Received: xxx **Accepted:** xxx **Published:** xxx.
DOI. xxxxxxx
xxx Volume xxx, Issue xxx, (1-31)
Research Article

Open Access

Control and Optimization in Applied Mathematics - COAM

A New Energy-Efficient Clustering in Wireless Sensor Networks Using an Adaptive Fuzzy Neural Network Approach

Ahmad Jalili[1] ✉, Fatemeh Babakordi[2]

[1]Department of Computer Engineering, Faculty of Basic Sciences and Engineering, Gonbad Kavous University, Gonbad Kavous, Gonbad Kavous, Iran.
[2]Department of Mathematics and Statistics, Gonbad Kavous University, Gonbad Kavous, Iran.

✉ **Correspondence:**
Ahmad Jalili
E-mail:
jalili@gonbad.ac.ir

How to Cite
Jalili, A., Babakordi, F. (2025). "A new energy-efficient clustering in wireless sensor networks using adaptive fuzzy neural network approach", Control and Optimization in Applied Mathematics, 10(): 1-31, doi: 10.30473/coam.2025.74415.1303.

Abstract. Energy constraint is the most critical challenge in Wireless Sensor Networks (WSNs), particularly in dynamic environments with mobile nodes. This paper proposes an intelligent clustering protocol based on Fuzzy Neural Networks (FNN) that adaptively optimizes energy consumption by dynamically selecting cluster heads and determining optimal cluster configurations. The FNN integrates fuzzy logic's uncertainty handling with neural networks' learning capabilities, using key parameters including residual energy, node distance, neighbor density, and signal-to-noise ratio. Unlike static clustering approaches such as LEACH and HEED, our method continuously adapts to changing network conditions through real-time parameter evaluation. Extensive MATLAB simulations with 100 nodes demonstrate significant performance improvements: the proposed FNN extends network lifetime by 35% compared to LEACH, 28% compared to HEED, and 15% compared to ANN-based ELDC. The First Node Dies (FND) is delayed by 45%, 38%, and 22% respectively, while achieving 25% lower energy consumption. Results confirm the FNN approach's superior energy efficiency and network stability, making it highly suitable for dynamic WSN applications.

Keywords. Wireless sensor networks, Energy efficiency, Clustering, Fuzzy neural networks, Intelligent systems

MSC. 03B53; 93C42.

https://mathco.journals.pnu.ac.ir

1 Introduction

Wireless Sensor Networks (WSNs) represent a cornerstone technology in the modern era of ubiquitous computing and the Internet of Things (IoT) [19]. Comprising numerous spatially distributed, autonomous sensor nodes, WSNs facilitate unprecedented capabilities for monitoring physical or environmental conditions, with applications spanning from precision agriculture and industrial automation to critical healthcare and military surveillance [22, 26]. However, the practical deployment and operational longevity of WSNs are fundamentally constrained by the limited energy resources of their individual sensor nodes. These nodes typically rely on small, irreplaceable batteries, making energy conservation the most critical design consideration to maximize the network's functional lifetime [14].

Among the various strategies developed to mitigate energy depletion, hierarchical clustering has been widely recognized as a highly effective technique for enhancing both energy efficiency and network scalability [18]. Clustering protocols organize nodes into groups (clusters), each electing a Cluster Head (CH). Non-CH nodes then transmit their data over potentially shorter distances to their local CH, which aggregates the collected information, reducing redundancy, and forwards it to a distant Base Station (BS). This hierarchical structure significantly reduces overall energy consumption compared to direct transmission. However, many seminal and conventional clustering protocols, such as LEACH and HEED [12, 29], were designed with static or slowly changing networks in mind. Their reliance on simplified probabilistic models or fixed, predefined parameters for CH selection renders them ill-equipped to handle the challenges of dynamic environments. In many real-world scenarios, sensor nodes may be mobile, channel conditions can fluctuate, and data generation rates may vary [15]. In such dynamic settings, the rigid nature of conventional protocols leads to suboptimal cluster formation, imbalanced load distribution, and the premature death of critical nodes, ultimately degrading network coverage and shortening its lifespan. This gap between the requirements of dynamic applications and the capabilities of existing static protocols constitutes the central problem this research aims to solve [7].

The clear limitations of traditional protocols necessitate a move towards more intelligent and adaptive mechanisms. Machine learning offers a promising avenue for developing such solutions. Approaches based on Artificial Neural Networks (ANNs) [26], for instance, can learn complex, non-linear relationships between network state parameters (like energy and distance) and optimal clustering decisions. However, a purely ANN-based approach is not without its own challenges in the WSN context. ANNs can be sensitive to the noisy and imprecise data inherent in sensor readings and wireless communications. Furthermore, they operate as "black boxes," which can struggle with the conceptual, linguistic-style reasoning that is often intuitive for describing network states (e.g., a node's energy is "low," or its distance to the BS is "far"). It is this specific limitation that motivates our proposed solution. We identify an opportunity to create a more robust and effective model by synergizing the adaptive learning strengths of neural networks with the uncertainty-handling capabilities of fuzzy logic. Fuzzy logic excels at modeling and reasoning with imprecise information and linguistic variables. By integrating these two paradigms, we can construct a Fuzzy Neural Network (FNN) that not only learns from data but also embraces and manages uncertainty [7, 31]. This hybrid approach is uniquely suited to build an intelligent clustering mechanism that is both adaptive and resilient, capable of making nuanced, context-aware decisions in the face of real-world network dynamics and data imprecision [6].

To address the challenges outlined above, this paper introduces a new energy-efficient clustering protocol for WSNs based on a dynamic Fuzzy Neural Network. We propose a comprehensive framework that leverages the FNN to intelligently guide the clustering process, significantly extending network lifetime and stability. The primary contributions of this work are:

- *Design of a Novel FNN-based Clustering Protocol:* We introduce an intelligent clustering mechanism where an FNN adaptively determines CH suitability. The FNN is designed to process multiple, often conflicting, real-time network parameters—including residual energy, distance to the base station, local node density, and signal-to-noise ratio (SNR)—to make holistic and informed decisions.

- *A Practical Framework for WSN Deployment:* We propose and validate a two-phase operational framework. An intensive, one-time offline training phase builds the intelligent FNN model without consuming any node energy. This is followed by a lightweight online operational phase where each node can efficiently use the trained model for distributed, round-by-round decision-making, confirming the feasibility of our approach for resource-constrained sensor nodes.

- *Comprehensive Performance Validation:* We conduct extensive simulations to rigorously evaluate our proposed protocol. The results provide definitive evidence that our FNN-based approach demonstrably outperforms not only the benchmark legacy protocols (LEACH and HEED) but also a baseline ANN-based clustering model (ELDC) across all key metrics, including network stability period, total lifetime, and overall energy efficiency. This empirically confirms the tangible benefits of integrating fuzzy logic for enhanced robustness and performance.

The remainder of this paper is organized as follows: Section 2 reviews related work. Section 3 details the proposed FNN-based clustering methodology. Section 4 describes the simulation setup and evaluation metrics. Section 5 presents and discusses the simulation results. Scalability and Complexity Analysis are presented in Section 6. Overhead Comparison Summary is discussed in Section 7. Finally, Section 8 concludes the paper and suggests future research directions.

2 Related Work

Energy efficiency remains a primary concern in WSNs, leading to the development of numerous clustering and routing protocols. Existing works can be categorized into conventional clustering, machine learning-based clustering, and bio-inspired/metaheuristic clustering techniques.

2.1 Conventional Clustering Protocols

- *LEACH (Low-Energy Adaptive Clustering Hierarchy):* A seminal distributed clustering protocol where nodes probabilistically elect themselves as CHs based on a target percentage and rotate the CH role to distribute energy load [12, 29]. While simple, it doesn't guarantee optimal CH distribution or number. LEACH-C (Centralized) uses the BS for CH selection based on node energy and location [9].

- *HEED (Hybrid Energy-Efficient Distributed Clustering):* Improves upon LEACH by using residual energy as the primary parameter for CH selection and intra-cluster communication cost (e.g., node degree) as a secondary parameter, aiming for better CH distribution and prolonged lifetime [16, 29].

- *PEGASIS (Power-Efficient Gathering in Sensor Information Systems):* Forms a chain among sensor nodes instead of clusters. Each node transmits only to its neighbor, and a designated leader transmits aggregated data to the BS [11, 17]. Reduces overhead but increases delay and is sensitive to node failures.

- *TEEN (Threshold sensitive Energy Efficient sensor Network protocol):* Designed for reactive networks, reducing transmissions by only reporting data when sensed values cross predefined hard and soft thresholds. APTEEN adapts TEEN for periodic reporting alongside reactive monitoring [4, 23].

- *VGA (Virtual Grid Architecture routing):* A location-aware protocol that partitions the network into fixed zones, selecting a local and master aggregator for data fusion [19, 1]. Reduces energy but assumes location awareness.

- *Other Protocols:* Numerous variations exist, such as CBHRP (Cluster Based Hierarchical Routing Protocol) [8, 21], LLACA (adaptive localized clustering scheme named localized learning automata-based clustering algorithm) [9, 24], GAF (Geographic Adaptive Fidelity) [5, 10], and GEAR (Geographic and Energy Aware Routing) [25, 30], each addressing specific aspects like mobility, localization, or data delivery quality.

2.2 Learning-Based Clustering

Machine learning offers adaptive solutions.

- *ANN-based Clustering (e.g., ELDC):* As presented in [26], ANNs can be trained to learn the relationship between network state (node energy, location, etc.) and optimal clustering parameters (e.g., number of clusters). Backpropagation is typically used for training. The inputs often include residual energy, distance to BS, distance to cluster boundary, distance to CH, number of neighbors, and SNR [18, 26]. While effective, ANNs might lack robustness to uncertainty and the interpretability of fuzzy systems.
- *Ant Colony Optimization (ACO):* ACO algorithms have been applied to WSN routing [21], including energy-aware variants like EEABR [13]. These bio-inspired methods explore paths based on pheromone trails representing path quality (e.g., lower energy cost), but can have slow convergence.
- *Fuzzy Logic:* Fuzzy logic has been used in WSNs for CH selection (e.g., using energy, centrality, distance) [3, 7] and routing, leveraging its ability to handle imprecise data and make rule-based decisions.

2.3 Bio-Inspired and Metaheuristic Optimization Approaches

Recently, several metaheuristic optimization algorithms have been applied to clustering in WSNs to enhance energy efficiency:

- *Cuckoo Optimization Algorithm (COA):* In [20], a COA-based energy-aware clustering protocol was proposed. COA mimics the brood parasitism behavior of cuckoos to optimize CH selection. While effective in reducing energy consumption, COA involves high computational complexity and may not scale well in real-time WSN operations.
- *Black Hole and Ant Colony Algorithms (BH-ACO):* The work in [28] combined black hole and ant colony optimization to design an adaptive cluster-based data transmission scheme. This hybrid approach improved energy balancing and routing efficiency. However, its reliance on iterative path optimization makes it less suitable for networks with frequent topology changes.
- *Firefly Algorithm (FA):* In [27], the firefly algorithm was used for data aggregation and clustering. FA exploits swarm intelligence to minimize redundant transmissions, leading to reduced energy consumption. Nevertheless, FA-based clustering still suffers from parameter tuning challenges and convergence delays in large-scale WSNs.

These metaheuristic approaches demonstrate promising energy efficiency improvements, but they often require significant computational overhead, making them less practical for resource-constrained sensor nodes. These algorithms are typically iterative and may require substantial computation to converge to a solution in each round. This can be a considerable drawback in highly dynamic WSNs that require rapid, low-latency re-clustering.

2.4 Gap Analysis

While these works have established the potential of neuro-fuzzy systems, a critical review reveals specific gaps that our research aims to address. Firstly, many existing approaches focus solely on the CH selection problem for a predefined or heuristically determined number of clusters (k). They do not address the equally important problem of dynamically determining the optimal number of clusters based on the network's current state, which is crucial for true load balancing and energy efficiency.

Secondly, the practical deployment model is often not fully detailed. Our work proposes a clear two-phase framework, decoupling the high-cost offline training from the low-cost online execution, which is a critical consideration for demonstrating real-world feasibility on resource-constrained sensor nodes. Finally, few studies provide a direct, empirical comparison against a pure ANN baseline to specifically isolate and validate the contribution of the fuzzy logic component in enhancing robustness and performance. Our research directly addresses these three aspects.

3 Proposed Methodology: FNN for Energy-Efficient Clustering

The core objective of our methodology is to dynamically optimize the clustering process in WSNs to significantly reduce energy consumption and prolong network operational lifetime. We propose an intelligent approach based on a FNN, which synergizes the adaptive learning capabilities of ANNs with the uncertainty handling and linguistic reasoning strengths of Fuzzy Logic [2]. This allows the system to make robust and context-aware decisions about cluster formation and Cluster Head (CH) selection in each round, adapting to the potentially changing conditions of the network, including variations in node energy and topology.

The proposed FNN adapts its clustering decisions at every communication round by evaluating real-time parameters (residual energy, neighbor density, SNR, distance to BS). This per-round decision-making allows the protocol to handle typical gradual network changes. However, since the FNN was trained offline, it assumes static or low-mobility networks. Sudden large-scale topology changes, such as rapid mobility or simultaneous mass node failures, are not explicitly addressed in this study.

3.1 System Model and Energy Consumption

We adopt a standard WSN model consistent with related works [14, 15]:

- A set of N sensor nodes is deployed (initially randomly) within a defined sensing field (e.g., $100m \times 100m$).
- A stationary Base Station (BS) resides at a known location (e.g., the center (50, 50)).
- Each node begins with an initial energy, E_0, (e.g., 0.5 J).
- Nodes can adjust their transmission power based on distance.
- The first-order radio energy model [12] is used:
  - Energy to run transmitter/receiver circuitry: E_{elec} (e.g., 0.5 nJ/bit).
  - Energy for data aggregation: E_{DA} (e.g., 0.5 nJ/bit/signal).
  - Amplifier energy depends on distance d:
    - $E_{fs}.d^2$ if $d < d_0$ (free space),
    - $E_{mp}.d^4$ if $d \geq d_0$ (multipath).

E_{TX} shows Transmission Energy, the energy consumed by a node transmitting a k-bit message over distance d is:

$$E_{TX}(k.d) = E_{\text{elec}} \cdot k + \begin{cases} E_{fs} \cdot k \cdot d^2, & \text{if} \quad d < d_0, \\ E_{mp} \cdot k \cdot d^4, & \text{if} \quad d \geq d_0, \end{cases} \tag{1}$$

where E_{elec} is the energy dissipated per bit to run the transmitter or receiver circuitry, and E_{fs} or E_{mp} are the amplifier energy costs depending on the distance $d$relative to the threshold distance d_0. This equation shows that energy consumption grows quadratically with distance for short links (free space) but quartically for long links (multipath). Therefore, minimizing the transmission distance (via clustering) significantly reduces energy consumption.

Reception Energy $(E_{RX}(k))$, the energy consumed receiving a k-bit message is:

$$E_{RX}(k) = E_{\text{elec}} \cdot k. \tag{2}$$

Unlike transmission, reception energy depends only on the number of bits and not on the distance. The energy for reception depends only on the number of bits k and the electronics energy E_{elec}. Cluster Heads also expend energy for aggregating data received from cluster members, calculated as E_{DA} per bit per aggregation signal, data aggregation energy (E_{DA}). Minimizing the total energy consumed across all nodes per round is the primary goal.

Cluster Heads (CHs) aggregate data from their member nodes to reduce redundancy before forwarding it to the Base Station. The energy spent for aggregating k bits is:

$$E_{DA}(k) = E_{DA} \cdot k, \tag{3}$$

where E_{DA} is the energy required per bit for data aggregation (e.g., 5 nJ/bit). Data aggregation significantly lowers the volume of transmitted data, saving energy at the network level.

The total energy consumed in a round is the sum of transmission, reception, and aggregation energy across all nodes:

$$E_{round} = \sum_{i=1}^{N} \left(E^{i}_{TX} + E^{i}_{RX} + E^{i}_{DA}\right), \tag{4}$$

where NNN is the number of nodes. The clustering protocol's goal is to minimize E_{round} to prolong network lifetime.

The proposed FNN-based clustering protocol uses these Equations (1) to (4) indirectly to guide its learning and decision-making:

- The inputs (residual energy, distance, neighbor density, SNR) relate to the components of these formulas.
- The FNN learns to minimize total energy per round by selecting optimal CHs and forming energy-efficient clusters.

3.2 The FNN Architecture

The intelligence of our clustering mechanism resides in the FNN. It processes real-time network information to output decisions guiding the clustering process. Our FNN likely follows a structure similar to ANFIS (Adaptive Neuro-Fuzzy Inference System), comprising multiple layers, as conceptually shown in Figure 1.

The proposed FNN integrates the fuzzy inference system with neural network learning, forming a layered architecture analogous to an Adaptive Neuro-Fuzzy Inference System (ANFIS). The FNN comprises five layers: Input, Fuzzification, Rule, Normalization, and Output. Figure 1 illustrates the interconnections among these layers: the Input Layer accepts the network parameters, the Fuzzification Layer employs Membership Functions, the Rule Layer encodes IF-THEN rules, the Normalization/Consequent Layer performs normalization and combines rule outputs, and the Output Layer producing a crisp result (e.g., CH Suitability Score).

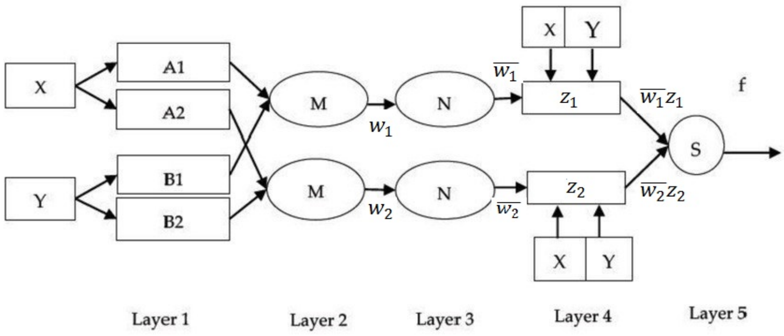

Figure 1: Conceptual layered architecture of the proposed FNN.

- **Layer 1 (Input Layer):** The input layer accepts crisp numerical values that represent the current state of a node and its surrounding environment. Drawing on factors known to influence energy-efficient clustering and the inputs employed in the baseline work of the cited literature [14, 26], we identify the following critical parameters (summarized in Table 1): Residual Energy, Distance to Base Station, Node Degree, Average Distance to Neighbors, and Signal-to-Noise Ratio.
  Inputs: $\mathbf{x} = [E_r, d_{BS}, ND, SNR]$, where:

  – E_r = residual energy

  – d_{BS} = distance to base station

  – ND = node degree

  – SNR = signal-to-noise ratio

  The output of Layer 1 is $O_1^i = x_i$.

- **Layer 2 (Fuzzification Layer):** Each input node from Layer 1 connects to nodes in this layer that represent fuzzy sets. This layer converts the crisp input values into fuzzy degrees of membership using Membership Functions (MFs). In this paper, we employ *Triangular MFs* due to their simplicity and computational efficiency. For instance, 'Residual Energy' might be mapped to fuzzy sets like 'Low', 'Medium', and 'High', each with its corresponding triangular MF, as illustrated conceptually in Figure 2. The parameters defining these MFs (e.g., the vertices of the triangles) are adapted during the FNN's training phase.

  - Each input x_i is fuzzified into linguistic terms (e.g., Low, Medium, and High) using the Triangular MFs:

$$\mu_A(x) = \begin{cases} 0, & x \le a, \\ \frac{x-a}{b-a}, & a < x \le b, \\ \frac{c-x}{c-b}, & b < x < c, \\ 0, & x \ge c. \end{cases}$$

- Here (a, b, c) define the triangle vertices for each fuzzy set.

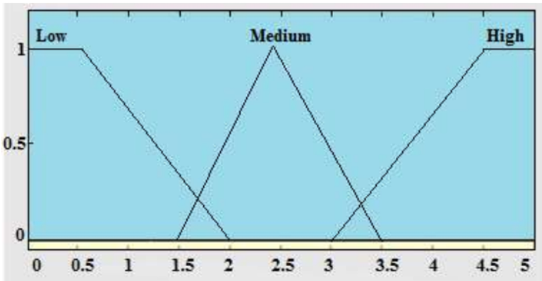

Figure 2: Triangular membership functions for 'residual energy'.

Figure 2 shows the input range for Residual Energy on the x-axis and Degree of Membership (0 to 1) on the y-axis. Three overlapping triangles representing 'Low', 'Medium', and 'High' energy levels are shown.

- **Layer 3 (Rule Layer):** This layer embodies the fuzzy inference engine. Each node in this layer corresponds to a fuzzy IF-THEN rule. These rules capture heuristic knowledge about desirable clustering characteristics. Example rules might be:

  - **Rule 1**: If E_{res} is High and D_{BS} is Low and ND is Medium, then $\text{CH}_{\text{Suitability}}$ is Very_High.
  - **Rule 2**: If E_{res} is High, then $\text{CH}_{\text{Suitability}}$ is Very_Low.
  - **Rule 3**: If ND is High and D_N is Low, then $\text{CH}_{\text{Suitability}}$ is High.

  The firing strength of each rule is calculated in this layer, typically using a fuzzy AND operator (T-norm). The output of each rule node represents how strongly that rule applies given the current fuzzified inputs.

  Firing strength of rule j:

$$w_j = \prod_i \mu_{A_i^j}(x_i).$$

- **Layer 4 (Normalization/Consequent Layer):** This layer aggregates the outcomes of the activated fuzzy rules. It often involves normalizing the firing strengths calculated in Layer 3 and applying these weights to the consequent part of each rule (e.g., the fuzzy set representing 'Very_High' suitability). The specific operations depend on the FNN model but aim to combine the contributions of all relevant rules.

- Firing strengths are normalized:

$$\overline{w}_j = \frac{w_j}{\sum_j w_j}.$$

- Consequent functions $f_j(x)$ (linear in inputs) are applied:

$$O_4^j = \overline{w}_j \times f_j(x) = \overline{w}_j \left(p_j E_r + q_j d_{BS} + r_j ND + s_j SNR + t_j\right).$$

- **Layer 5 (Defuzzification/Output Layer):** This layer converts the aggregated fuzzy output from Layer 4 back into a single, crisp numerical value. The paper indicates the use of the *Centroid (Center of Gravity)* or *Weighted Average of Centers* defuzzification methods, which are widely used and effective in practice, as conceptually illustrated in Figure 3.

The final output, denoted as the CH suitability score y), is computed as:

$$y = \sum_j O_4^j,$$

representing the crisp decision of the FNN. For example, y may indicate the node's overall suitability to act as a CH in the current round, or potentially provide an estimation of the optimal number of clusters (k) for the network to target.

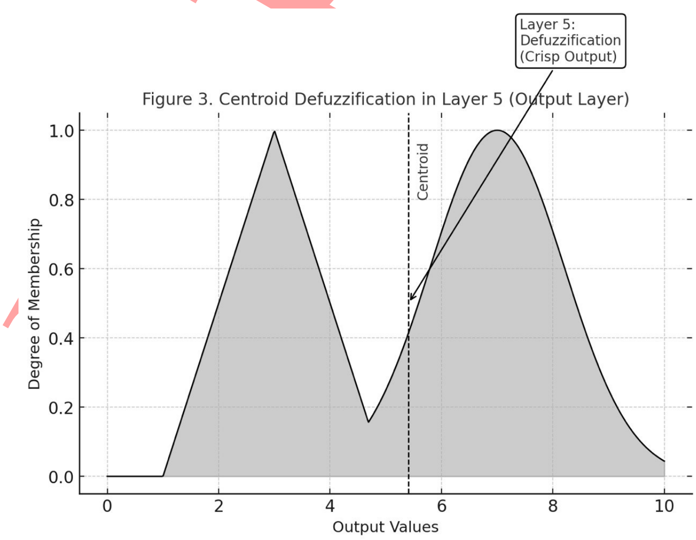

Figure 3: Illustration of centroid defuzzification.

Figure 3 presents the centroid defuzzification process. The horizontal axis x corresponds to output values, while the vertical axis y indicates their associated degrees of membership. The figure also depicts the centroid method used to derive a single crisp value.

3.3 Input Parameters

The choice of input parameters is critical for the effectiveness of FNN. Table 1 summarizes the rationale for the selected inputs.

Table 1: FNN Input parameters and rationale.

Parameter	**Symbol**	**Rationale for Energy-Efficient Clustering**	**Description**
Residual Energy	E_{res}	Nodes with higher energy are preferable CHs.	The current energy level of the node. Crucial for node survival.
Distance to Base Station	D_{BS}	Shorter distance to BS reduces transmission cost. (CHs closer to the BS require less energy for long-haul transmission, saving significant energy.)	Affects the energy required for CH-to-BS transmission.
Node Degree	ND	Indicates local node density and clustering potential. (Reflects local density; high degree may imply good data aggregation potential but higher CH load.)	Number of nodes within communication range; indicates local density.
Average Distance to Neighbors	D_N	Reflects intra-cluster communication efficiency. (Influences intra-cluster communication energy.)	Influences potential intra-cluster communication costs; lower D_N suggests more compact clustering.
Signal-to-Noise Ratio	SNR	High SNR implies better communication reliability. (Indicates channel quality; nodes with better links may be preferred for reliable transmission.)	Reflects channel quality, impacting transmission reliability and required power.

The FNN learns to weigh and combine these potentially conflicting factors (e.g., high energy but far from BS) using its fuzzy rules and learned parameters.

3.4 The FNN training and dynamic operational flow

The proposed FNN-based clustering mechanism operates in two distinct phases: an offline training phase where the FNN learns the optimal clustering behavior, and an online operational phase where the trained FNN dynamically guides the clustering process in each communication round. Algorithm 1 shows details of pseudocode for FNN-Based Dynamic Clustering.

Algorithm 1 Pseudocode for FNN-based dynamic clustering

```
// Phase 1: Offline (Executed once)
Function Train_FNN_Model():
TrainingData = GenerateTrainingData() // Input vectors & Target outputs
FNN = InitializeFNN()
Loop until stopping criterion met:
Error = CalculateFNNError(FNN, TrainingData)
AdjustFNNParameters(FNN, TrainingData, LearningRate)
End Loop
Save TrainedFNN
End Function
// Phase 2: Online (Executed each round)
Function Run_FNN_Clustering_Round(round_r, Nodes, TrainedFNN, BS):
// Step 1 & 2: Status Acquisition & FNN Evaluation
For each Node n in Nodes:
n.status = GetNodeStatus(n)
n.ch_score = EvaluateFNN(TrainedFNN, n.status)
End For
// Step 3: CH Selection (Simplified Example: Threshold & Target k)
k_optimal = DetermineOptimalK(Nodes) // Optional FNN output or heuristic
CandidateCHs = FilterNodesByScoreThreshold(Nodes)
SelectedCHs = ResolveContention(CandidateCHs, k_optimal)
For each Node n in Nodes:
n.role = (n in SelectedCHs) ? CH : NonCH
End For
// Step 4 & 5: Cluster Formation
BroadcastCHAdvertisements(SelectedCHs)
For each Node m where m.role == NonCH:
m.chosen_ch = FindBestCH(m, ReceivedAds)
SendJoinRequest(m, m.chosen_ch)
End For
// Step 5: TDMA Scheduling
For each Node c where c.role == CH:
c.schedule = CreateTDMASchedule(c.member_nodes) BroadcastSchedule(c, c.schedule)
End For
// Step 6: Steady-State Data Transmission
RunSteadyStatePhase(Nodes, BS) // Includes intra & inter-cluster transmission End Function
// Main Loop
Load TrainedFNN
InitializeNodes()
For r = 1 to MAX_ROUNDS:
If NumberOfAliveNodes > 0:
Run_FNN_Clustering_Round(r, AliveNodes, TrainedFNN, BS)
Else:
Break // Network dead
End If End For =0
```

1. **Offline Training:** The objective of the offline phase is to train the FNN parameters (Membership Functions, rule consequents/weights depending on the specific FNN type like ANFIS) to accurately map network state inputs to desired clustering outputs.

    1. *Data Generation:* A diverse dataset is crucial for effective training. This dataset consists of input-output pairs. Training data is generated by simulating various network scenarios (different node distributions, energy levels). For each scenario, an optimal or near-optimal clustering configuration (e.g., determined by running LEACH [12] and selecting its best outcomes, or defined by an optimization objective) is identified. The inputs are the node states, and the target output is the suitability score or optimal k for that scenario.
    2. *FNN Initialization:* The FNN architecture is defined, and its parameters (e.g., initial shapes and positions of triangular MFs, initial rule weights/consequents) are initialized, often randomly or based on preliminary domain knowledge.
    3. *Learning Algorithm:* The FNN is trained using the generated dataset. A hybrid learning algorithm, often combining gradient descent (like backpropagation) with least-squares estimation (common in ANFIS), is typically used. The algorithm iteratively adjusts the FNN's tunable parameters to minimize the error between the FNN's predicted output and the target output from the dataset.
    4. *Stopping Criterion:* Training continues until a predefined error threshold is met, a maximum number of training epochs is reached, or the error on a separate validation dataset stops improving (to prevent overfitting).
    5. *Result:* The outcome is a trained FNN capable of evaluating network states and predicting desirable clustering actions.

2. **Online Operation (Round-by-Round):** Once trained, the FNN is deployed in the WSN to guide clustering dynamically in each communication round (r). This phase executes dynamically in the WSN during operation, as depicted in Algorithm 1.

    - **Step 1:** *Parameter Acquisition:* At the beginning of each round, nodes determine their current input parameter values (measure E_{res}, estimate distances, count neighbors, assess SNR).
    - **Step 2:** *FNN Evaluation:* Each node feeds its current parameters into its locally stored, trained FNN model to compute its CH suitability score (or contribute to determining k).
    - **Step 3:** *CH Selection and Cluster Formation:* Nodes potentially broadcast their suitability scores. A distributed protocol selects CHs based on these scores (e.g., highest score within a neighborhood, or probabilistic selection weighted by score).

The FNN might also output an optimal k, influencing how many CHs are ultimately selected. Non-CH nodes join the CH that requires the minimum communication energy (usually the closest).

- **Step 4:** *Steady-State Data Transmission:* Nodes transmit data to their CHs within their allocated time slots. CHs aggregate data and transmit it to the BS.
- **Step 5:** *Re-Clustering:* The process repeats from Step 1 for the next round, allowing the network to dynamically adapt its cluster structure based on the most recent node states.

This dynamic, round-by-round adaptation, guided by the intelligently trained FNN, is key to achieving superior energy efficiency and network longevity, especially in environments where node energy and network topology are not static. The FNN provides a mechanism to continuously strive for an optimal clustering configuration based on the current network reality. Table 2 shows all notations will be used in the paper.

Table 2: The table of notations.

Notation	**Description**
N	Total number of sensor nodes in the network
k	Size of a data packet (bits)
r	Current simulation round
$E_{TX}(k \cdot d)$	Energy consumed to transmit a k-bit packet over distance d
$E_{RX}(k)$	Energy consumed to receive a k-bit packet
$E_{DA}(k)$	Energy consumed for aggregating a k-bit data packet at a Cluster Head (CH)
E_{elec}	Energy required to operate transmitter/receiver circuitry (nJ/bit)
$\mathbf{E}_{\mathbf{fs}}$	Energy consumption coefficient for free-space amplifier model (pJ/bit/m^2)
$\mathbf{E}_{\mathbf{mp}}$	Energy consumption coefficient for multipath amplifier model (pJ/bit/m^4)
D	Distance between the transmitting node and the receiving node (m)
D_0	Threshold distance distinguishing free-space and multipath propagation models
E_{round}	Total energy consumed by all nodes during a single communication round
E^i_{residual}	Residual energy of node i at a given round
CH	Cluster Head node responsible for aggregating and forwarding data
BS	Base Station that collects aggregated data from CHs
ND	Node degree, representing the number of neighboring nodes within range
SNR	Signal-to-Noise Ratio, indicating link quality
K_{optimal}	Optimal number of clusters determined by the FNN
FND	First Node Dies: round at which the first node depletes energy
HND	Half Nodes Die: round at which 50% of nodes have died
LND	Last Node Dies: round at which the last node depletes energy

4 Simulation Setup and Evaluation Metrics

To assess the performance of the proposed FNN based clustering protocol, we conducted comprehensive simulations using MATLAB. The simulation environment and parameters were designed to reflect typical WSN scenarios and allow for fair comparison with established benchmark algorithms. The experiments were implemented using *MATLAB R2016*, chosen for its robust support for matrix computations, fuzzy systems, and neural network modeling. Simulations ran on a Windows 10 Pro workstation with the following specifications:

- *Processor:* Intel(R) Core(TM) i5-3210M CPU @ 2.50GHz
- *RAM:* 8 GB
- *Operating system:* 64-bit Windows

The FNN model was implemented using MATLAB's *Fuzzy Logic Toolbox*, complemented by custom scripts for adaptive clustering.

Benchmark protocols (LEACH, HEED, ELDC) were implemented following their standard formulations in the literature to ensure fair comparisons.

4.1 Simulation environment

The network consists of N sensor nodes randomly distributed within a square area. A single, stationary Base Station (BS) with unlimited energy is located at the center of the simulation field. All sensor nodes are assumed to have the same initial energy and capabilities. The key simulation parameters are summarized in Table 3.

4.2 Comparison protocols

The performance of the proposed FNN protocol is compared against three well-known clustering algorithms:

1. *LEACH (Low-Energy Adaptive Clustering Hierarchy):* A distributed, probabilistic clustering protocol serving as a common baseline [12, 18].
2. *HEED (Hybrid Energy-Efficient Distributed Clustering):* A distributed protocol that primarily uses residual energy for CH selection [29].
3. *ELDC (Energy-efficient and Load-balanced clustering using ANNs):* The Artificial Neural Network-based approach described in [26], which serves as a direct machine-learning baseline for our FNN model.

Table 3: Simulation parameters.

Parameter	**Value**	**Description**
Network Area	100m $\times 100m$	Simulation field dimensions
Number of Nodes (N)	100	Total sensor nodes deployed
Node Deployment	Random uniform distribution	Initial placement of nodes
BS Location	(50m, 50m)	Central location of the Base Station
Initial Node Energy $\mathbf{E_0}$	0.5 J	Starting energy for each sensor node
Data Packet Size (k)	4000 bits	Size of data packet transmitted/received
Control Packet Size	Assumed smaller, (e.g., 200 bits)	Size of overhead/control messages
$\mathbf{E}_{\text{elec}}$	50 nJ/bit	Energy for transmitter/receiver electronics
$\mathbf{E_{fs}}$	10 pJ/bit/m	Amplifier energy for free space model ($d < d_0$)
$\mathbf{E_{mp}}$	0.0013 pJ/bit/m^4	Amplifier energy for multi-path model ($d \geq d_0$)
$\mathbf{E_{DA}}$	5 nJ/bit/signal	Energy for data aggregation at CH
Maximum Rounds (r_{max})	2000	Simulation duration limit
Simulator	MATLAB R20xx	Software used for simulation

4.3 Experimental procedure

The experimental evaluation was conducted as follows:

1. Network Initialization: 100 nodes were randomly distributed within the 100m $\times$ 100m area, and all nodes were assigned equal initial energy.

2. Protocol Execution: Each protocol (LEACH, HEED, ELDC, FNN) was executed under identical conditions for 2000 simulation rounds.

3. FNN Operation:

    - The FNN was trained offline with simulated network states to learn optimal clustering decisions.
    - During runtime, each node locally evaluated its CH suitability score using the trained FNN model.

4. Repetition: Each simulation was repeated 10 times with different random node placements to ensure statistical reliability, and the average results were reported.

4.4 Evaluation metrics

The following metrics are used to quantitatively assess the performance of the clustering protocols:

- **Network Lifetime:** This is the primary metric, measured in terms of simulation rounds. We consider three key milestones:
  - *First Node Dies (FND):* The round number when the first sensor node depletes its energy. This indicates the start of network degradation and loss of sensing coverage.
  - *Half Nodes Die (HND):* The round number when 50% of the sensor nodes have depleted their energy. This represents significant network functionality loss.
  - *Last Node Dies (LND):* The round number when the last active sensor node depletes its energy. This indicates the total operational span of the network.
- **Stability Period:** The duration, in rounds, from the start of the simulation until the FND occurs. A longer stability period is desirable.
- **Number of Alive Nodes per Round:** Tracks the count of nodes with remaining energy in each round, providing insight into the network's degradation rate.
- **Energy Consumption:**
  - Cumulative Energy Consumption: The total energy consumed by all nodes in the network over the simulation rounds. Lower cumulative consumption indicates higher efficiency.
  - Energy Consumption per Round: The rate at which energy is consumed, providing insight into the efficiency of operations within each round.

These metrics collectively provide a comprehensive view of the energy efficiency, longevity, and robustness of the proposed FNN clustering protocol compared to the benchmark algorithms.

5 Results and discussion

This section presents the simulation results comparing the performance of the proposed Fuzzy Neural Network (FNN) based clustering protocol against the benchmark algorithms: LEACH [18], HEED [29], and the ANN-based ELDC [26]. The evaluation focuses on key metrics including network lifetime, stability, and energy consumption. All reported results represent the average performance over 30 independent simulation runs with different random seeds to ensure robustness

5.1 Network lifetime and stability analysis

Network lifetime and stability are critical indicators of a WSN's operational effectiveness. We analyze these using the number of active nodes over time and specific node death milestones.

- *Alive Nodes vs. Rounds:* Figure 4 illustrates the number of nodes remaining operational (with energy > 0) as the simulation progresses over rounds.

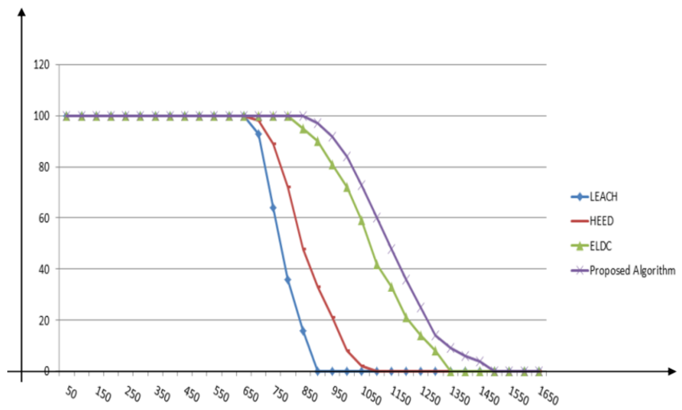

Figure 4: The number of alive nodes vs. rounds for LEACH, HEED, ELDC, and FNN.

  As depicted in Figure 4, the curve representing the proposed FNN protocol remains at 100% (all nodes alive) for a significantly longer duration compared to the other protocols. Furthermore, the rate of node death (slope of the curve) for the FNN is considerably slower. This visually confirms that the FNN approach sustains network operation more effectively and exhibits a more graceful degradation compared to LEACH, HEED, and ELDC. LEACH shows the most rapid decline in active nodes.

- *Stability Period (FND):* The First Node Death marks the end of the network's fully operational phase. Figure 5 specifically compares this metric.

  Figure 5 clearly shows that the FND occurs latest for the proposed FNN protocol. This extended stability period signifies that the FNN's intelligent clustering prevents the premature depletion of any single node's energy, likely due to better load balancing in CH selection and rotation. FNN outperforms ELDC, which in turn outperforms HEED and LEACH in this regard.

- *Total Network Lifetime (LND):* The Last Node Death indicates the maximum operational lifespan the network can achieve. Figure 6 compares the LND across protocols.

  Consistent with the FND results, Figure 6 demonstrates that the proposed FNN achieves the longest total network lifetime. The round number at which the last node dies is signif-

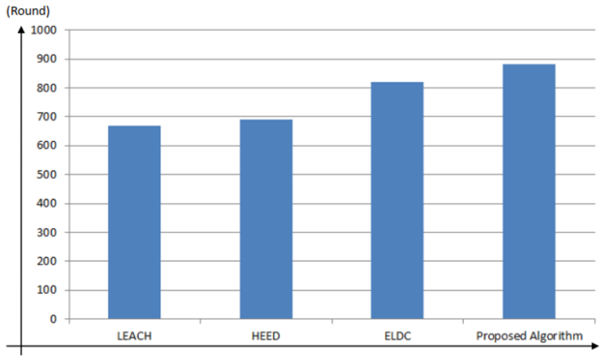

Figure 5: The round number for First Node Dies (FND) for LEACH, HEED, ELDC, and FNN.

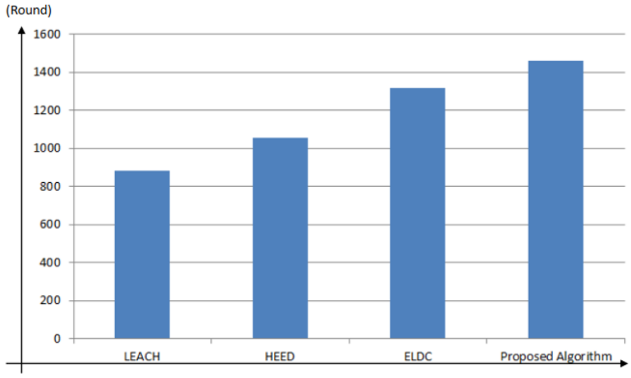

Figure 6: Comparing the round number for Last Node Dies (LND) for LEACH, HEED, ELDC, and FNN.

icantly higher for FNN compared to ELDC, HEED, and LEACH, indicating its superior ability to manage energy resources over the long term.

5.2 Energy consumption analysis

Efficient energy management is the core objective. We analyze the cumulative energy consumption and the energy consumed in specific phases of the network's life.

- *Cumulative Energy Consumption:* Figure 7 tracks the total energy consumed by all nodes in the network from the beginning of the simulation.

The curve for the proposed FNN in Figure 7 exhibits the lowest slope, indicating the lowest rate of energy consumption per round. This implies that the clustering and data transmission strategy employed by the FNN is inherently more energy-efficient than those of the other protocols. LEACH shows the highest rate of energy depletion.

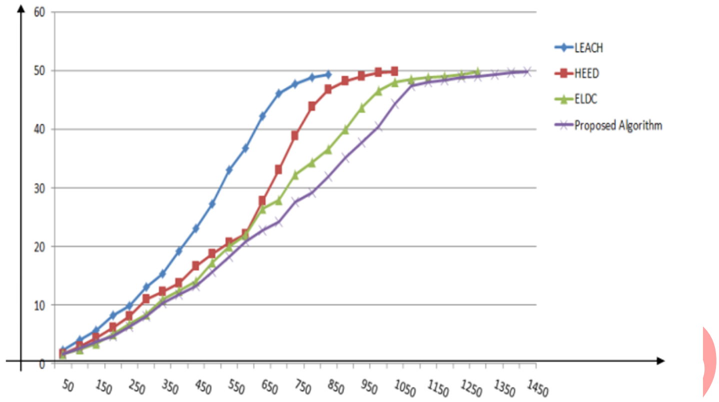

Figure 7: Cumulative energy consumption (e.g., in Joules) vs. rounds for LEACH, HEED, ELDC, and FNN.

- *Energy Consumption in Round Intervals:* Figure 8 provides a more detailed view by comparing the energy consumed within specific intervals (e.g., Rounds 1-250, 251-500, 501-750, 751-900).

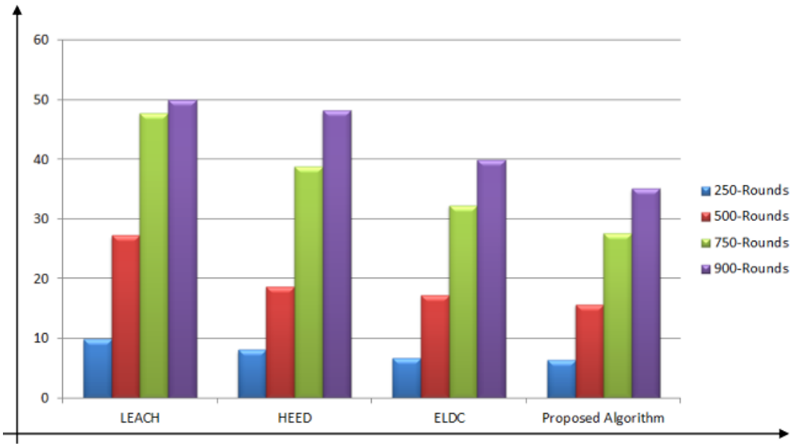

Figure 8: Total energy consumed within specific round intervals for LEACH, HEED, ELDC, and FNN.

Figure 8 demonstrates that the FNN protocol consistently consumes less energy not only overall but also within different operational phases of the network. This consistent efficiency across early, mid, and late stages highlights the robustness and adaptability of the FNN's energy management strategy.

5.3 Node death progression summary

Figure 9 provides a consolidated view comparing the key lifetime milestones: FND, HND (Half Nodes Die), and LND.

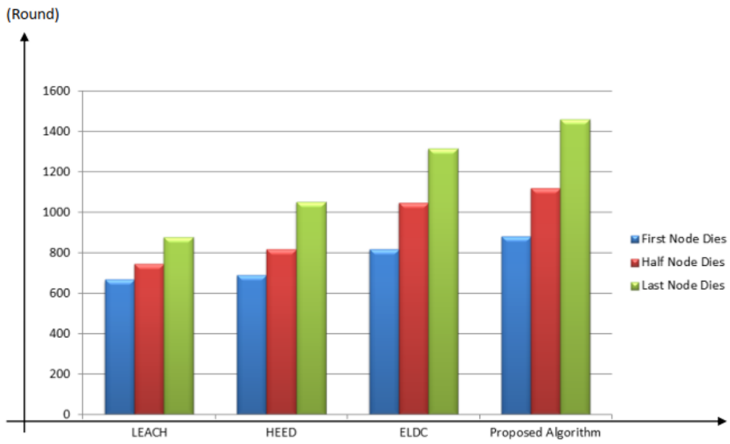

Figure 9: Comparing FND, HND, and LND round numbers for LEACH, HEED, ELDC, and FNN.

This summary chart, Figure 9, reinforces the previous findings. The proposed FNN significantly delays all stages of network degradation (FND, HND, LND) compared to ELDC, HEED, and LEACH. Notably, the interval between FND and LND is also wider for FNN, suggesting that even after some nodes start dying, the network under FNN control maintains partial functionality for a longer period, indicating more graceful degradation.

5.4 Discussion

The collective simulation results consistently underscore the advantages of the proposed FNN-based dynamic clustering approach. The FNN demonstrably outperforms LEACH, HEED, and the baseline ANN model (ELDC) across all evaluated metrics: stability period (FND), total lifetime (LND), number of active nodes over time, and overall energy efficiency.

The superior performance of the FNN can be attributed to several factors inherent in its design:

1. *Intelligent Decision Making:* The FNN leverages its trained knowledge base to make more informed decisions. Unlike many approaches that only output a CH suitability score, our framework allows the FNN to influence both CH selection and the optimal number of clusters (k). This dual optimization leads to a more globally efficient network structure in each round, adapting not just which nodes are CHs, but also how many CHs there should be. It considers multiple factors (residual energy, distance metrics, density

via neighbors, SNR) simultaneously within a framework that handles inherent uncertainties and imprecision.

2. *Adaptive Load Balancing:* Unlike LEACH's purely probabilistic approach or HEED's more fixed criteria, the FNN dynamically adapts the clustering structure based on the current state of the network. This leads to a more equitable distribution of the energy-intensive CH role among nodes, preventing energy hotspots and premature node failure, thus enhancing both stability (FND) and overall lifetime (LND).

3. *Enhanced Robustness:* Compared to the ELDC (ANN) baseline, the integration of fuzzy logic provides greater robustness to noisy sensor readings and minor fluctuations in network parameters. Fuzzy logic's ability to work with linguistic variables and overlapping membership functions allows for smoother transitions and less sensitivity to crisp input thresholds. This theoretical benefit is empirically validated by our FNN's superior performance over ELDC across all key metrics (Figures 4-9), confirming that the fuzzy component adds tangible value in managing the inherent uncertainty of WSNs.

4. *Optimized Energy Usage:* By forming more effective clusters and selecting strategically located CHs with sufficient energy, the FNN minimizes intra-cluster and inter-cluster communication distances and reduces redundant data transmissions through efficient aggregation, leading to lower overall energy consumption (as seen in Figures 4-3 and 4-5).

In essence, the FNN acts as an intelligent controller that continuously optimizes the network's clustering configuration round by round, leading to significant gains in longevity and efficiency, particularly highlighting its suitability for dynamic WSN environments where conditions are not static.

The proposed FNN is evaluated using the first-order radio model, which provides a widely accepted baseline for energy consumption analysis in WSNs. While this model simplifies radio behavior, real-world WSNs often face additional energy drain due to hardware inconsistencies, interference, and environmental factors. The FNN incorporates limited robustness to these factors through fuzzy reasoning and parameter noise during training. However, its current formulation does not explicitly model all non-idealities. Future work will extend this approach by incorporating stochastic radio models and real-world measurement data to enhance the FNN's reliability under practical deployment conditions.

6 Scalability and complexity analysis

For any WSN protocol to be considered practical, it must be able to scale to networks of varying sizes and operate within the severe computational and memory constraints of individual sensor nodes. Although the proposed FNN protocol was tested on a network of 100 nodes, its distributed decision-making structure ensures that computational cost per node remains constant and total overhead scales linearly with network size. Therefore, the protocol is theoretically scalable to larger WSN deployments. However, explicit simulations with networks of 500+ nodes and different area sizes were not performed in this study. Future work will involve large-scale experiments to empirically validate the protocol's performance under high-density deployments. This section analyzes the scalability and complexity of our proposed FNN-based approach.

6.1 Scalability of the proposed FNN algorithm

The proposed FNN protocol is designed to be highly scalable. This scalability is achieved primarily through its distributed intelligence and hierarchical structure.

1. *Distributed Decision-Making:* The most computationally significant task during the online phase, the FNN evaluation to calculate a node's CH suitability score, is performed locally at each node. The inputs to the FNN (residual energy, SNR, etc.) are either known internally or determined through communication with only immediate one-hop neighbors (for node degree). A node does not require information from the entire network to make its decision. This localized nature prevents the communication overhead and processing bottlenecks that would occur if a central entity had to collect data from all N nodes to make a decision.

2. *Hierarchical Operation:* By leveraging a clustering hierarchy, the protocol naturally partitions the network. The energy-intensive task of long-range communication is restricted to the small subset of nodes elected as CHs. As the network scales in size and density, the number of clusters can be adaptively managed by the FNN, ensuring the network architecture remains efficient.

Assumptions and Conditions:

The scalability of the system assumes that the underlying clustered topology remains effective. In extremely large or dense networks, the Base Station (BS) could become a bottleneck if the number of CHs reporting to it becomes excessive. Similarly, a CH in a very dense region could become overloaded. However, this is a general challenge for all CH-based protocols, and our FNN's ability to dynamically adjust clustering based on local density (Node Degree) inherently works to mitigate this issue.

The proposed FNN-based protocol scales effectively under the following assumptions:

1. *Nodes have minimal computational capacity* to execute lightweight FNN evaluation (a small neural-fuzzy inference model).

2. *Local parameter acquisition* (residual energy, neighbor density, SNR, etc.) is available via periodic sensing or low-overhead control messages.

3. The Base Station (BS) is stationary and does not impose centralized control, ensuring that network expansion does not increase processing overhead at a single point.

4. Clustering decisions are made locally, and communication overhead remains proportional to the number of neighboring nodes, not the total network size.

Under these conditions, the algorithm's overhead grows linearly with the number of nodes $O(N)$, which is acceptable for WSNs.

6.2 Complexity analysis of the proposed FNN algorithm

We analyze the complexity of algorithm by separating its two distinct phases: offline training and online operation.

- **Offline Training Phase (Executed Once):**
  - *Computational Complexity:* The training of the FNN, similar to any neural network, is a computationally intensive process. Its complexity depends on the size of the training dataset (D), the number of training epochs (E), and the number of parameters in the FNN model. This is a high-cost operation, but it is critical to note that it is a one-time, pre-deployment cost. It is performed on a powerful external computer and consumes zero energy or processing time from the sensor nodes themselves.
  - *Space Complexity:* The training phase requires significant memory on the host computer to store the dataset and the model, but this has no bearing on the sensor nodes.
- **Online Operational Phase (Executed Each Round):**
  - *Computational Complexity:* The per-round complexity for a single node is the sum of its tasks:
    1. *Parameter Acquisition:* A node measures its own residual energy and assesses SNR ($O(1)$). To determine its Node Degree (ND) and Average Distance to Neighbors (DN), it must exchange beacon messages with its k immediate neighbors. This requires $O(k)$ computation.

2. *FNN Evaluation:* The node performs a forward pass through the trained FNN. The number of calculations is determined by the fixed structure of the FNN (number of inputs, rules, and outputs). This is a constant-time operation, $O(1)$, with respect to the total number of nodes N in the network.
3. *CH Selection & Cluster Formation:* A node broadcasts its CH suitability score and listens for the scores of its k neighbors. It then joins the best CH. This process is dominated by local communication and comparison, making its complexity $O(k)$.

Therefore, the total online computational complexity for a single node per round is dominated by its interaction with its neighbors, resulting in $O(k)$. Since k is typically much smaller than $N(k << N)$, this is a highly efficient and scalable complexity.

- *Space Complexity:* The memory required on each sensor node is also minimal:

1. *The FNN Model Storage:* The primary storage requirement is for the parameters of the trained FNN model (membership functions, rule weights, etc.). The size of this model is fixed after training and is independent of the network size N.
2. *Neighbor Table:* A node needs temporary storage to maintain information about its k neighbors. This requires $O(k)$ space.

The total space complexity per node is therefore constant for a given FNN model and $O(k)$ for neighbor data. This small and predictable memory footprint makes our protocol perfectly suitable for deployment on resource-constrained sensor nodes.

7 Overhead comparison summary

The proposed FNN requires each node to compute a CH suitability score by evaluating a small neuro-fuzzy model. This computation is lightweight, involving only a few membership evaluations and rule-based calculations, resulting in constant-time complexity per node $O(1)$. Communication overhead remains comparable to LEACH since no additional control messages or global optimization steps are required. Therefore, the overhead introduced by the FNN is minimal and does not compromise its suitability for resource-constrained WSN nodes. All results have been shown in Table 4.

Computational Overhead

- LEACH:
  - Uses a *probabilistic CH selection* based on a simple threshold function.

Table 4: Overhead Comparison summary.

Protocol	Computation per Node	Communication Overhead	Remarks
LEACH	O(1), minimal	Low	Very lightweight, but energy-inefficient
HEED	O(1), moderate	Low to moderate	Static parameters, improved over LEACH
ELDC (ANN)	O(W),where W=ANN weights	Low	Higher than LEACH, adaptive but noise-sensitive
FNN (Proposed)	O(L×M), small constant	Low (similar to LEACH)	Slightly higher computation, negligible impact on runtime

- Complexity per node is $O(1)$ with minimal computations.

- **Proposed FNN:**
  - Each node evaluates a lightweight FNN model with a small number of fuzzy rules and membership functions.
  - Complexity per node is $O(L \times M)$, where LLL is the number of network layers and MMM is the number of fuzzy rules.
  - Since LLL and MMM are fixed and small, the per-node computation remains constant $(O(1))$, only slightly higher than LEACH.
  - The FNN computations involve simple arithmetic operations (membership evaluation, rule firing, and weighted summation), which are well within the processing capacity of typical WSN nodes.

Communication Overhead

- LEACH:
  - Requires nodes to send local information to candidate CHs and for CHs to broadcast their status.
  - Communication overhead is low but limited by its probabilistic selection, which can lead to suboptimal energy use.
- Proposed FNN:
  - Similar to LEACH, the FNN does not require extra global communication.
  - Nodes make decisions locally and only send a single suitability score or CH advertisement to neighbors or the base station as needed.
  - No additional iterations or global information exchange are required, unlike meta-heuristic or swarm-based protocols.

8 Conclusion and future work

Addressing energy constraints, particularly in dynamic WSNs, is crucial for their widespread adoption. This paper introduced an intelligent and dynamic clustering mechanism utilizing a Fuzzy Neural Network (FNN). The FNN adaptively optimizes the clustering process by learning complex relationships between real-time network parameters (energy, distance, density, SNR) and ideal cluster configurations. Simulation results validated the proposed approach, demonstrating significant improvements in network lifetime, stability, and energy efficiency compared to benchmark protocols like LEACH, HEED, and the ANN-based ELDC. The FNN's ability to handle uncertainty and adapt dynamically makes it particularly suitable for WSNs with mobile nodes or changing environmental conditions. Future research will focus on extending the FNN model to explicitly incorporate node mobility prediction, investigate its performance in larger-scale, heterogeneous networks, and explore the integration of Quality of Service (QoS) metrics alongside energy efficiency within the FNN framework. Further work could also involve hardware implementation and real-world testing to validate the simulation findings.

Declarations

Availability of Supporting Data

All data generated or analyzed during this study are included in this published paper.

Funding

The authors have been supported by Gonbad Kavous University within the project with title "*Adaptive Clustering in WSN using neural network*" with grant no. 6/03/153.

Competing Interests

The authors declare that they have no competing interests relevant to the content of this paper.

Authors' Contributions

The main text of manuscript is collectively written by the authors.

References

[1] Akram, M., Cho, T.H. (2016). "Energy efficient fuzzy adaptive selection of verification nodes in wireless sensor networks". *Ad Hoc Networks*, 47, 16-25, doi:https://doi.org/10.1016/j.adhoc.2016.04.010.

[2] Babakordi, F. (2023). "An efficient method for solving the fuzzy AH1N1/09 influenza model using the fuzzy Atangana-Baleanu-Caputo fractional derivative", *Fuzzy Optimization and Modeling Journal*, 4(2), 27-38, doi:https://doi.org/10.30495/fomj.2023.1988760.1096.

[3] Babakordi, F. (2024). "Arithmetic operations on generalized trapezoidal hesitant fuzzy numbers and their application to solving generalized trapezoidal hesitant fully fuzzy equation", *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems*, 32(1), 85-108, doi:https://doi.org/10.1142/S0218488524500041.

[4] Babakordi, F., Allahviranloo, T., Shahriari, M.R., Catak, M. (2024). "Fuzzy Laplace transform method for a fractional fuzzy economic model based on market equilibrium", *Information Sciences*, 665, 120308, doi:https://doi.org/10.1016/j.ins.2024.120308.

[5] Chamam, A., Pierre, S. (2010). "A distributed energy-efficient clustering protocol for wireless sensor networks", *Computers & Electrical Engineering*, 36(2), 303-312, doi:https://doi.org/10.1016/j.compeleceng.2009.03.008.

[6] Chang, J.Y., Ju, P.H. (2012). "An efficient cluster-based power saving scheme for wireless sensor networks", *EURASIP Journal on Wireless Communications and Networking*, 2012, 1-10, doi:https://doi.org/10.1186/1687-1499-2012-172.

[7] Deepa, K., Zaheeruddin, Vashist, S. (2021). "Density based fuzzy C means clustering to prolong network lifetime in smart grids", *Wireless Personal Communications*, 119(3), 2817-2836, doi:https://doi.org/10.1007/s11277-021-08371-w.

[8] Garcia, M., Sendra, S., Lloret, J., Canovas, A. (2013). "Saving energy and improving communications using cooperative group-based wireless sensor networks", *Telecommunication Systems*, 52, 2489-2502, doi:https://doi.org/10.1007/s11235-011-9568-3.

[9] Gou, H., Yoo, Y. (2010). "An energy balancing LEACH algorithm for wireless sensor networks", *2010 Seventh International Conference on Information Technology: New Generations, Las Vegas, NV, USA*, (pp. 822-827), doi:https://doi.org/10.1109/ITNG.2010.12.

[10] Grover, J., Sharma, M. (2014). "Optimized GAF in wireless sensor network", *Proceedings of 3rd International Conference on Reliability, Infocom Technologies and Optimization*,Noida, India, (pp. 1-6), doi:https://doi.org/10.1109/ICRITO.2014.7014686.

[11] Guo, W., Zhang, W., Lu, G. (2010). "PEGASIS protocol in wireless sensor network based on an improved ant colony algorithm", *2010 Second International Workshop on Education Technology and Computer Science, Wuhan, China,* (pp. 64-67), doi:https://doi.org/10.1109/ETCS.2010.285.

[12] Heinzelman, W.R., Chandrakasan, A., Balakrishnan, H. (2000). "Energy-efficient communication protocol for wireless microsensor networks", *Proceedings of the 33rd Annual Hawaii International Conference on System Sciences, Maui, HI, USA,* (pp. 10 pp. vol.2), doi:https://doi.org/10.1109/HICSS.2000.926982.

[13] Ibrahim, M.E., Ahmed, A.E. (2022). "Energy□aware intelligent hybrid routing protocol for wireless sensor networks", *Concurrency and Computation: Practice and Experience*, 34(3), e6601, doi:https://doi.org/10.1002/cpe.6601.

[14] Jalili, A. (2024). "A hybrid fuzzy-genetic algorithm for energy-efficient routing in wireless sensor networks", *Fuzzy Optimization and Modeling Journal*, 5(4), doi:https://doi.org/10.71808/fomj.2024.1194011.

[15] Jalili, A., Gheisari, M., Alzubi, J.A., Fernández-Campusano, C., Kamalov, F., Moussa, S. (2024). "A novel model for efficient cluster head selection in mobile WSNs using residual energy and neural networks", *Measurement: Sensors*, 33, 101144, doi:https://doi.org/10.1016/j.measen.2024.101144.

[16] Jalili, A., Alzubi, J.A., Rezaei, R., Webber, J.L., Fernández-Campusano, C., Gheisari, M., Mehbodniya, A. (2024). "Markov chain-based analysis and fault tolerance technique for enhancing chain-based routing in WSNs", *Concurrency and Computation: Practice and Experience*, 36(12), e8032, doi:https://doi.org/10.1002/cpe.8032.

[17] Jung, S.M., Han, Y.J., Chung, T.M. (2007). "The concentric clustering scheme for efficient energy consumption in the PEGASIS", *Proceedings of the 9th International Conference on Advanced Communication Technology, Gangwon, Korea (South)*, 12-17, 260-265, doi:https://doi.org/10.1109/ICACT.2007.358351.

[18] Farooq, M.O., Dogar, A.B., Shah, G.A. (2010). "MR-LEACH: multi-hop routing with low energy adaptive clustering hierarchy", *2010 Fourth International Conference on Sensor Technologies and Applications, Venice, Italy*, 262-268, doi:https://doi.org/10.1109/SENSORCOMM.2010.48.

[19] Fouladlou, M., Khademzadeh, A. (2017). "An energy efficient clustering algorithm for wireless sensor devices in internet of things", *2017 Artificial Intelligence and Robotics (IRANOPEN), Qazvin, Iran*, (pp. 39-44), doi:https://doi.org/10.1109/RIOS.2017.7956441.

[20] Khabiri, M., Ghaffari, A. (2018). "Energy-aware clustering-based routing in wireless sensor networks using cuckoo optimization algorithm", *Wireless Personal Communications*, 98(3), 2473-2495, doi:https://doi.org/10.1007/s11277-017-4983-8.

[21] Klaoudatou, E., Konstantinou, E., Kambourakis, G., Gritzalis, S. (2011). "A survey on cluster-based group key agreement protocols for WSNs", *IEEE Communications Surveys & Tutorials*, 13(3), 429-442, doi:https://doi.org/10.1109/SURV.2011.061710.00109.

[22] Lee, J.S., Kao, T.Y. (2016). "An improved three-layer low-energy adaptive clustering hierarchy for wireless sensor networks", *IEEE Internet of Things Journal*, 3(6), 951-958, doi:http://dx.doi.org/10.1109/JIOT.2016.2530682.

[23] Ma, J., Wang, S., Meng, C., Ge, Y., Du, J. (2018). "Hybrid energy-efficient APTEEN protocol based on ant colony algorithm in wireless sensor network", *EURASIP Journal on Wireless Communications and Networking*, 2018(1), 102, doi:https://doi.org/10.1186/s13638-018-1106-5.

[24] Mehmood, A., Lloret, J., Sendra, S. (2016). "A secure and low-energy zone-based wireless sensor networks routing protocol for pollution monitoring", *Wireless Communications and Mobile Computing*, 16(17), 2869-2883, doi:https://doi.org/10.1002/wcm.2734.

[25] Mehmood, A., Khan, S., Shams, B., Lloret, J. (2015). "Energy-efficient multi-level and distance-aware clustering mechanism for WSNs", *International Journal of Communication Systems*, 28(5), 972-989, doi:https://doi.org/10.1002/dac.2720.

[26] Mehmood, A., Lv, Z., Lloret, J., Umar, M.M. (2017). "ELDC: An artificial neural network based energy-efficient and robust routing scheme for pollution monitoring in WSNs", *IEEE Transactions on Emerging Topics in Computing*, 8(1), 106-114, doi:http://dx.doi.org/10.1109/TETC.2017.2671847.

[27] Mosavvar, I., Ghaffari, A. (2019). "Data aggregation in wireless sensor networks using firefly algorithm", *Wireless Personal Communications*, 104(1), 307-324, doi:https://doi.org/10.1007/s11277-018-6021-x.

[28] Sefati, S., Abdi, M., Ghaffari, A. (2021). "Cluster-based data transmission scheme in wireless sensor networks using black hole and ant colony algorithms", *International Journal of Communication Systems*, 34(9), e4768, doi:https://doi.org/10.1002/dac.4768.

[29] Younis, O., Fahmy, S. (2004). "HEED: A hybrid, energy-efficient, distributed clustering approach for ad hoc sensor networks", *IEEE Transactions on Mobile Computing*, 3(4), 366-379, doi:https://doi.org/10.1109/TMC.2004.41.

[30] Zeng, K., Ren, K., Lou, W., Moran, P.J. (2006). "Energy-aware geographic routing in lossy wireless sensor networks with environmental energy supply", *Proceedings of the 3rd International Conference on Quality of Service in Heterogeneous Wired/Wireless Networks (QShine '06). Association for Computing Machinery, New York, NY, USA*, 8-es, doi:https://doi.org/10.1145/1185373.1185384.

[31] Zhixiang, D., Bensheng, Q. (2008). "Three-layered routing protocol for WSN based on LEACH algorithm", *2007 IET Conference on Wireless, Mobile and Sensor Networks (CCWMSN07), Shanghai*, 72-75, doi:http://dx.doi.org/10.1049/cp:20070086.