_{ij}\right)^{q} \quad (11)$$

Therefore, γ_{ij}It is the weight between the nodes $x_j$and class I, as per equation (11). The centroid matrix $(c\times 1)$, typicality matrix $(c\times n)$, and membership matrix $(c\times n)$ are represented by U, T, and V, respectively. Here, u_{ij} represents the node x_j membership in a cluster c_i while t_{ij} It represents its typicality within the cluster c_i. $K=1$ and $\eta_i = K\left(\sum_{j=1}^{n}u_{ij}^{m}\right)$. This process is repeated until the minimum cluster size exceeds the defined threshold, ensuring smaller intra-cluster distances.

In the first round, when WPFCM forms clusters, the setup process is performed only once. Only the rotation of the MCH and SCH roles occurs at the start of each round; the generated clusters remain fixed for subsequent rounds. There are sectors and levels within the network region. A cluster is created with each level and sector converging. Each node will need to be sensitive to its cluster number, with the area number (nc_{la}), and level number (l). An MCH node and, if available, an SCH node are required for every cluster. During setup, the distance determines which MCH and SCH nodes are selected. The sink node initiates the process of determining each node's level number by repeatedly broadcasting a "HELLO" message with an increasing radius until the entire network is covered. Each node assigns itself a level based on the number of messages it receives. To maintain balance, each cluster is designed to contain an equal number of nodes, and the optimal cluster size is calculated accordingly:

$$nc_{la} = \left[\frac{n_l}{n_a}\right] \quad (12)$$

In equation (12), where n_l- is the number of nodes in level (l), n_a- is the total number of areas, and nc_{la}-is the number of cluster nodes in level (l) and area (a). The SCH, the node in each cluster that is most remote from the MCH node, is used by the MCH for data transfer. The CHs at the following level are located in the MCH, which was chosen as the node closest to the prelisted MCH. Before selecting the MCH, each cluster balances the number of nodes. In the IoT-WSN model, this process is repeated until every node in the network has been assigned to a specific cluster with accurate division and level numbers. After the nodes have been clustered to communicate with each other, the MCH and SCH nodes need to be selected. For the best CH election, the Oppositional Puffer Fish Optimisation Algorithm (OPOA) has been presented.

3.3. Oppositional Puffer Fish Optimization Algorithm (OPOA) Based Cluster Head Selection

A population-based method for selecting the best MCH is the OPOA approach. Based on its node position in the CH search space, each optimisation member selects the optimal values for the MCH. From a mathematical perspective, an OPOA member is a potential solution to the MCH selection in the IoT-WSN model. Each component of this vector represents energy efficiency, distance, and trust. The algorithm's population consists of all OPOA members. These natural processes are modelled, including: (i) a predator attacking a pufferfish, and (ii) the pufferfish's defensive mechanism against the predator. Equation (13) is used to initialise each member location at the start of the OPOA:

$$X = \begin{bmatrix} X_1 \\ \vdots \\ X_i \\ \vdots \\ X_N \end{bmatrix}_{N\times m} \quad (13)$$

$$x_{i,d} = lb_d + rn.(ub_d - lb_d) \quad (14)$$

Where N is the no. of population, m is the number of parameters such as energy, distance, and trust, $rn \in [0,1]$ is a random number, $lb_d$and ub_d are the lower and upper bounds of the parameters in the MCH selection, X is the population matrix, and $x_{i,d}$ is its d^{th} dimension in the MCH search space. Equation (15), fitness function as a vector for each OPOA member as the best MCH selection:

$$F = \begin{bmatrix} F_1 \\ \vdots \\ F_i \\ \vdots \\ F_N \end{bmatrix}_{N\times 1} = \begin{bmatrix} F(X_1) \\ \vdots \\ F(X_i) \\ \vdots \\ F(X_N) \end{bmatrix}_{N\times 1} \quad (15)$$

Where $F_i$is the objective function based on the i[th] member, and F is the objective function vector. The best MCH member is associated with the goal function's best value, and the worst MCH member is associated with the objective function's worst value.

3.3.1. Phase 1: Exploration Phase

The location of the population members (MCH) within the clustering is adjusted during the first phase of OPOA according to the pufferfish predator-attack strategy. Each population member is a predator in the OPOA design, and the ideal node position for the prospective pufferfish to attack is determined by the MCH positions of other population members with higher energy efficiency, shorter distances, and greater trust. Equation (16) is utilised to identify it:

$CP_i = \{X_k : F_k < F_i and k \neq i\}$ where i=1 to N and $k \in \{1\ to\ N\}$ (16)

Where X_k is the population member with the highest fitness value compared to the i[th] predator, $F_k$is the objective function, and CP_i is the set of potential pufferfish placements as the best MCH solution for the i[th] predator. The predator chooses a pufferfish at random; this pufferfish is called the selected pufferfish (SP) in the OPOA. The objective function value of this SP is then enhanced in the new MCH position, which, in accordance with Equation (17), takes the place of the member's old MCH position:

$$x_{i,j}^{P1} = x_{i,j} + rn_{i,j} \cdot \left(SP_{i,j} - I_{i,j} \cdot x_{i,j}\right) \quad (17)$$

$$X_i = \begin{cases} X_i^{P1}, F_i^{P1} \leq F_i \\ X_i, else, \end{cases} \quad (18)$$

Where SP_i Does the ith predator choose the pufferfish from the CP_iset, $SP_{i,j}$is the j[th] dimension of the i[th] predator, $X_i^{P1}$is the new MCH position of the i[th] predator, $x_{i,j}^{P1}$ is the j[th] dimension of its predator, F_i^{P1} is its objective function, $I_{i,j} = 1$ or 2 are randomly chosen numbers, dependent on the position of MCH, and $rn_{i,j} \in [0,1]$ are random numbers.

3.3.2. Phase 2: Exploitation Phase

The OPOA algorithm modifies the positions of individuals in the population by replicating the defence mechanism pufferfish use against predators. Equation (19) is used to determine a new MCH position for each OPOA member by modelling the predator's position and the CH movement away from it. Equation (20), the corresponding member is then updated with this new MCH position if it results in an improved fitness value:

$$x_{i,j}^{P2} = x_{i,j} + (1 - 2rn_{i,j}) \cdot \frac{ub_j - lb_j}{t} \quad (19)$$

$$X_i = \begin{cases} X_i^{P2}, F_i^{P2} \leq F_i \\ X_i, else, \end{cases} \quad (20)$$

Where X_i^{P2} is based on the defence mechanism, $x_{i,j}^{P2}$ is the new MCH location for the i[th] predator. The predator P2 has the j[th] dimension, $rn_{i,j} \in [0,1]$ are random numbers, and t is the iteration count. The initial iteration completes when the clustering framework identifies the optimal Main Cluster Head (MCH) by considering the positional information of all nodes in the context of exploration and exploitation dynamics. In the POA model, the degree of convergence is enhanced through the Oppositional Based Learning (OBL) mechanism. OBL helps identify the optimal global solution and accelerates convergence. OBL is the generation of the current population's opposite position, which is much closer to the global CH solution by equation (21):

$$x' = lb + ub - x \quad (21)$$

X is denoted as the current MCH solution, and x' SCH represented the opposite solution. Equations (16–21) are used to update the positions of OPOA members until the last iteration ends the process. The best OPOA member location is updated and saved as optimal MCH and SCH in each iteration. Algorithm 1 presents pseudo code. Figure 2 displays the flowchart of an optimization-based clustering approach for wireless sensor networks (WSNs) that finds the optimal Mobile Cluster Head (MCH) and Sensor Cluster (SC) solution. The first step is to set parameters such as the number of nodes, distance, energy, and trust. Then, the population size (N) and the maximum number of iterations (T) are set.

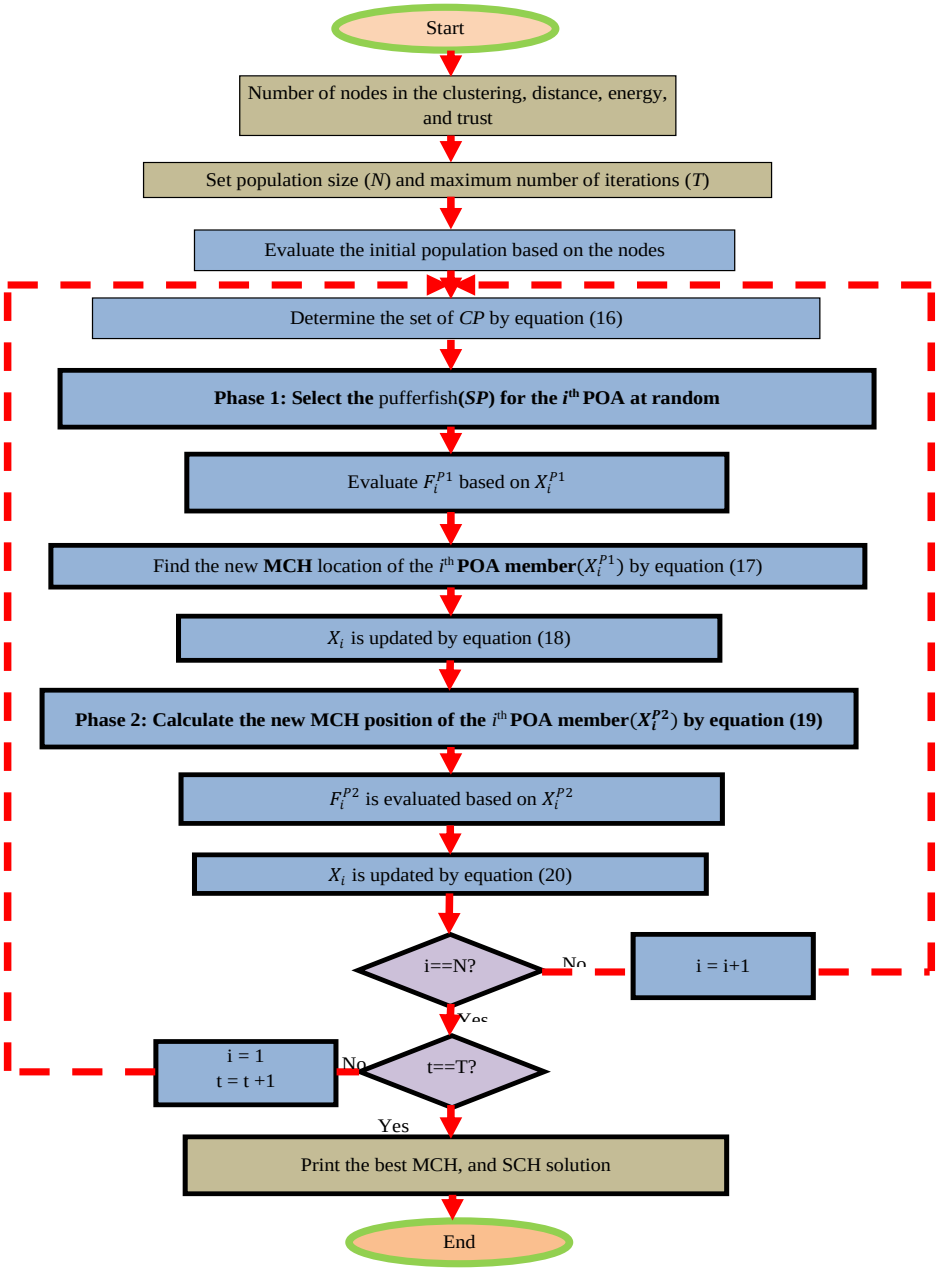

Figure 2: Flowchart of OPOA

The first group is reviewed, and the candidates are chosen. The algorithm goes through two steps: first, it picks a solution at random, checks how well it works, and then it uses the equations to update the MCH location and node positions. This process continues for all members of the population until the maximum number of iterations is reached. Lastly, the algorithm gives the optimum MCH and SC solution.

ALGORITHM 1: Oppositional Puffer Fish Optimization Algorithm (OPOA)
Start

1. Input CH selection based on the clusters, distance, energy, and trust
2. Set population size (N) based on the number of nodes, MCH, SCH, and iterations (T)
3. Create the initial population matrix X based on MCH, SCH at random by Equation (14)
4. Compute the fitness function.
5. For t = 1 to T
6. For i= 1 to N
Phase 1: Exploration Phase
1. Determine the candidate pufferfish set as optimal MCH and SCH for the i^{th} member by Equation (16)
2. Choose the target MCH pufferfish for the i^{th} member.
3. Choose the best MCH location of the i^{th} member by Equation (17)
Phase 2: Exploration Phase
4. Calculate the new MCH location of the i^{th} member by Equation (18)
5. Update with member as MCH by Equation (19)
6. Update with member as SCH by Equation (20)
7. end for i
8. Save the best candidate MCH and SCH solution.
9. end for t
10. Output the best MCH solution obtained by OPOA
End

3.4. CTAEECR Protocol

There are several rounds in the CTAEECR procedure. There are setup and transmission steps in every round. At the start of the first round, a setup step is completed, and WPFCM divides the network region into sectors S and levels L to build clusters. Every level and sector intersect to form a cluster, which is led by the SCH and its Main Cluster Head (MCH). The responsibilities of the MCH and SCH nodes are rotated among the cluster nodes during a brief setup phase that precedes subsequent rounds. The data transmission stage follows.

3.4.1. Trust Evaluation Model

The evaluation, diffusion, and discovery of nodes serve as the basis for determining trust during data transmission:

- **Node Discovery:** To understand their neighbours' actions, nodes use direct observation. The requested nodes, or evaluators, will decide what information they need from their neighbours to make the selection. After monitoring, related nodes may observe specific neighbour actions.
- **Trust Evaluation:** Direct Trust (DRT) is measured through direct exchanges between a source node and its adjacent nodes, along with interactions among those neighbours. Indirect Trust, on the other hand, captures the reliability inferred from the way nodes communicate with each other. To determine overall trust during the routing process, the direct node transmits the Indirect Trust (IDRT) value to the source node. The evaluated node's trust in its direct or indirect neighbours is known as Witness Trust (WT). Dependability, coverage, energy efficiency, and reputation are all seen as critical components in evaluating trust at all levels. Each node at each tier has its score determined using the Analytical Hierarchical Process (AHP).
- **Trust Update and Distribution:** In a distributed network, nodes cannot monitor the overall status or progress of the WSN. Nodes cannot be recharged; they will ultimately run out of power. By examining their past and present activities, nodes in this scenario learn more about their network. Each evaluating node stores trust values. An evaluator can either construct a new trust table or use the route in its trust table to transfer a packet when it gets one that has to be moved. Trust values are not updated continuously; instead, they are updated only in response to changes. Any expired trust values are discarded. Direct or indirect nodes will send the latest information upon request from source nodes. The trust value is updated using Dual Reward Improved Q-Learning (DWIQL). One part of Q-Learning is the reward function. The agent may find it simpler to determine the most effective route to the goal. Consequently, a dual reward function comprises both a dynamic and a static component.

3.4.2. Congestion Control Mechanism and Routing

Congestion during packet transmission from source to destination is also calculated once the trust level has been determined in the IoT-WSN network. The network includes N static sensor nodes randomly distributed in the surroundings. There are three steps: 1) setup, 2) request distribution, and 3) intra- and inter-cluster routing:

- **Setup Step:** This step is executed only once when the network is first initialised. Following deployment, each node in this phase determines which nodes are its single-hop neighbours, and the deployed nodes are further categorised into various tiers.
- **Request Distribution Step:** Phase of request distribution: In this phase, the gateway determines the location and level of the cluster member node and determines the packet lifespan in the IoT-WSN network. The packet lifetime can reduce message overhead in the IoT-WSN and improve data transmission reliability. Equation (22) determines the query packet lifespan at the node level:

$$nl_{lifetime\_packet} = \sum_{1}^{des(level)} (T_d + R_d) * P_d \quad (22)$$

Where des(level) is the destination node level. Transmission delay is denoted by T_d, receiving delay is denoted as R_d, and processing delay by P_d. The gateway then sends an RREQ message to all nodes in the level one cluster. The message includes a query for the destination node identifier (ID), level (l) value, lifetime of the request ($nl_{lifetime_packet}$link capacity, and destination node location. The destination node provides the BS with its updated information after the request propagates. Several requests for the same or different IoT devices may be sent to the BS simultaneously. The network may get congested if the BS processes these requests simultaneously. As a result, the suggested system uses the energy-efficient, highest-capacity link to determine the query's priority. To do this, the node uses the highest-capacity link to send a packet to the closest parent node containing information about the perceived event. When traffic demand in an IoT network exceeds the network's bandwidth, packets accumulate in node caches, leading to congestion. This indicates that in the IoT-WSN model, network congestion occurs when a node n_i delivers packets to n_j, $sr_i > sr_j$. This causes the cache queue length of node n_j to steadily increase. Equation (23) provides the congestion index (CI), which helps determine the degree of congestion in the Internet of Things network:

$$CI = \sum_{i,j \in n, n_i, n_j \in N} (sr_i - sr_j) + Q_i \quad (23)$$

$$Q_i = \begin{cases} 1, & Length(Q) \geq Thershold \\ 0, & Length(Q) < Thershold \end{cases} \quad (24)$$

For an IoT-WSN network, the CTAEECR protocol is used to evaluate routing delay, energy consumption, and congestion throughout the data collection process. The entire network lifetime is separated into $p + 1$ stages. $[S_0, S_1, S_2, \ldots., S_p]$, where S_i is the i^{th} network stage. The $[a_0, a_1, a_2, \ldots., a_p]$ is denoted as the duration at each stage, with the number of data phases at each stage S_iAll packets are divided into three groups based on the CI: high priority (HP), low priority (LP), and medium priority (MP). The packet header of every packet indicates its type. Several packets are transmitted using the suggested priority queue-based scheduler. The scheduler stops the transmission of LP and MP packets and begins the transmission of HP packets when they are detected in the queue. The scheduler initiates the transmission of LP packets and MP packets once all HP packets have been transmitted:

- **Intra-Cluster Routing:** This method uses single-hop communication to deliver packets from each cluster's conventional nodes to their MCH nodes. Time Division Multiple Access (TDMA) divides time among nodes for transmission. During its designated time periods, each node sends packets. For each cluster node, the MCH node must identify the destination node (MCH or SCH). In the initial transmission time slot, each cluster node is assigned a certain number of slots to send packets. For the remaining slots, each node enters sleep mode after sending its packets to the MCH nodes in its slots. When routing within a cluster, the MCH node is essentially always awake. The CH nodes (MCH and SCH) combine the packets sent by each node to their CH node for inter-cluster routing.
- **Inter-Cluster Routing:** Every MCH node attempts to transfer its packets to the sink node via inter-cluster routing. Every sector at all levels uses the same Code-Division Multiple Access (CDMA) to transmit packets, since each division forwards its packets to the same division at the pre-level until they arrive at the sink node. As the packets travel to the sink node, they are aggregated once more in each MCH. Depending on the number of MCH and SCH nodes and the size of the packets that each level must send, transmission is split up into multiple phases. Deployed nodes choose the optimal routing path based on the routing prediction of a lower likelihood of congestion. Depending on how many packets are in the current path, packets are routed from one route to another to control congestion. The data routing algorithm is used to implement congestion control.

4. Results and Discussion

Matrix Laboratory R2021b (MATLABR2021b) is used to simulate the performance comparison of routing algorithms. Two significant phases have been employed to illustrate its significance and coherence. The first step was to assess each clustering

model. The second step examined how the suggested protocol performed compared with current techniques. A WSN in the network scenario was used to test the protocol. One BS was situated outside the network, and nodes were arranged at random throughout a 2-dimensional square region of length (M×M). Two distinct scenarios were employed in the simulation to evaluate the outcomes of the clustering techniques. Every section offers a detailed discussion of Table 1, which contains the simulation parameters. It is explained as follows:

- Simulation: there are 100 nodes, 100×100 m^2. Simulation 2 involves measuring their length and the area of 1000 nodes by 1000×1000 m^2.
- It includes determining the size of the sensing region; many nodes are placed in expansive areas.

Table 1: Simulation parameters

Parameters	Value
En_{elec}	50 nj/bit
ε_{elec}	10pj/bit/m^2
ε_{amp}	0.0013pj/bit/m^4
L-Packet size	3200 bit
En_{ini}	1 j
En_{AD}	5Nj/bit
N	100;1000
M	100m×100m;1000m×1000m
BS location	(50, 125); (500, 1250)
No. of CH	4
No. of the source node	1
BS	1

Many metrics, such as FND, LND, WFND, HND, throughput, average energy consumption, delay, PDR, and PLR, are used to evaluate the performance of routing protocols:

- **First Node Dies (FND):** The first node's death is commonly referred to as FND. It is an input factor in determining WSN network stability and survival, and is used to evaluate network performance.
- **Last Node Dies (LND):** The last node death is commonly referred to as LND. It is also a round number at which the last node in the network fails. The time frame between FND and LND is referred to as the instability period. This period should be as short as possible.
- **Half Node Dies (HND):** The time interval between the beginning of the network process and the death of 50% of the network nodes is known as HND. Because it reduces the number of communication hops available to the sink node, which covers a smaller area and negatively impacts network performance, this option is crucial.
- **Weighted First Node Dies (WFND):** Weighted First Node Dies (WFND) showed a correlation between the steady and uneven periods. WFND was determined by applying equation (25):

$$WFND = \frac{FND}{LND-FND} \tag{25}$$

- **Throughput:** In IoT-WSN, throughput is the quantity of bits sent to the BS. Bits per second are used to measure it.
- **Average Energy Consumption:** The average amount of energy used at each node per iteration.
- **Delay:** The entire amount of time a packet spends travelling across a network, including processing, waiting, and transmission times at each hop, is known as delay, according to equation (26):

$$\mathrm{D(i,j)} = \sum_{k=1}^{H}\left(D_{trans}^{(k)} + D_{prop}^{(k)} + D_{proc}^{(k)} + D_{queue}^{(k)}\right) \tag{26}$$

Where the hop count between source i and destination j is denoted by H.$D_{trans}^{(k)}$ is denoted as the transmission delay at node k. The propagation delay at node k is represented by $D_{prop}^{(k)}$. The processing delay at node k is represented by $D_{proc}^{(k)}$.$D_{queue}^{(k)}$ is denoted as the queuing/waiting delay at node k:

- **Packet Delivery Ratio (PDR):** The percentage of sent packets that reach the destination. PDR is calculated by equation (27):

$$\text{PDR} = \frac{\text{Number of packets received}}{\text{Total number of packets sent}} * 100 \quad (27)$$

- **Packet Loss Ratio (PLR):** PLR is denoted as the percentage of sent packets that fail to reach the destination. PLR is calculated by equation (28):

$$\text{PLR} = \frac{\text{Number of packets Lost}}{\text{Total number of packets sent}} * 100 \quad (28)$$

A higher PDR indicates a lower PLR, indicating better network reliability and performance.

Table 2: Network lifetime comparison of clustering protocols

Protocols	Simulation-100 nodes				Simulation-1000 nodes			
	Rounds			Ratio	Rounds			Ratio
	FND	HND	LND	WFND	FND	HND	LND	WFND
ELR-C	879	1895	2216	0.657	459	1651	2917	0.187
OCM-FCM	1078	1786	2369	0.835	627	1489	2694	0.303
IEECP	1861	2218	2394	3.491	1296	1835	2882	0.817
WPFCM-EEC	1977	2379	2486	3.884	1598	2014	3088	1.072
EECTAR	2092	2566	2597	4.142	1745	2160	3355	1.084
CTAEECR	2251	2789	2716	4.840	1928	2357	3492	1.233

Network lifetime analysis among active nodes (100 and 1000) using clustering protocols (ELR-C, OCM-FCM, IEECP, WPFCM-EEC, EECTAR, and CTAEECR) is illustrated in Figure 3.

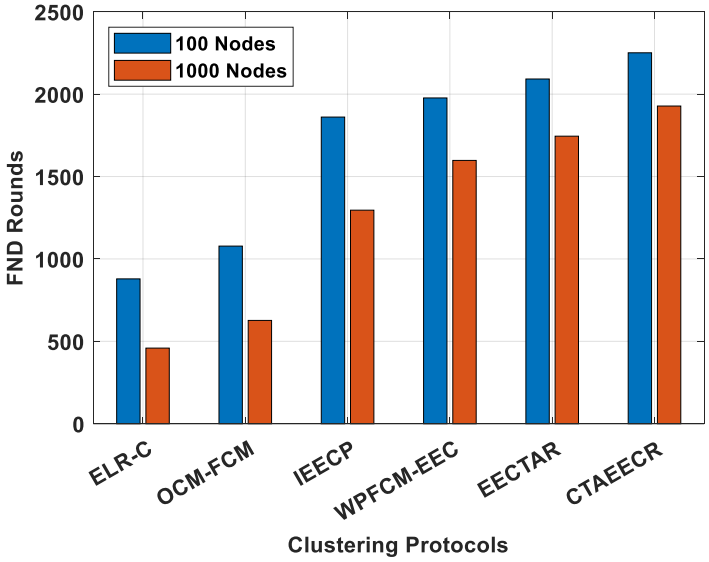

Figure 3: FND comparison of clustering protocols

For 100 nodes, the CTAEECR protocol achieves the highest First Node Death (FND) of 2251 rounds, whereas ELR-C, OCM-FCM, IEECP, WPFCM-EEC, and EECTAR achieve lower FND values of 879, 1078, 1861, 1977, and 2092, respectively (Table 2). For 1000 nodes, the CTAEECR protocol again demonstrates the highest FND of 1928 rounds, compared to 459 rounds (ELR-C), 627 rounds (OCM-FCM), 1296 rounds (IEECP), 1598 rounds (WPFCM-EEC), and 1745 rounds (EECTAR) (Table 2). Notably, CTAEECR extends the FND by approximately 7.06% over EECTAR for 100 nodes and 9.49% for 1000 nodes, confirming its superior energy-balancing capability. Half Node Death (HND) analysis with respect to clustering protocols (ELR-C, OCM-FCM, IEECP, WPFCM-EEC, EECTAR, and CTAEECR) is presented in Figure 4. For 100 nodes, the CTAEECR protocol achieves the highest HND of 2789 rounds, whereas ELR-C, OCM-FCM, IEECP, WPFCM-EEC, and EECTAR achieve lower HNDs of 1895, 1786, 2218, 2379, and 2566 rounds, respectively (Table 2). For 1000 nodes, the proposed CTAEECR protocol again outperforms others with the highest HND of 2357 rounds, compared to 1651 rounds (ELR-C), 1489 rounds (OCM-FCM), 1835 rounds (IEECP), 2014 rounds (WPFCM-EEC), and 2160 rounds (EECTAR) (Table 2).

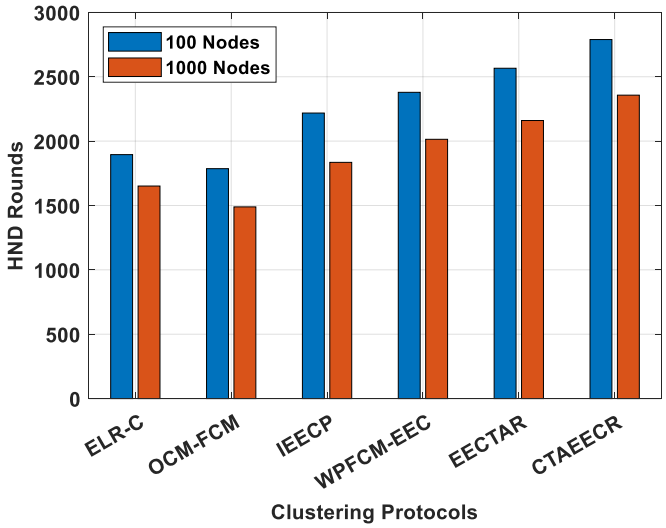

Figure 4: HND comparison of clustering protocols

Notably, CTAEECR improves HND by approximately 7.99% over EECTAR for 100 nodes and 8.35% over EECTAR for 1000 nodes, demonstrating its effectiveness in prolonging network stability.

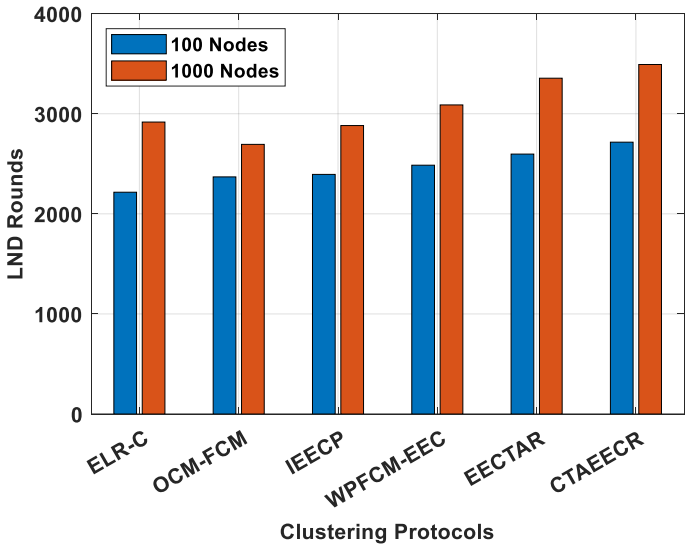

Figure 5: LND comparison of clustering protocols

Last Node Death (LND) comparison with respect to clustering protocols (ELR-C, OCM-FCM, IEECP, WPFCM-EEC, EECTAR, and CTAEECR) is illustrated in Figure 5. For 100 nodes, the CTAEECR protocol achieves the highest LND of 2716 rounds, while ELR-C, OCM-FCM, IEECP, WPFCM-EEC, and EECTAR achieve lower LNDs of 2216, 2369, 2394, 2486, and 2597, respectively (Table 2). For 1000 nodes, the proposed CTAEECR protocol again demonstrates superior performance, achieving the highest LND of 3492 rounds, compared to 2917 (ELR-C), 2694 (OCM-FCM), 2882 (IEECP), 3088 (WPFCM-EEC), and 3355 (EECTAR) (Table 2). Notably, CTAEECR extends LND by approximately 4.38% over EECTAR for 100 nodes and 3.92% for 1000 nodes, confirming its effectiveness in prolonging network lifetime until the final node death.

Weighted First Node Death (WFND) comparison with respect to clustering protocols (ELR-C, OCM-FCM, IEECP, WPFCM-EEC, EECTAR, and CTAEECR) is illustrated in Figure 6. For 100 nodes, the CTAEECR protocol achieves the highest WFND of 4.840, while ELR-C, OCM-FCM, IEECP, WPFCM-EEC, and EECTAR achieve WFND values of 0.657, 0.835, 3.491, 3.884, and 4.142, respectively (Table 2). For 1000 nodes, the proposed CTAEECR protocol again achieves the highest WFND of 1.233, compared to 0.187 (ELR-C), 0.303 (OCM-FCM), 0.817 (IEECP), 1.072 (WPFCM-EEC), and 1.084 (EECTAR) (Table 2). Notably, CTAEECR improves WFND by approximately 14.42% over EECTAR for 100 nodes and 12.08% over EECTAR for 1000 nodes.

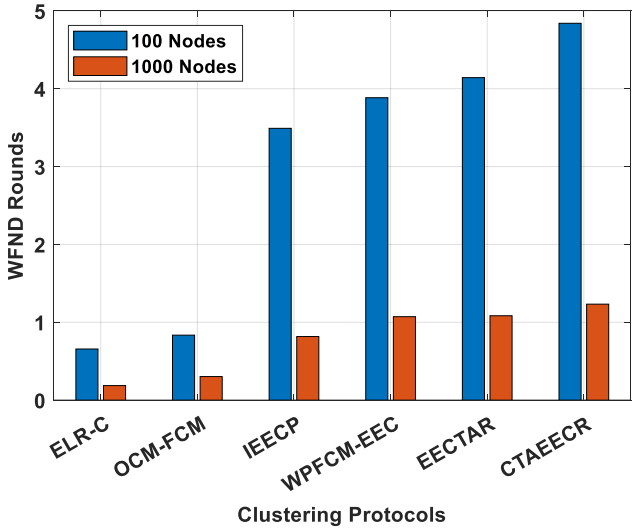

Figure 6: WFND comparison of clustering protocols

Since the network includes resource-constrained nodes, energy efficiency is a critical factor, and the proposed OPOA mechanism effectively addresses it.

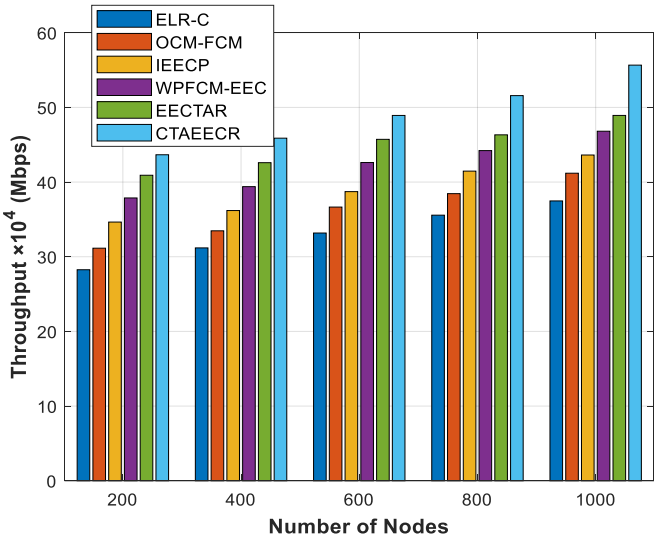

Figure 7: Throughput comparison of clustering protocols

A throughput comparison among clustering protocols (ELR-C, OCM-FCM, IEECP, WPFCM-EEC, EECTAR, and CTAEECR) is illustrated in Figure 7.

Table 3: Throughput comparison of clustering protocols

THROUGHPUT *10^4(Mbps)					
Protocols/No. of Nodes	**200**	**400**	**600**	**800**	**1000**
ELR-C	28.258	31.184	33.178	35.569	37.475
OCM-FCM	31.147	33.471	36.654	38.448	41.187
IEECP	34.646	36.187	38.719	41.466	43.624
WPFCM-EEC	37.874	39.386	42.617	44.214	46.811
EECTAR	40.915	42.597	45.723	46.325	48.926
CTAEECR	43.657	45.879	48.924	51.578	55.661

The proposed protocol achieves the highest result of 55.661*104 Mbps, while ELR-C, OCM-FCM, IEECP, WPFCM-EEC, and EECTAR achieve the lowest results of 37.475*104 Mbps, 41.187*104 Mbps, 43.624*104 Mbps, 46.811*104 Mbps, and 48.926*104 Mbps, respectively, for 1000 nodes (Table 3).

Table 4: Average energy consumption comparison of clustering protocols

Average Energy Consumption (mJ)					
Protocols/No. of Nodes	**200**	**400**	**600**	**800**	**1000**
ELR-C	0.223	0.244	0.291	0.366	0.438
OCM-FCM	0.205	0.228	0.265	0.351	0.404
IEECP	0.185	0.211	0.246	0.315	0.387
WPFCM-EEC	0.163	0.204	0.228	0.262	0.355
EECTAR	0.142	0.173	0.207	0.231	0.326
CTAEECR	0.125	0.154	0.176	0.207	0.292

Average energy consumption comparison with respect to clustering protocols (ELR-C, OCM-FCM, IEECP, WPFCM-EEC, EECTAR, and CTAEECR) is illustrated in Figure 8. For 1000 nodes, the proposed CTAEECR protocol demonstrates the lowest energy consumption of 0.292 mJ.

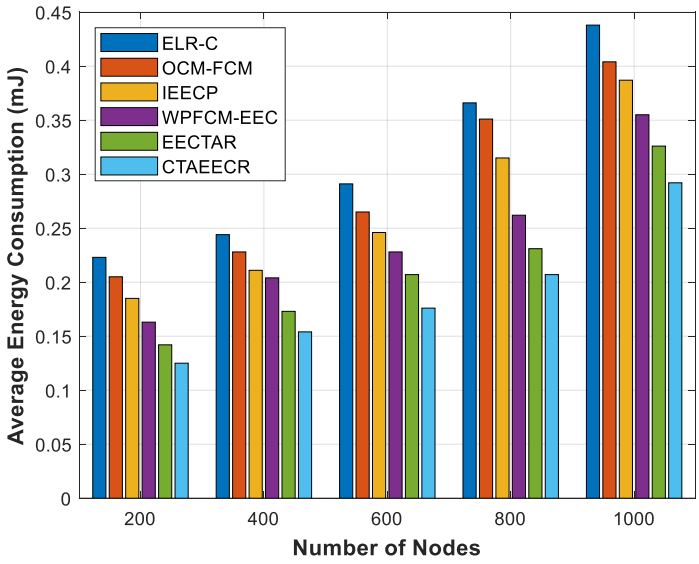

Figure 8: Average energy consumption of clustering protocols

In contrast, ELR-C, OCM-FCM, IEECP, WPFCM-EEC, and EECTAR require higher consumptions of 0.438 mJ, 0.404 mJ, 0.387 mJ, 0.355 mJ, and 0.326 mJ, respectively (Table 4). Notably, compared to the next-best protocol, EECTAR (0.326 mJ), the CTAEECR protocol achieves an additional ~10.40% reduction in energy consumption, highlighting its superior energy efficiency.

Table 5: Delay comparison vs Trust routing protocols

No. of Nodes	**Delay (s)**					
	TBSEER	**SEAMHR**	**CTRF**	**Crow Whale -ETR**	**EECTAR**	**CTAEECR**
200	0.0085	0.0069	0.0055	0.0043	0.0032	0.0025
400	0.0108	0.0091	0.0073	0.0057	0.004	0.0034
600	0.0133	0.0112	0.0094	0.0073	0.0055	0.0046
800	0.0158	0.0136	0.0112	0.0094	0.0072	0.0057
1000	0.0185	0.0168	0.0136	0.0114	0.0095	0.0078

A comparison of delay among trust-based routing techniques (TBSEER, SEAMHR, CTRF, CrowWhale-ETR, EECTAR, and CTAEECR) is illustrated in Figure 9.

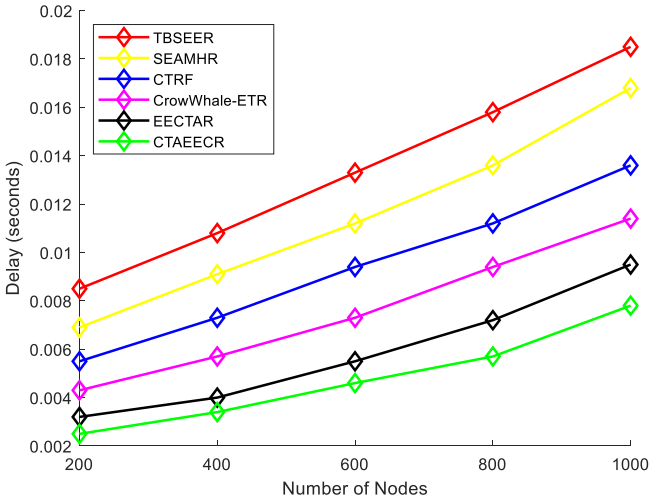

Figure 9: Delay comparison of trust routing protocols

For 1000 malicious sensor nodes, the proposed CTAEECR model achieves the lowest delay of 0.0078 seconds. In contrast, TBSEER, SEAMHR, CTRF, CrowWhale-ETR, and EECTAR incur higher delays of 0.0185, 0.0168, 0.0136, 0.0114, and 0.0095 seconds, respectively (Table 7).

Table 6: PDR comparison vs Trust routing protocols

No. of Nodes	PDR					
	TBSEER	SEAMHR	CTRF	Crow Whale-ETR	EECTAR	CTAEECR
200	0.8575	0.8754	0.8907	0.9115	0.9263	0.9554
400	0.8319	0.8505	0.8681	0.8892	0.9149	0.9415
600	0.8186	0.8342	0.8546	0.8781	0.9046	0.9371
800	0.8094	0.8297	0.8465	0.8676	0.8937	0.9289
1000	0.8072	0.8215	0.8387	0.8564	0.8856	0.9177

Packet Delivery Ratio (PDR) comparison of trust-based routing techniques, including TBSEER, SEAMHR, CTRF, CrowWhale-ETR, EECTA, and CTAEECR, is presented in Figure 10.

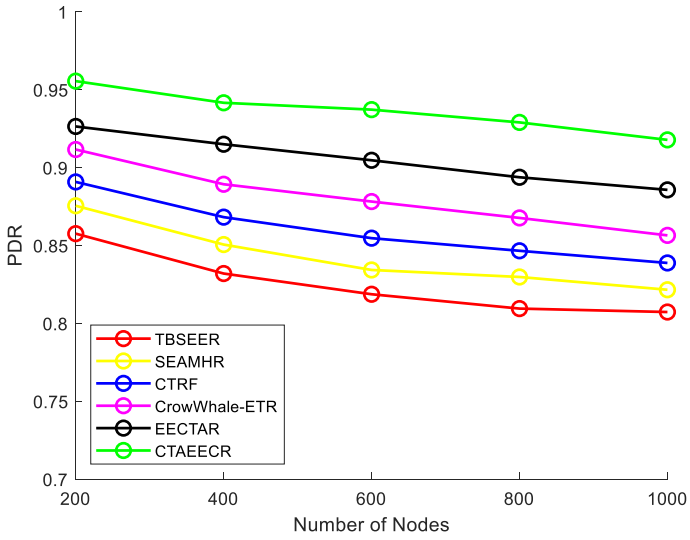

Figure 10: PDR comparison of trust routing protocols

For 200 nodes, the proposed CTAEECR approach achieves the highest PDR of 95.54%, whereas TBSEER, SEAMHR, CTRF, CrowWhale-ETR, and EECTA yield PDRs of 85.75%, 87.54%, 89.07%, 91.15%, and 92.63%, respectively (Table 5). For 1000

nodes, the proposed CTAEECR again demonstrates superior performance with a PDR of 91.77%, compared to 80.72% (TBSEER), 82.15% (SEAMHR), 83.87% (CTRF), 85.64% (CrowWhale-ETR), and 88.56% (EECTA) (Table 6). Notably, CTAEECR improves PDR by 3.21% over EECTA at 200 nodes and by 2.91% at 1000 nodes, underscoring its robustness in maintaining reliable packet delivery even in large-scale networks.

Table 7: PLR comparison vs Trust routing protocols

No. of Nodes	PLR					
	TBSEER	SEAMHR	CTRF	Crow Whale ETR	EECTAR	CTAEECR
200	0.1425	0.1246	0.1093	0.0885	0.0737	0.0446
400	0.1681	0.1495	0.1319	0.1108	0.0851	0.0585
600	0.1814	0.1658	0.1454	0.1219	0.0954	0.0629
800	0.1906	0.1703	0.1535	0.1324	0.1063	0.0711
1000	0.1928	0.1785	0.1613	0.1436	0.1144	0.0823

Packet Loss Ratio (PLR) comparison across trust-based routing techniques (TBSEER, SEAMHR, CTRF, CrowWhale-ETR, EECTAR, and CTAEECR) is illustrated in Figure 11.

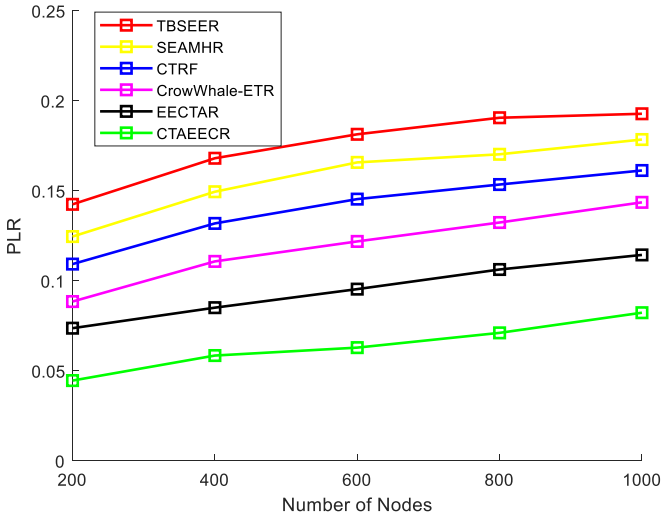

Figure 11: PLR comparison of trust routing protocols

For 200 nodes, the proposed CTAEECR system achieves the lowest PLR of 4.46%, whereas TBSEER, SEAMHR, CTRF, CrowWhale-ETR, and EECTAR report higher PLR values of 14.25%, 12.46%, 10.93%, 8.85%, and 7.37%, respectively. For 1000 nodes, the proposed CTAEECR approach again demonstrates superior performance with a PLR of only 8.23%, compared to 19.28% (TBSEER), 17.85% (SEAMHR), 16.13% (CTRF), 14.36% (CrowWhale-ETR), and 11.44% (EECTAR). Notably, CTAEECR reduces packet loss by approximately 2.91% compared to EECTAR at 200 nodes and 3.21% at 1000 nodes, highlighting its robustness in minimizing communication failures even in dense networks.

5. Conclusion and Future Work

In this paper, the CTAEECR protocol is proposed for the IoT-WSN model to enhance clustering, routing, and congestion control. The protocol integrates multiple techniques to achieve these goals. Firstly, the Weight Possibilistic Fuzzy C-Means (WPFCM) algorithm is employed to form clusters, balancing intra-cluster distance minimisation and energy efficiency. Secondly, Cluster Head (CH) selection and rotation are optimised using the Oppositional Puffer Fish Optimisation Algorithm (OPOA). Each cluster is assigned a Main Cluster Head (MCH) and a Secondary Cluster Head (SCH). Inspired by the pufferfish defence mechanism, Oppositional-Based Learning (OBL) is used to accelerate convergence during the CH election process, thereby ensuring effective data transmission. Thirdly, a Dual Reward Improved Q-Learning (DWIQL) approach is adopted for trust evaluation and updating, which strengthens secure communication. Fourthly, the congestion management system is structured into three phases: (i) intra-cluster and inter-cluster routing, (ii) setup, and (iii) request distribution. This layered design enhances packet delivery reliability while reducing data overhead. The CTAEECR protocol outperforms existing approaches across network lifetime, average energy consumption, throughput, latency, PDR, and PLR. For future work, sleep scheduling mechanisms will be used to further reduce delay, minimise overhead, enhance energy efficiency, prolong network

lifetime, and improve data transmission. Furthermore, the proposed congestion control mechanism will be validated in real-world IoT-WSN scenarios, including smart home systems and agricultural automation.

Acknowledgment: N/A

Data Availability Statement: The datasets generated and analyzed during the current study are available from the corresponding author upon reasonable request, subject to applicable data-sharing guidelines.

Funding Statement: The authors declare that this research was conducted independently and did not receive any specific grant from funding agencies in the public, commercial, or not-for-profit sectors.

Conflicts of Interest Statement: The authors affirm that there are no financial or personal relationships that could have influenced the work reported in this manuscript, and all sources have been properly acknowledged.

Ethics and Consent Statement: The study was conducted in accordance with established ethical standards, with informed consent obtained from all participants and appropriate measures taken to ensure confidentiality and data protection.

References

1. H. Kharrufa, H. A. A. Al-Kashoash, and A. H. Kemp, “RPL-based routing protocols in IoT applications: A review,” *IEEE Sensors Journal*, vol. 19, no. 15, pp. 5952–5967, 2019.
2. J.-W. Lin, P. R. Chelliah, M.-C. Hsu, and J.-X. Hou, "Efficient fault-tolerant routing in IoT wireless sensor networks based on bipartite-flow graph modeling," *IEEE Access*, vol. 7, no. 1, pp. 14022–14034, 2019.
3. A. Ö. Ercan, M. O. Sunay, and I. F. Akyildiz, "RF energy harvesting and transfer for spectrum sharing cellular IoT communications in 5G systems," *IEEE Transactions on Mobile Computing*, vol. 17, no. 7, pp. 1680–1694, 2017.
4. Y. W. Kuo, C. L. Li, J. H. Jhang, and S. Lin, "Design of a wireless sensor network-based IoT platform for wide area and heterogeneous applications," *IEEE Sensors Journal*, vol. 18, no. 12, pp. 5187–5197, 2018.
5. G. Kaur, P. Chanak, and M. Bhattacharya, "Energy-efficient intelligent routing scheme for IoT-enabled WSNs," *IEEE Internet of Things Journal*, vol. 8, no. 14, pp. 11440–11449, 2021.
6. M. Rathee, S. Kumar, A. H. Gandomi, K. Dilip, B. Balusamy, and R. Patan, "Ant colony optimization based quality of service aware energy balancing secure routing algorithm for wireless sensor networks," *IEEE Transactions on Engineering Management*, vol. 68, no. 1, pp. 170–182, 2019.
7. S. Dehghani, B. Barekatain, and M. Pourzaferani, "An enhanced energy-aware cluster-based routing algorithm in wireless sensor networks," *Wireless Personal Communications*, vol. 98, no. 1, pp. 1605–1635, 2018.
8. S. Su and S. Zhao, "An optimal clustering mechanism based on Fuzzy-C means for wireless sensor networks," *Sustainable Computing: Informatics and Systems*, vol. 18, no. 6, pp. 127–134, 2018.
9. S. B. Shah, Z. Chen, F. Yin, I. U. Khan, and N. Ahmad, "Energy and interoperable aware routing for throughput optimization in clustered IoT-wireless sensor networks," *Future Generation Computer Systems*, vol. 81, no. 4, pp. 372–381, 2018.
10. M. Mathapati, T. S. Kumaran, A. Muruganandham, and M. Mathivanan, "Secure routing scheme with multi-dimensional trust evaluation for wireless sensor network," *Journal of Ambient Intelligence and Humanized Computing*, vol. 12, no. 6, pp. 6047–6055, 2021.
11. W. Fang, W. Zhang, W. Chen, Y. Liu, and C. Tang, "TMSRS: Trust management-based secure routing scheme in industrial wireless sensor network with fog computing," *Wireless Networks*, vol. 26, no. 5, pp. 3169–3182, 2020.
12. A. Ahmed, K. A. Bakar, M. I. Channa, and A. W. Khan, "A secure routing protocol with trust and energy awareness for wireless sensor network," *Mobile Networks and Applications*, vol. 21, no. 2, pp. 272–285, 2016.
13. K. A. Awan, I. Din, A. Almogren, M. Guizani, A. Altameem, and S. U. Jadoon, "RobustTrust: A pro-privacy robust distributed trust management mechanism for Internet of Things," *IEEE Access*, vol. 7, no. 5, pp. 62095–62106, 2019.
14. R. W. Anwar, A. Zainal, F. Outay, A. Yasar, and S. Iqbal, "BTEM: Belief-based trust evaluation mechanism for wireless sensor networks," *Future Generation Computer Systems*, vol. 96, no.7, pp. 605–616, 2019.
15. J. Yan and B. Qi, "CARA: A congestion-aware routing algorithm for wireless sensor networks," *Algorithms*, vol. 14, no. 7, p. 199, 2021.
16. P. S. Prakash, D. Kavitha, and P. C. Reddy, "Energy and congestion-aware load balanced multi-path routing for wireless sensor networks in ambient environments," *Computer Communications*, vol. 195, no. 11, pp. 217–226, 2022.
17. M. Farsi, M. Badawy, M. Moustafa, H. A. Ali, and Y. Abdulazeem, "A congestion-aware clustering and routing (CCR) protocol for mitigating congestion in WSN," *IEEE Access*, vol. 7, no. 8, pp. 105402–105419, 2019.
18. J. Yan and B. Qi, "CARA: A congestion-aware routing algorithm for wireless sensor networks," *Algorithms*, vol. 14, no. 7, p. 199, 2021.

19. G. Sangeetha, M. Vijayalakshmi, S. Ganapathy, and A. Kannan, "An improved congestion-aware routing mechanism in sensor networks using fuzzy rule sets," *Peer-to-Peer Networking and Applications*, vol. 13, no. 3, pp. 890–904, 2020.
20. P. Chanak and I. Banerjee, "Congestion-free routing mechanism for IoT-enabled wireless sensor networks for smart healthcare applications," *IEEE Transactions on Consumer Electronics*, vol. 66, no. 3, pp. 223–232, 2020.
21. S. Tumula, Y. Ramadevi, E. Padmalatha, G. K. Kumar, M. V. Gopalachari, L. Abualigah, P. Chithaluru, and M. Kumar, "An opportunistic energy-efficient dynamic self-configuration clustering algorithm in WSN-based IoT networks," *International Journal of Communication Systems*, vol. 37, no. 1, pp. 1–21, 2024.
22. V. Sharma, R. Beniwal, and V. Kumar, "Multi-level trust-based secure and optimal IoT-WSN routing for environmental monitoring applications," *The Journal of Supercomputing*, vol. 80, no. 8, pp. 11338–11381, 2024.
23. D. K. Shende and S. S. Sonavane, "CrowWhale-ETR: CrowWhale optimization algorithm for energy and trust aware multicast routing in WSN for IoT applications," *Wireless Networks*, vol. 26, no. 6, pp. 4011–4029, 2020.
24. H. Hu, Y. Han, M. Yao, and X. Song, "Trust based secure and energy efficient routing protocol for wireless sensor networks," *IEEE Access*, vol. 10, no. 4, pp. 10585–10596, 2021.
25. A. A. H. Hassan, W. M. Shah, A. H. H. Habeb, M. F. I. Othman, and M. N. Al-Mhiqani, "An improved energy-efficient clustering protocol to prolong the lifetime of the WSN-based IoT," *IEEE Access*, vol. 8, no. 11, pp. 200500–200517, 2020.
26. A. F. Raslan, A. F. Ali, A. Darwish, and H. M. El-Sherbiny, "An improved sunflower optimization algorithm for cluster head selection in the Internet of Things," *IEEE Access*, vol. 9, no. 11. pp. 156171–156186, 2021.
27. M. Rizwanullah, H. Alsolai, M. K. Nour, A. S. A. Aziz, M. I. Eldesouki, and A. A. Abdelmageed, "Hybrid muddy soil fish optimization-based energy aware routing in IoT-assisted wireless sensor networks," *Sustainability*, vol. 15, no. 10, pp. 1–15, 2023.
28. A. Srivastava and R. Paulus, "ELR-C: A multi-objective optimization for joint energy and lifetime aware cluster-based routing for WSN-assisted IoT," *Wireless Personal Communications*, vol. 132, no. 2, pp. 979–1006, 2023.
29. H. Mostafaei, "Energy-efficient algorithm for reliable routing of wireless sensor networks," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 7, pp. 5567–5575, 2019.
30. S. Augustine and J. P. Ananth, "Taylor kernel fuzzy C-means clustering algorithm for trust and energy-aware cluster head selection in wireless sensor networks," *Wireless Networks*, vol. 26, no. 7, pp. 5113–5132, 2020.