People's Democratic Republic of Algeria

Ministry of Higher Education and Scientific Research

University AMO of Bouira

Faculty of Exact Sciences

Department of Computer Science

# Master's thesis

## in Computer Science

***Specialty:*** *AI*

# Theme

---

## Hybrid AI-Based Protocol for Energy-Efficient Routing in WSNs Using GA and Q-Learning

---

**Supervised by**

- Mr. Akli ABBAS

**Made by**

- Ayoub Louaidi
- Ahmed Cherik

2024/2025

# *Acknowledgments*

**All praise and thanks be to Allah — Alhamdulillah — for granting us the strength, patience, and determination to complete this thesis. Without His guidance and blessings, none of this would have been possible.**

**We would like to express our heartfelt gratitude to our supervisor, Dr. ABBAS, for his invaluable support, patience, and insightful guidance throughout this journey. His expertise and continuous encouragement were essential in shaping the direction and quality of this work.**

**We extend our sincere appreciation to Mr.OUKAS Noureddine, Mr.DJOUABRI Abderrezak, and Mr.HAMID Rabah for accepting to be members of the jury. We are truly honored by your time and the consideration you gave to our work.**

**A special mention goes to Mr. YAHOUI Soheib, whose work on the previous version of this thesis laid a valuable foundation upon which we built and improved. Your contributions were instrumental in helping us develop this project further.**

**We also want to thank our families and loved ones for their continuous support, sacrifices, and prayers.**

*With sincere respect and gratitude,*

*CHERIK Ahmed & LOUAIDI Ayoub*

# *Dedication (Ayoub)*

*******************************************************************************************

To my dear parents — your unwavering belief in me has been my greatest motivation. This work is a small reflection of the values and perseverance you've instilled in me. Ayoub

*Ayoub.*

# *Dedication (Ahmed)*

********************************************************************************

To my beloved parents, sisters, and grandmother — your love, sacrifices, and prayers have always been my greatest source of strength. I dedicate this work to you with all my heart. Ahmed

*Ahmed.*

## Abstract

Wireless Sensor Networks (WSNs) play a critical role in enabling autonomous data acquisition in diverse application domains, including environmental monitoring, industrial control, and smart agriculture. However, these networks remain severely constrained by limited energy resources, unpredictable node failures, and scalability challenges. To address these issues, this thesis proposes **EAGLE** (Energy-Aware Genetic and Learning-based Engine), a novel hybrid protocol that integrates Genetic Algorithms (GA) and Q-learning to enhance energy efficiency, reliability, and adaptability in WSNs.

The proposed architecture employs a hierarchical, layered-zone topology in which sensor nodes are organized into concentric levels and angular zones. Within each zone, cluster heads (CHs) and backup CHs are selected using a GA-based optimization strategy that favors high residual energy and spatial centrality. Routing between CHs is managed by a distributed Q-learning model, where each CH learns optimal next-hop decisions based on distance-driven reward functions and dynamic network conditions.

Extensive simulations were conducted on Google Colab to compare EAGLE with a standard K-means-based clustering method. Performance metrics such as first node death (FND), half node death (HND), cumulative packet delivery, energy distribution fairness, and per-packet energy cost were analyzed. Results show that EAGLE significantly outperforms the baseline in terms of network longevity, energy balance, and transmission reliability.

EAGLE demonstrates that combining evolutionary computation with reinforcement learning offers a powerful and scalable solution for intelligent WSN routing. This hybrid approach provides a pathway toward autonomous, energy-aware, and self-organizing sensor networks capable of operating efficiently in dynamic environments.

# الملخص

تلعب شبكات الاستشعار اللاسلكية (WSNs) دوراً حيوياً في تمكين جمع البيانات بشكل ذاتي في مجموعة متنوعة من المجالات التطبيقية، بما في ذلك المراقبة البيئية، والتحكم الصناعي، والزراعة الذكية. ومع ذلك، تظل هذه الشبكات مقيدة بشدة بسبب محدودية موارد الطاقة، وفشل العقد غير المتوقع، وتحديات التوسع.

لمعالجة هذه التحديات، تقترح هذه الأطروحة بروتوكولاً هجينيًا جديدًا يُدعى **EAGLE** (محرك قائم على الخوارزميات الجينية والتعلم المعزز للتوعية الطاقوية)، والذي يدمج بين الخوارزميات الجينية (GA) وخوارزمية التعلم المعزز (Q-learning) بهدف تحسين كفاءة الطاقة، والموثوقية، والقدرة على التكيف داخل شبكات الاستشعار اللاسلكية.

يعتمد المعمار المقترح على بنية هرمية مقسمة إلى طبقات ومناطق زاوية، حيث يتم تنظيم العقد الاستشعارية في مستويات دائرية متراكزة ومناطق حسب الزاوية. يتم اختيار رؤساء العناقيد (CHs) ونوابهم ضمن كل منطقة باستخدام استراتيجية تحسين قائمة على الخوارزميات الجينية، تفضل العقد ذات الطاقة المتبقية العالية والموقع المركزي. أما التوجيه بين رؤساء العناقيد، فيتم التحكم فيه عبر نموذج تعلّم معزز موزّع (Q-learning)، حيث تتعلم كل عقدة رئيسية اتخاذ القرار الأمثل للوجهة التالية استنادًا إلى دوال مكافأة تعتمد على المسافة وظروف الشبكة المتغيرة.

تم إجراء محاكاة موسعة على منصة Google Colab لمقارنة EAGLE مع طريقة تجميع قياسية تعتمد على خوارزمية K-means. تم تحليل مؤشرات الأداء مثل أول وفاة لعقدة (FND)، وفاة نصف العقد (HND)، العدد التراكمي للرزم المرسلة، عدالة توزيع الطاقة، وتكلفة الطاقة لكل رزمة. أظهرت النتائج أن EAGLE يتفوق بشكل كبير على النموذج الأساسي من حيث عمر الشبكة، وتوازن الطاقة، وموثوقية الإرسال.

تُظهر هذه الدراسة أن الجمع بين الخوارزميات التطورية والتعلم المعزز يوفر حلاً فعالًا وقابلاً للتوسع من أجل التوجيه الذكي في شبكات الاستشعار اللاسلكية. هذا النهج الهجين يفتح الطريق نحو شبكات استشعار ذاتية التنظيم، موفرة للطاقة، وذات كفاءة عالية في بيئات ديناميكية.

# Résumé

Les réseaux de capteurs sans fil (Wireless Sensor Networks, WSN) constituent une technologie essentielle pour la collecte autonome de données dans de nombreux domaines d'application, tels que la surveillance environnementale, l'automatisation industrielle et l'agriculture intelligente. Cependant, ces réseaux sont confrontés à des contraintes majeures, notamment la limitation des ressources énergétiques, les risques de défaillance des nœuds, et les défis liés à l'évolutivité.

Dans ce contexte, ce mémoire propose une nouvelle approche hybride appelée **EAGLE** (Energy-Aware Genetic and Learning-based Engine), combinant les Algorithmes Génétiques (GA) et le Q-learning, une méthode d'apprentissage par renforcement. Cette architecture a pour objectif d'optimiser la consommation énergétique, d'améliorer la fiabilité des communications, et de renforcer l'adaptabilité du réseau face aux changements dynamiques.

Le protocole EAGLE repose sur une topologie hiérarchique en couches et en zones angulaires. À l'intérieur de chaque zone, les têtes de cluster (CH) et leurs remplaçants sont sélectionnés à l'aide d'un algorithme génétique prenant en compte l'énergie résiduelle et la centralité spatiale. Le routage inter-cluster est assuré par un mécanisme de Q-learning distribué, où chaque CH apprend à choisir le meilleur prochain saut en fonction de récompenses négatives basées sur la distance euclidienne.

Des simulations étendues ont été réalisées sur la plateforme Google Colab afin de comparer les performances d'EAGLE à une méthode de clustering classique basée sur K-means. Les résultats ont mis en évidence des améliorations notables en termes de durée de vie du réseau (FND et HND), de répartition équitable de l'énergie, de volume de données transmises et de coût énergétique par paquet.

EAGLE démontre que la combinaison de l'intelligence évolutive et de l'apprentissage adaptatif constitue une solution puissante et évolutive pour les WSNs modernes. Cette approche hybride ouvre la voie à la conception de réseaux de capteurs intelligents, autonomes et économes en énergie, capables de fonctionner efficacement dans des environnements dynamiques.

# Contents

# List of Figures

# List of Tables

# List of abbreviations

| | |
|---|---|
| WSN | Wireless Sensor Network |
| MEMS | Micro Electro Mechanical Systems |
| ADC | Analog-to-Digital Converter |
| SPIN | Sensor Protocols for Information via Negotiation |
| LEACH | Low-Energy Adaptive Clustering Hierarchy |
| PEGASIS | Power-Efficient GAthering in Sensor Information System |
| TEEN | Threshold-sensitive Energy Efficient sensor Network protocol |
| GEAR | Geographic and Energy Aware Routing |
| GPSR | Greedy Perimeter Stateless Routing |
| SPEED | Stateless Protocol for End-to-End Delay |
| SAR | Sequential Assignment Routing |
| Fire-LEACH | Firefly-based Low-Energy Adaptive Clustering Hierarchy |
| NN-LEACH | Neural Network-based Low-Energy Adaptive Clustering Hierarchy |
| AIC (SOM) | Artificial Immune Clustering using Self-Organizing Maps |
| BTCWSN | Balanced Trust Clustering for Wireless Sensor Networks |
| GA-ANFIS | Genetic Algorithm–Adaptive Neuro-Fuzzy Inference System |
| BS-WCA | Base Station–Weighted Clustering Algorithm |

| | |
|---|---|
| EEHRP | Energy-Efficient Hierarchical Routing Protocol |
| CLCP | Circular Layered Clustering Protocol |
| HEED | Hybrid Energy-Efficient Distributed Clustering |
| EAGLE | Energy-Aware Genetic and Learning-based Engine |
| CH | Cluster Head |
| BCH | Backup Cluster Head |
| BS | Base Station |
| AI | artificial Intelligence |
| ML | Machine Learning |
| GA | Genetic Algorithm |
| RL | Reinforcement Learning |
| Q-learning | Quality-learning |
| K-Means | Machine Learning Clustering Algorithm |
| TDMA | Time Division Multiple Access |
| FND | First Node Death |
| HND | Half Nodes Dead |
| GPU | Graphics Processing Unit |
| RAM | Random Access Memory |
| CDF | Energy Distribution Fairness |

# General Introduction

In the era of pervasive digital transformation, **Wireless Sensor Networks (WSNs)** have emerged as a foundational technology, enabling autonomous data collection and real-time environmental monitoring across a wide range of critical applications—including smart agriculture, industrial automation, structural health monitoring, and environmental surveillance. These networks are typically composed of a large number of distributed sensor nodes equipped with sensing, processing, and wireless communication capabilities.

Despite their growing relevance, WSNs are fundamentally constrained by the limited energy resources of individual nodes. Once deployed—often in inaccessible or hostile environments—these battery-powered nodes are rarely replaced or recharged, making **energy efficiency** a central concern in WSN design. In addition to energy limitations, WSNs must also address challenges such as **network scalability**, **topological robustness**, and **data reliability**, especially in large-scale and dynamic deployments.

To mitigate these issues, one promising technique is **clustering**, wherein sensor nodes are organized into logical groups led by **Cluster Heads (CHs)**. CHs perform local data aggregation and serve as relays to the Base Station (BS), thereby reducing redundant transmissions and conserving energy. However, conventional clustering algorithms often suffer from **suboptimal CH selection**, **unbalanced energy consumption**, and **early node depletion**, particularly near the BS, where communication loads are concentrated.

Recent research has increasingly adopted **Artificial Intelligence (AI)**-driven approaches to enhance WSN performance, particularly those that incorporate **Reinforcement Learning (RL)** and **Evolutionary Computation**. These intelligent methods provide adaptability to environmental changes and allow the system to learn optimal operational strategies over time. Yet, many existing methods either lack architectural scalability or fail to balance energy and load distribution effectively.

In this context, in this thesis we introduces our protocol **EAGLE**—an **Energy-Aware Genetic and Learning-based Architecture**—designed to significantly extend the operational lifespan and efficiency of WSNs. The proposed protocol integrates two synergistic AI techniques:

- **Genetic Algorithms (GA)** are employed to perform optimal and zone-specific selection of Cluster Heads (CH) and Backup Cluster Heads, ensuring energy-efficient, centrally located leadership within each network zone.
- **Q-learning**, a model-free reinforcement learning method, is used to enable each CH to adaptively determine the most efficient routing path toward the BS, considering both energy costs and topological dynamics.

Our EAGLE's architecture is built upon a **layered and zoned network topology**, where hierarchical clustering and learning-based routing are coupled with real-time energy awareness and failure resilience. Notably, the design includes the use of **backup CHs** and intelligent fallback mechanisms, which improve fault tolerance and continuity of service. This hybrid AI-driven strategy allows EAGLE to maintain **network stability**, enhance **energy fairness**, and scale efficiently with increasing network size and complexity.

## Thesis Structure

This document is organized into the following chapters:

- **Chapter 1:** Fundamental Concepts and Challenges in Wireless Sensor Networks
  Presents the architectural models, energy models, and core issues affecting the performance of WSNs.
- **Chapter 2:** State of the Art in WSN Optimization Approaches
  Reviews existing clustering, routing, and AI-based techniques in the literature and identifies limitations that motivate the proposed solution.
- **Chapter 3:** Proposed our protocol Energy-Aware Genetic and Learning-Based Architecture (EAGLE)

Provides a detailed exposition of the proposed hybrid model, including network structuring, GA-based CH election, Q-learning routing, and system adaptivity.

- **Chapter 4:** Simulation and Results
  Demonstrates the effectiveness of EAGLE through a comprehensive set of simulations, comparing it against a traditional K-means clustering protocol across multiple performance metrics.

- **Conclusion :** General conclusion and Perspectives

Chapter 1

# Fundamental concepts and challenges in Wireless Sensor Networks

## 1.1 Introduction to Wireless Sensor Networks

WSNs represent a paradigm shift in distributed sensing and data acquisition systems. These networks consist of spatially distributed autonomous devices equipped with sensors to cooperatively monitor physical or environmental conditions [4]. The fundamental components of a WSN include sensor nodes, base stations, and communication infrastructure, working in unison to collect, process, and transmit data from the monitored environment to end-users [5].

The evolution of WSNs has been driven by advancements in micro-electro-mechanical systems (MEMS), wireless communications, and digital electronics. Modern sensor nodes integrate sensing, data processing, and wireless communication capabilities in compact, low-power devices [6]. This technological convergence has enabled the deployment of WSNs in diverse application domains, including:

- **Environmental monitoring**: Tracking climate conditions, pollution levels, and wildlife habitats [4]
- **Industrial automation**: Monitoring equipment status and process variables in manufacturing plants [7]
- **Healthcare systems**: Remote patient monitoring and emergency response systems [8]

- **Military applications**: Battlefield surveillance and intrusion detection [4]
- **Smart infrastructure**: Structural health monitoring of bridges and buildings [9]

Despite their widespread adoption, WSNs present unique challenges that stem from their constrained resources and often harsh deployment environments. The fundamental limitations include finite energy supplies, limited computational capabilities, restricted communication bandwidth, and the need for robust operation in potentially adverse conditions [10]. These constraints necessitate careful design of network protocols and algorithms to ensure efficient operation throughout the network lifetime.

## 1.2 Architectural components of WSNs

### 1.2.1 Sensor node architecture

A typical wireless sensor node comprises four primary subsystems [5], and may include additional optional components depending on application needs [11]:

1. **Sensing unit**:
    - Composed of sensors (temperature, humidity, pressure, etc.) and analog-to-digital converters (ADCs)
    - Responsible for data acquisition from the physical environment
2. **Processing unit**:
    - Consists of a microcontroller or microprocessor
    - Executes the node's operating system and application programs
    - Performs local data processing and aggregation
    - Manages all components of the sensor node
3. **Communication unit**:
    - Contains a radio transceiver for wireless communication
    - Handles modulation, encoding, and transmission/reception of data
4. **Power unit**:

- Typically battery-powered with possible energy harvesting
- May include power management circuitry
- Primary constraint on node lifetime and performance

5. **Location Finding System** (optional):
    - Determines the geographical position of the sensor node
    - Can use GPS modules or localization algorithms based on signal strength, angle of arrival, or time of flight
    - Essential for applications requiring spatial context, such as environmental monitoring, target tracking, and geographic routing

6. **Mobilizer** (optional):
    - Enables physical repositioning of the sensor node
    - May be implemented using actuators, robotic platforms, or drones
    - Used in dynamic environments to improve coverage, connectivity, or tracking accuracy
    - Increases deployment flexibility but adds energy and mechanical complexity

7. **Power Generator** (optional):
    - Supplements or replaces the battery as an energy source
    - Includes energy harvesting mechanisms such as solar panels, piezoelectric elements, wind turbines, or thermoelectric generators
    - Helps extend network lifetime and supports sustainable operation, especially in remote or inaccessible locations

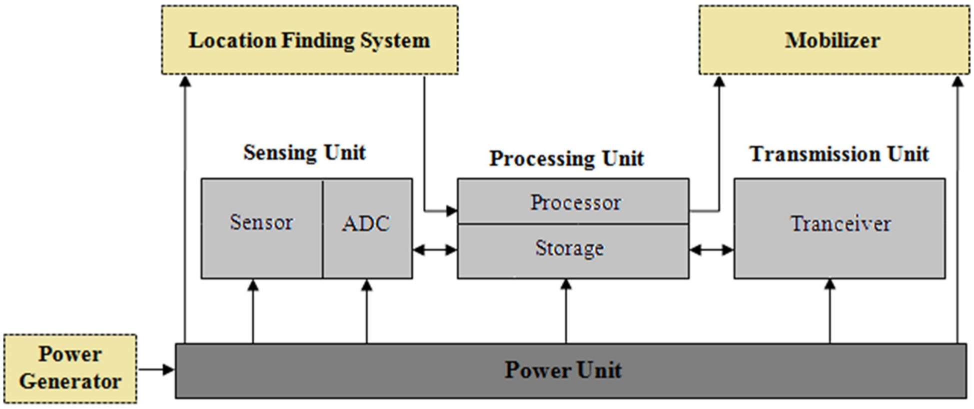


Figure 1.1: Sensor Node Architecture [1]

### 1.2.2 Network topologies

WSNs can be organized in various topological configurations, each with distinct advantages and limitations [6]:

1. **Flat architectures**:

   Flat architectures represent the simplest form of WSN organization where all nodes are treated equally. This approach is particularly useful for small-scale deployments where network management simplicity is prioritized over advanced functionality.

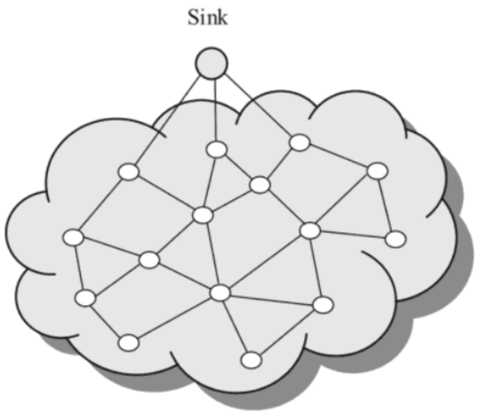


Figure 1.2: Flat network architecture [2]

   - All nodes have equal roles and responsibilities
   - Simple to implement but scales poorly

2. **Hierarchical (Heterogeneous) architectures**:

Hierarchical architectures introduce structured organization to WSNs by creating different tiers of nodes. This design significantly improves network scalability and energy efficiency, making it suitable for larger sensor deployments with varying node capabilities.

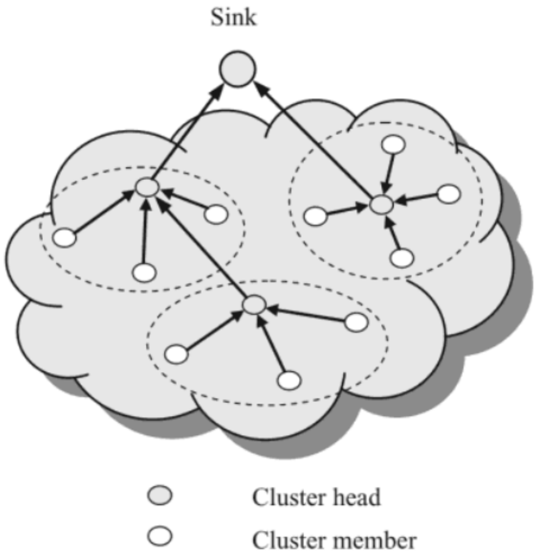


Figure 1.3: Multihop clustering architecture [2]

- Nodes organized in tiers or clusters
- Enables data aggregation and energy efficiency

3. **Mobile WSNs**:

- Some or all nodes are mobile, allowing dynamic network reconfiguration.
- Used in applications like wildlife tracking and disaster response.

Table 1.1: Comparison of Flat vs. Hierarchical network architectures

| **Feature** | **Flat architecture** | **Hierarchical architecture** |
|---|---|---|
| **Structure** | All nodes have equal status and functionality | Nodes are organized into clusters with designated Cluster Heads (CHs) |
| **Energy efficiency** | Less efficient (multi-hop routing increases energy consumption) | More efficient (CHs perform data aggregation, reducing transmissions) |
| **Scalability** | Limited scalability for large networks | Highly scalable for large networks |
| **Example protocols** | Directed Diffusion, SPIN | LEACH, PEGASIS, TEEN |

The choice of network architecture significantly impacts energy consumption, scalability, and overall network performance [12].

## 1.3 Energy constraints and optimization challenges

### 1.3.1 Energy consumption profile

Energy consumption in WSNs occurs primarily in three domains [10]:

1. **Sensing operations**
   - Power requirements vary by sensor type
   - Active sensors (e.g., cameras) consume more than passive ones
   - Sampling rate significantly impacts energy use
2. **Data processing**
   - Includes computation and storage operations
   - Generally consumes less energy than communication
   - Energy proportional to computational complexity

3. **Wireless communication**
   - Dominant energy consumer (often >70% of total)
   - Transmission energy proportional to $distance^n$ $(2 \leq n \leq 4)$
   - Reception energy often comparable to transmission

The disproportionate energy cost of communication has led to in-network processing techniques to reduce transmission volume [13].

### 1.3.2 Key energy optimization strategies

Several approaches have been developed to address energy constraints [10, 13]:

- **Duty cycling**
  - Nodes alternate between active and sleep states
  - Reduces idle listening energy waste
  - Requires careful synchronization
- **Data aggregation**
  - Combines related data packets
  - Reduces total number of transmissions
  - May incur processing overhead
- **Topology control**
  - Adjusts transmission power and connectivity
  - Maintains network coverage with minimal energy
  - Requires distributed coordination
- **Energy-aware routing**
  - Selects paths based on residual energy
  - Balances load across the network
  - Extends network lifetime

These strategies often work in combination to achieve optimal energy efficiency. The effectiveness of each approach depends on network size, density, and application requirements [12].

## 1.4 Clustering in WSNs

### 1.4.1 Clustering fundamentals

Clustering has emerged as a predominant approach for organizing WSNs due to its scalability and energy efficiency benefits [14]. The clustering process involves:

1. **Cluster formation:**
    - Network is partitioned into distinct clusters.
    - Each cluster has a Cluster Head (CH) and member nodes.
    - Formation may be based on geographical proximity or other metrics.
2. **Cluster Head (CH) selection:**
    - CHs act as local coordinators
    - Responsible for data aggregation and relaying.
    - Selection criteria may include energy level, node location, or connectivity.
3. **Intra-cluster communication:**
    - Member nodes transmit data to their CH.
    - Typically uses TDMA scheduling to avoid collisions.
    - Power control is often employed to minimize energy use.
4. **Inter-cluster communication:**
    - CHs forward processed data to the base station.
    - May involve single-hop or multi-hop routing.
    - Often forms a backbone network.

### 1.4.2 LEACH protocol

The Low-Energy Adaptive Clustering Hierarchy (LEACH) protocol represents a seminal work in WSN clustering [15]. Its key features include:

- **Self-organizing clusters:**
  - Distributed cluster formation with no centralized control.
  - Adapts to dynamic changes like node additions or failures.
- **Randomized CH rotation:**
  - CH role rotates among nodes to evenly distribute energy load.
  - Uses probabilistic selection based on a predefined CH percentage.
- **TDMA-based medium access:**
  - CH assigns time slots to cluster members.
  - Eliminates contention and collisions.
  - Members can sleep between transmissions.
- **Data aggregation:**
  - CH aggregates and compresses data from member nodes.
  - Reduces total transmissions to the base station and improves energy efficiency.

Despite its importance, LEACH has limitations that motivated further research:

- **Random CH selection** may lead to non-optimal clusters.
- **Single-hop communication** to the base station limits scalability.
- **Uniform energy assumption** does not reflect real-world heterogeneity.
- **Static clustering parameters** hinder adaptation to dynamic network changes.

### 1.4.3 K-means clustering in WSNs

K-means clustering, borrowed from machine learning, provides an alternative approach for WSN organization [12, 14].

- **Algorithm operation:**
  - Partitions nodes into $k$ clusters based on distance minimization.
  - Iteratively refines cluster assignments to improve compactness.
- **Advantages for WSNs:**
  - Forms geographically compact clusters.
  - Reduces intra-cluster communication distance.
  - Offers more stability than random clustering.
- **Implementation challenges:**
  - Requires knowledge of node locations.
  - High computation overhead in large-scale networks.
  - Optimal value of $k$ must be determined adaptively.

The integration of K-means with LEACH and other techniques has shown promise in addressing energy and scalability limitations.

## 1.5 Routing protocols in WSNs

### 1.5.1 Protocol Classification

WSN routing protocols can be categorized based on their operational strategy [16]

1. **Data-centric protocols:**
   - Focus on the data rather than node identities.
   - Employ attribute-based naming.
   - Examples: SPIN, Directed Diffusion.

2. **Hierarchical protocols:**
    - Use layered architectures such as clusters or trees.
    - Enable data aggregation and efficient communication.
    - Examples: LEACH, PEGASIS.
3. **Location-based protocols:**
    - Use geographic information to guide routing.
    - Support energy-aware and distance-based forwarding.
    - Examples: GEAR, GPSR.
4. **QoS-aware protocols:**
    - Consider latency, reliability, and other Quality of Service metrics.
    - Balance trade-offs between energy and performance.
    - Examples: SPEED, SAR.

### 1.5.2 Energy-aware routing considerations

Effective WSN routing must address multiple energy-related constraints [16, 6]:

- **Energy imbalance:**
    - Nodes closer to the base station relay more traffic.
    - Leads to faster energy depletion and ”energy holes.”
- **Path eelection:**
    - Shortest path may not be energy-optimal.
    - Must consider residual energy and load balancing.
- **Topology changes:**
    - Node failures due to energy depletion cause network dynamics.
    - Routing must adapt to maintain connectivity and performance.

A variety of energy-aware and adaptive protocols have been developed to address these issues and improve the lifetime and robustness of WSN deployments.

## 1.6 Foundational Artificial Intelligence (AI) techniques

In recent years, artificial intelligence (AI) techniques have become integral to solving complex optimization and decision-making problems in distributed systems. Two such techniques—**Genetic Algorithms (GAs)** and **Q-learning**—have proven particularly effective in environments where classical rule-based methods fail due to dynamic, high-dimensional, or partially observable characteristics. This section presents an in-depth technical overview of both algorithms to establish the theoretical foundation upon which advanced solutions are built.

### 1.6.1 Genetic Algorithms (GA)

GeAs are adaptive heuristic search algorithms based on the principles of natural selection and genetics. Originally proposed by Holland in 1975 [17], GAs are widely used to solve combinatorial, multi-objective, and non-linear optimization problems where gradient-based methods are insufficient or inapplicable.

A typical GA follows these steps:

- **Encoding:** Candidate solutions (chromosomes) are encoded, usually as binary strings or real-valued vectors. Each gene corresponds to a decision variable.
- **Initialization:** A diverse population is initialized, often randomly, to cover a broad search space.
- **Fitness evaluation:** Each chromosome is evaluated using a problem-specific fitness function to determine its suitability.
- **Selection:** High-fitness individuals are probabilistically selected as parents using methods such as roulette wheel, tournament, or rank-based selection.
- **Crossover:** Pairs of parents undergo recombination to produce offspring. This introduces new patterns into the population. Techniques include single-point, two-point, and uniform crossover.
- **Mutation:** With low probability, random genes are altered to maintain genetic diversity and avoid local optima.

- **Replacement:** The next generation is formed, typically by replacing the worst-performing individuals with the new offspring.

Mathematically, the evolution of the population from generation $t$ to $t+1$ is summarized as:

$$\text{Population}_{t+1} = \text{Selection}(\text{Mutation}(\text{Crossover}(\text{Population}_t)))$$

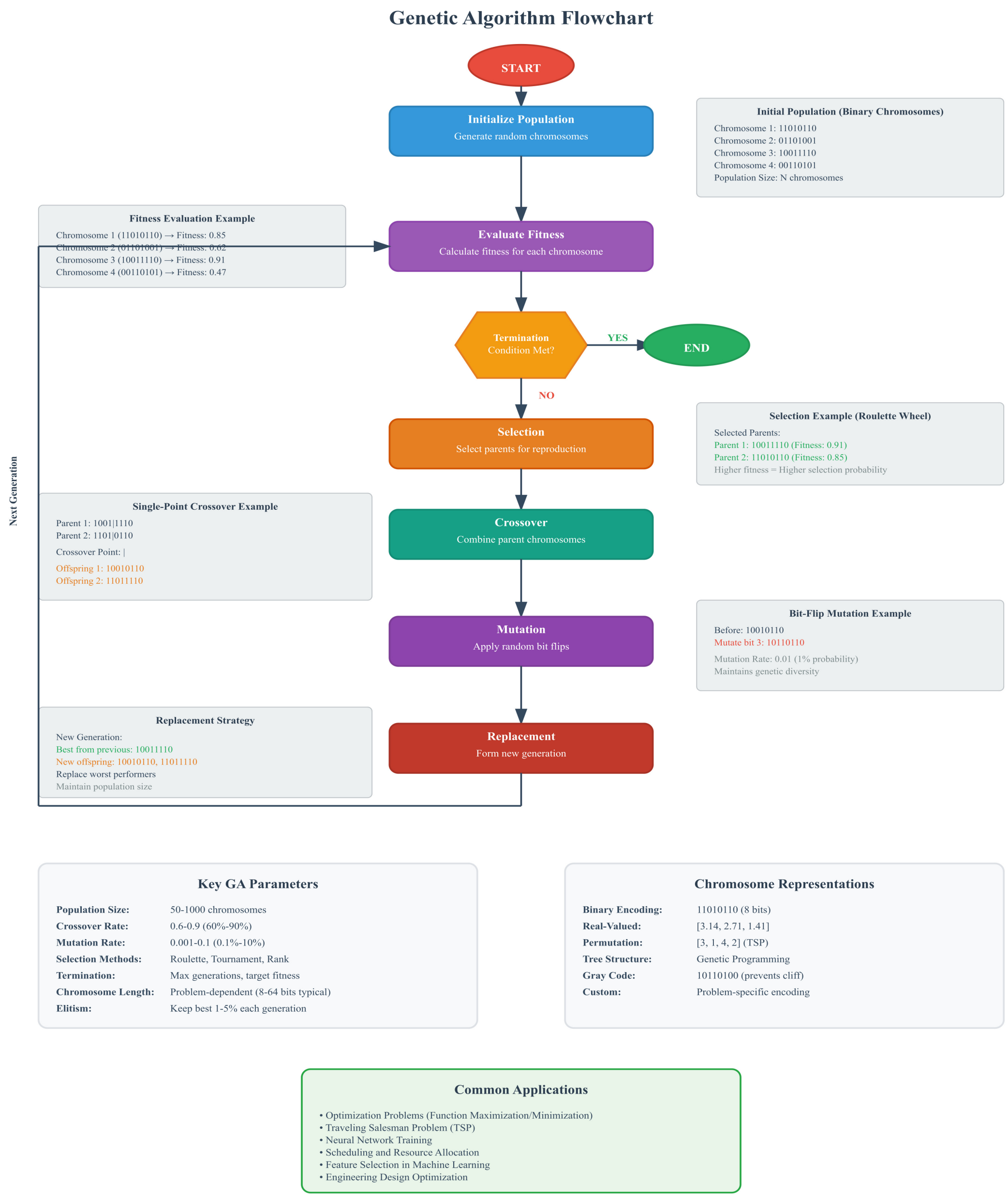


Figure 1.4: Detailed Genetic Algorithm Flowchart with Binary Chromosome Examples

GAs are stochastic and population-based, which allows them to explore large and complex search spaces robustly and effectively.

### 1.6.2 Q-Learning

Q-learning is a model-free reinforcement learning algorithm that enables agents to learn optimal policies through trial-and-error interactions with their environment. Initially proposed by Watkins in 1989 and later extended by Watkins and Dayan in 1992 [18], it is particularly useful in scenarios with unknown or partially observable dynamics.

The goal of Q-learning is to estimate the optimal action-value function $Q^*(s, a)$, which represents the maximum expected cumulative reward that an agent can obtain by taking action $a$ in state $s$ and following the optimal policy thereafter.

At each time step, the agent performs the following:

1. Observe the current state $s$
2. Select an action $a$ using an exploration strategy (e.g., $\epsilon$-greedy)
3. Execute the action and receive a reward $r$, and transition to the new state $s'$
4. Update the Q-value as follows:

$$Q(s, a) \leftarrow Q(s, a) + \alpha \left[ r + \gamma \max_{a'} Q(s', a') - Q(s, a) \right]$$

where:

- $\alpha$ is the learning rate ($0 < \alpha \leq 1$)
- $\gamma$ is the discount factor ($0 \leq \gamma \leq 1$)
- $r$ is the immediate reward
- $\max_{a'} Q(s', a')$ is the estimate of the best future reward achievable from state $s'$

Q-learning is considered an off-policy algorithm because it evaluates and improves the target policy independently of the agent's current actions. It is proven to converge to the optimal policy under the assumptions of sufficient exploration and decaying learning rates.

Q-learning is especially valuable in real-world environments where transitions are stochastic and models are hard to construct. Its ability to adaptively refine routing and decision-making makes it a foundational technique in many AI-based applications.

## 1.7 Conclusion

This chapter has established the theoretical and practical foundations of WSNs, outlining their core architectural elements, energy constraints, and communication strategies. Particular attention was given to hierarchical network organization, clustering techniques, and routing protocols, including well-known methods such as LEACH and K-means clustering. These traditional approaches have been shown to offer partial solutions to energy efficiency and scalability, but they often fall short in adaptability and robustness under dynamic network conditions.

In response to these limitations, this chapter introduced two powerful artificial intelligence techniques GA and Q-learning as promising tools to address the shortcomings of conventional methods. GA offers a robust mechanism for energy-aware cluster head selection through population-based optimization, while Q-learning enables adaptive routing by allowing nodes to learn from past decisions and environmental feedback.

Together, these AI paradigms provide a compelling foundation for developing intelligent and self-optimizing protocols tailored to the evolving demands of WSNs. The concepts introduced here will serve as the basis for the proposed hybrid approach detailed in subsequent chapters, where GA and Q-learning are integrated into a unified framework aimed at significantly improving the efficiency, scalability, and resilience of WSN operations.

Chapter 2

# State of the Art in WSN optimization approaches

## 2.1 Introduction

With the continuous evolution of WSNs, optimizing network longevity, energy efficiency, and secure data transmission has become a central research objective. The limitations in energy, bandwidth, and computational power necessitate intelligent routing and clustering algorithms. In recent years, advanced methodologies using **machine learning**, **artificial intelligence**, and **bio-inspired techniques** have been proposed to improve the performance of WSNs, both from **energy consumption** and **security perspectives**.

This chapter reviews state-of-the-art clustering and routing approaches in WSNs, including traditional energy-centric algorithms, AI-enhanced protocols, and secure trust-aware models. We place particular focus on two recent proposals: the **Energy-Efficient Hierarchical Routing Protocol (EEHRP)** using K-Means clustering, and the **Circular Layered Clustering Protocol (CLCP)**. These are contrasted with established solutions such as LEACH, HEED, PSOHC, and others.

## 2.2 Related Works

Table 2.1: Comparative Analysis of WSN Clustering Protocols

| **Protocol** | **Energy Efficiency** | **Security Focus** | **AI/ML-Based** | **CH Selection Strategy** | **Scalability** | **Remarks** |
|---|---|---|---|---|---|---|
| LEACH | Medium | No | No | Random rotation | Low | Widely used baseline |
| Fire-LEACH | High | No | Yes (FA) | Signal + energy | Medium | Improved clustering |
| NN-LEACH | High | No | Yes (NN) | Learned energy profile | Medium | Needs training data |
| HEED | High | No | No | Energy + communication cost | High | Distributed protocol |
| EEHRP | Very High | No | Yes (K-Means) | Distance + energy score | High | Layer-based clusters |
| AIC (SOM) | Very High | Partial | Yes (SOM) | Self-learning clustering | High | Predictive routing |
| BTCWSN | Medium | Yes | Partial | Bio-inspired + trust | Medium | Security enhancement |
| GA-ANFIS | High | No | Yes | Genetic + fuzzy logic | Medium | Adaptive decision-making |
| BS-WCA | High | Yes | No | Behavior + energy | Medium | Trust-integrated clustering |
| CLCP | High | No | Yes (K-Means) | Distance + energy (radial-based) | Medium | Balanced CH load near BS |

### 2.2.1 Clustering based on energy efficiency

1. **LEACH**

   LEACH is one of the foundational hierarchical protocols in WSNs. It divides the network into clusters where randomly selected CHs aggregate and forward data to the base station (BS). While energy-efficient due to reduced long-range transmissions, it suffers from unbalanced CH distribution and scalability limitations [19].

2. **Firefly-LEACH (Fire-LEACH)**
   Fire-LEACH integrates the Firefly Algorithm (FA), a nature-inspired meta-heuristic, to enhance CH selection. By simulating firefly attraction behavior based on signal strength and residual energy, this protocol achieves more robust clustering and longer network lifespan [20].

3. **NN-LEACH (Neural Network LEACH)**
   NN-LEACH employs an artificial neural network with three layers (input, hidden, output) to predict optimal CHs based on node energy levels. This model improves over LEACH by learning optimal energy distributions for CH selection [21].

4. **HEED (Hybrid Energy-Efficient Distributed Clustering)**
   HEED considers residual energy and communication cost for CH selection. Unlike LEACH, it avoids randomness in CH rotation and achieves better load balancing and energy distribution [22].

5. **EEHRP (Energy-Efficient Hierarchical Routing Protocol)**
   EEHRP introduces a layered hierarchical architecture in which the network is divided into concentric layers. Each layer uses K-Means clustering to form compact clusters. Cluster Head (CH) selection is based on a weighted score that considers both residual energy and distance to the cluster centroid. Intra-cluster communication is managed through TDMA, while inter-cluster communication uses multi-hop routing from outer to inner layers [3].

6. **CLCP (Circular Layered Clustering Protocol)**
   This model organizes the network into concentric circular levels. Each level forms K-Means clusters independently. More clusters are placed closer to the BS to distribute forwarding load. CH selection uses a combination of distance to centroid and energy levels. Data is routed from outer rings to inner ones, ultimately reaching the BS [23].

### 2.2.2 Clustering based on AI and ML

1. **AIC (Artificial Intelligence-based Clustering with SOMs)**

   AIC leverages Self-Organizing Maps (SOMs), a type of unsupervised neural network, to cluster sensor nodes based on energy profiles and spatial proximity. SOMs dynamically form clusters and predict the most energy-efficient paths based on historical data [24].

2. **GA-ANFIS (Genetic Algorithm with Adaptive Neuro-Fuzzy Inference System)**

   This hybrid approach combines genetic algorithms and fuzzy logic to manage clustering adaptively in dynamic WSN environments [25].

### 2.2.3 Clustering based on trust and security

1. **PSO-HC (Particle Swarm Optimization for Hierarchical Clustering)**

   Employs PSO to minimize the number of CHs while ensuring balanced load distribution. This optimization reduces routing overhead and extends the network lifetime [26].

2. **BTCWSN (Bio-Inspired Trust Clustering for WSNs)**

   **BTCWSN** introduces bio-inspired techniques to address vulnerabilities in traditional protocols like LEACH. It guards against attacks such as Sybil or wormhole by introducing secure clustering based on trust metrics [27].

3. **BS-WCA (Balanced and Safe Weighted Clustering Algorithm)**

   Selects CHs based on energy, connectivity, and behavioral trust. It ensures secure data aggregation and avoids compromised nodes acting as CHs. This results in balanced energy usage and higher network resilience [28].

4. **SD-CA (Spatially Distributed Collaborative Anomaly Detection)**

   Combines local anomaly detection with distributed collaborative filtering. It addresses communication overhead and single-point failures in centralized systems, effectively identifying misbehaving nodes [29].

5. **WCA and DWCAr (Weighted Clustering Algorithms for MANETs)**

   Both **WCA** and its distributed variant **DWCAr** aim to reduce communication costs and increase stability in mobile networks. They select CHs using metrics like residual energy, transmission range, and node mobility [30, 31].

## 2.3 Detail of key protocols

### 2.3.1 EEHRP

#### a. Motivation

Traditional protocols such as LEACH suffer from poor scalability, unbalanced CH selection, and limited cluster distribution — especially in large-scale deployments. Even LEACH's successors, like LCHREP, attempt to organize nodes into layers but still lack optimized clustering and suffer from energy depletion near the base station (BS), leading to **hotspots**.

The **EEHRP** [3] introduces two key innovations:

- **Layer-wise K-Means clustering**, instead of random CH assignment
- **Distance- and energy-based CH selection**, instead of random or energy-only selection

This results in:

- Balanced cluster sizes per layer
- Reduced inter-node communication cost
- Significantly prolonged network lifetime

#### b. Network Architecture

The network is partitioned into **L concentric layers** (zones) centered around the BS. Each layer has a different number of nodes and hence, a different number of clusters.

- Nodes in each layer are grouped using **K-Means**, producing $K(l)$ clusters, where the number of clusters increases as we approach the BS (to alleviate the relay burden on those nodes).
- The **Base Station (BS)** controls the clustering and CH selection process and is assumed to know the node positions and initial energy levels.

### c. Configuration Phase

This phase establishes the network's topology and is executed only once at the beginning.

**1. Layer Formation** The BS measures the farthest distance $\delta$ from all nodes and computes:

$$L = \left\lfloor \frac{\delta}{Tr_{min}} \right\rfloor$$

Each node $sni$ is assigned to a layer $l$ based on:

$$l = \left\lfloor \frac{d(sni, BS)}{Tr_{min}} \right\rfloor$$

**2. Cluster Formation** For each layer $l$, the number of clusters $K(l)$ is determined by:

$$K(l) = \left\lceil \frac{n(l)}{l \cdot L} \right\rceil$$

Where $n(l)$ is the number of nodes in layer $l$. The BS applies **K-Means** within each layer to minimize intra-cluster distances.

**3. CH Selection** For each cluster, the BS calculates a **score** for every node using:

$$\text{score}_i = \alpha \cdot d(sni, \text{centroid}) + \beta \cdot \frac{1}{E_i}$$

Where:

- $d(sni, \text{centroid})$: distance from node to cluster centroid
- $E_i$: residual energy
- $\alpha, \beta$: tunable weights

The node with the lowest score becomes the CH.

### d. Routing Phase

- **Intra-cluster communication** uses **TDMA** to reduce collisions and allow non-CH nodes to sleep when idle.
- **Inter-cluster routing** is **multi-hop**: CHs in higher-numbered (farther) layers forward their data to CHs in lower-numbered (closer) layers.

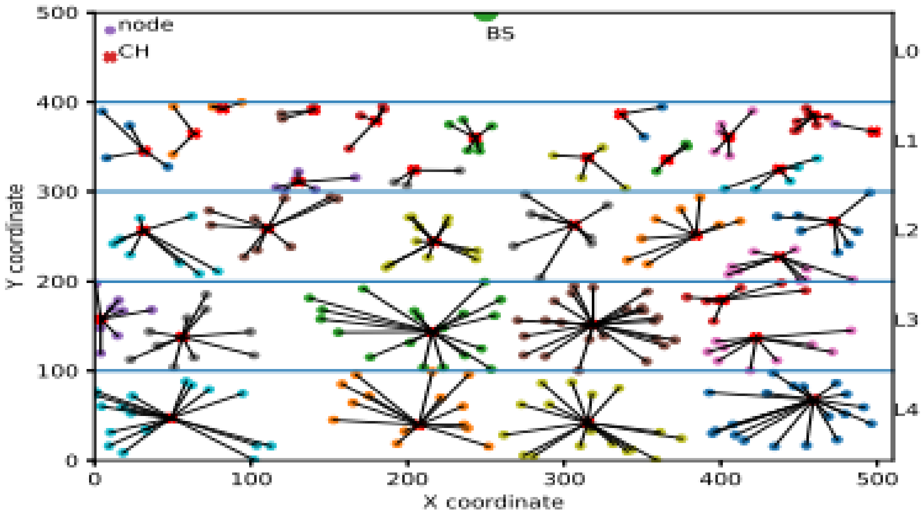


Figure 2.1: Hierarchical Clustering using EEHRP [3]

This approach:

- Minimizes energy used in long transmissions
- Balances energy usage by increasing CH density near BS

### e. Maintenance Phase

Over time, CHs may deplete their energy. The BS initiates re-clustering or simply re-assigns CHs within existing clusters based on updated residual energy and the scoring formula.

### f. Performance evaluation

In simulation, EEHRP showed:

- **Reduced packet loss** compared to LEACH and LCHREP
- **Higher percentage of alive nodes** after many rounds
- **Better load balancing** among CHs
- **Prolonged network lifetime**

It is particularly effective in **dense and large-scale WSNs**, where random clustering fails to maintain network stability.

### 2.3.2 CLCP

#### a. Motivation

The **CLCP** proposed in the master's thesis by Yahoui Soheib (2024) [23], introduces a **geometrically optimized topology** to solve two key problems:

1. **Hotspot problem**: Nodes near BS drain faster due to high forwarding burden
2. **Inefficient cluster placement**: Random CH positions may result in high communication costs

CLCP organizes the network into **circular layers**, each layer forming **concentric rings** around the BS, ensuring even coverage and efficient routing paths.

#### b. Protocol Overview

The core idea of CLCP is:

- **Divide the network into circular levels (layers)**, each defined by a radial distance from the BS
- Perform **K-Means clustering in each level**, so clusters are geographically compact
- Select CHs based on **distance from centroid** and **available energy**

#### c. Configuration Phase

**1. Level Formation** The network is divided into **rings (levels)** centered on the BS:

- Level 1 = innermost ring, closest to BS
- Level L = outermost ring

This is done using the radio range $R$ as a unit for radial segmentation. Each node measures its distance to BS and determines its level based on:

$$\text{Level}(sni) = \left\lfloor \frac{d(sni, BS)}{R} \right\rfloor$$

**2. Clustering using K-Means**

- Nodes within each level are clustered using **K-Means**, forming localized clusters
- Clusters are **more numerous** in lower (inner) levels to **distribute forwarding load**

**3. CH Selection** Each cluster elects a CH using a **two-criteria optimization**:

- Distance from centroid
- Remaining energy

Same scoring function as EEHRP is applied:

$$\text{score}_i = \alpha \cdot d(sni, \text{centroid}) + \beta \cdot \frac{1}{E_i}$$

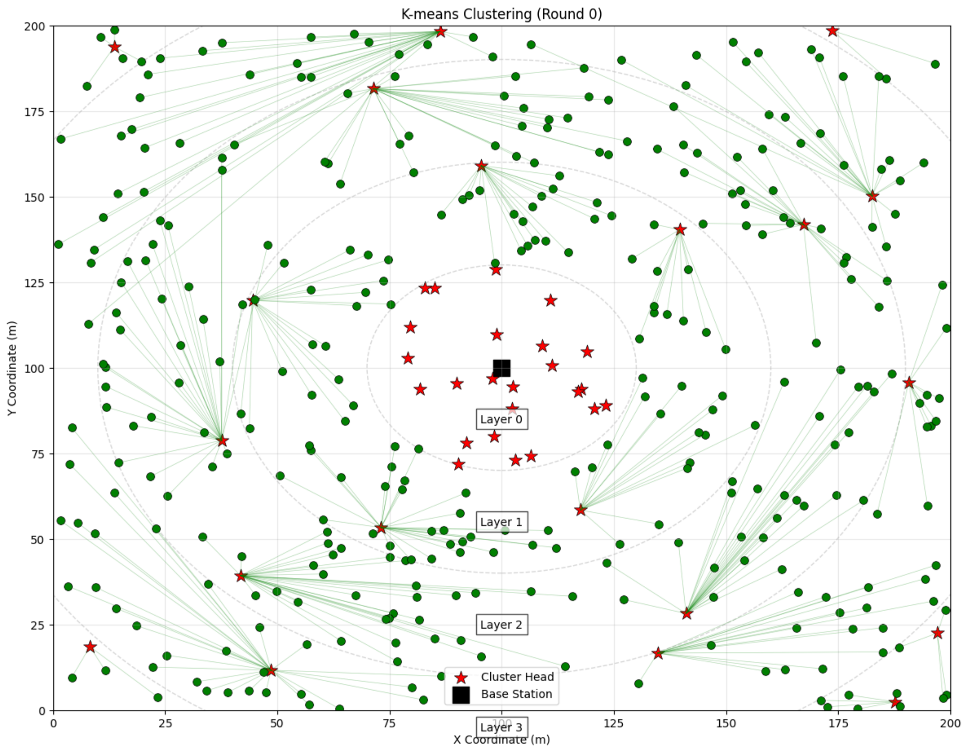


Figure 2.2: Hierarchical structure of CLCP

## d. Routing phase

### Intra-cluster communication

- Nodes send data to their CH in a **TDMA slot**
- Idle listening is minimized to reduce energy use

**Inter-cluster communication**

- CHs forward data to CHs in the level immediately inside (closer to BS)
- This **multi-hop routing** prevents long-range energy-intensive transmissions

**e. Maintenance phase**

- Periodic reassessment of CHs within clusters
- Re-clustering may occur if energy thresholds fall below a minimum level
- Uses residual energy as main indicator for cluster stability

**f. Simulation and Results**

The protocol was simulated using MATLAB. Results compared to LEACH showed:

| **Metric** | **LEACH** | **CLCP** |
|---|---|---|
| Network lifetime | Moderate | **High** |
| Number of alive nodes (after 1000 rounds) | Low | **Significantly Higher** |
| Data packets delivered to BS | Medium | **High** |
| Energy consumption distribution | Uneven | **Balanced** |

**Conclusion:** CLCP is highly efficient for applications requiring **uniform node distribution**, **predictable topology**, and **energy longevity**.

### 2.3.3 Key differences between EEHRP and CLCP

| **Feature** | **EEHRP** | **CLCP** |
|---|---|---|
| Layer Geometry | Linear (concentric by hop count) | Circular (radial distance) |
| Cluster Count Strategy | Based on node density and depth | Based on radial position |
| CH Selection | Based on energy and centroid distance | Same |
| BS Role | Centralized control | Centralized control |
| Adaptability | High (energy-aware updates) | High (periodic reassessment) |

## 2.4 Conclusion

The development of energy-efficient and secure clustering protocols in WSNs has evolved significantly from static heuristics like LEACH to intelligent, learning-based approaches like EEHRP and AIC. While early solutions emphasized basic energy saving, recent protocols integrate AI, fuzzy logic, and trust mechanisms to address both performance and reliability.

This evolution reflects a broader shift from purely heuristic designs to adaptive, data-driven strategies that account for network dynamics, node mobility, and security threats. Protocols like EEHRP and the Circular Layered Clustering Protocol demonstrate how structured layering, informed CH selection, and optimized routing can significantly extend network lifetime and reduce energy consumption. Meanwhile, AI-driven models like AIC and GA-ANFIS show promise in real-time adaptation and predictive routing.

Collectively, these approaches point toward a new generation of WSN protocols that combine energy efficiency, scalability, and resilience. These methods form the foundation for the proposed model described in the following chapter.

Chapter 3

# Proposed approach: Energy-Aware-routing with Genetic and Learning-based Engine (EAGLE)

## 3.1 Introduction

WSNs represent an indispensable component in a broad array of intelligent systems, ranging from environmental monitoring and industrial automation to defense and health applications. These networks consist of spatially distributed sensor nodes, each equipped with sensing, processing, and communication capabilities. However, the inherent constraints in energy availability, particularly the inability to recharge or replace batteries in many deployment scenarios, necessitate the development of highly energy-efficient communication protocols.

our protocol EAGLE emerges as a novel solution aimed at maximizing the operational longevity of WSNs while maintaining robust data transmission reliability. This protocol adopts a hybrid artificial intelligence-based approach that fuses evolutionary optimization with reinforcement learning in a structured, hierarchical framework. Specifically, it utilizes a layered and zoned network organization in which energy-aware CH selection is optimized via GA, and routing decisions are refined through Q-learning-based adaptation. The synergy of these techniques allows EAGLE protocol to dynamically adapt to changing network conditions, distribute energy consumption more uniformly, and mitigate the effects of node failures.

This chapter presents a comprehensive exposition of our proposal protocol EAGLE, encompassing its architectural framework, strategic mechanisms, operational phases, and the intelligent algorithms that govern its decision-making processes.

## 3.2 General description of the proposed approach

The EAGLE protocol is designed with the overarching goal of prolonging the network lifetime of WSNs without sacrificing performance or scalability. The protocol's architecture is predicated on four foundational principles:

1. **Topological hierarchy and zoning**: The sensor field is logically divided into concentric circular layers centered around the BS. Each layer is further partitioned into angular zones, facilitating local decision-making and reducing global communication overhead.

2. **Intelligent CH / Backup election**: Within each zone, two nodes are selected to serve as the CH and the backup CH. The selection process is driven by an evolutionary optimization strategy that prioritizes high-energy nodes with advantageous spatial positions.

3. **Adaptive multi-hop routing**: Rather than relying on static routing paths, CHs dynamically select forwarding nodes based on a learned model that reflects past transmission outcomes and energy efficiencies. This learning-based routing ensures minimal energy expenditure and optimal path selection over time.

4. **Fault tolerance and maintenance**: The protocol continuously monitors the viability of CHs. Backup CHs are proactively promoted when necessary, and emergency replacements are appointed using energy-aware heuristics to ensure continuous service and data delivery.

### 3.2.1 Work assumptions

The EAGLE protocol operates under several foundational assumptions to simulate realistic wireless sensor network behavior:

- Sensor nodes are stationary once deployed and uniformly distributed across the sensing field.
- The BS is static and centrally located within the monitoring area.
- All nodes begin with an equal, non-rechargeable energy supply and possess identical communication capabilities.
- Nodes are aware of their geographical positions through GPS or equivalent localization mechanisms.
- Data transmission is time-driven, occurring at fixed intervals.
- All nodes are capable of aggregating data but do so primarily at the cluster head level to reduce redundancy.

These assumptions ensure that the proposed protocol is evaluated under consistent and fair conditions, simulating typical WSN environments encountered in remote or mission-critical deployments.

### 3.2.2 Communication workflow

EAGLE protocol organizes network communication into two sequential phases:

- **Intra-cluster communication**: Regular nodes send sensed data to their CHs. The CHs apply data aggregation techniques to reduce redundancy and compress the payload, thereby conserving energy during subsequent transmissions.
- **Inter-cluster communication**: CHs forward the aggregated data either directly to the BS or to a CH in a lower layer, depending on distance and energy considerations. The routing path is chosen adaptively based on past performance and current energy metrics.

### 3.2.3 Energy model

EAGLE protocol employs a realistic radio energy model to evaluate the energy consumption incurred during data communication. Heinzelman et al. [32] first proposed this energy model in their seminal work on LEACH protocols. The model considers three principal operations: transmission, reception, and data aggregation.

**Transmission energy:**

The energy consumed to transmit a $k$-bit message over a distance $d$ :

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

Where:

- $E_{elec} = 50\,\mathrm{nJ/bit}$ is the energy cost for running transmitter electronics
- $k = 4000\,\mathrm{bit}$ (packet size)
- $\epsilon_{fs} = 10\,\mathrm{pJ/bit/m^2}$ (free-space model)
- $\epsilon_{mp} = 0.0013\,\mathrm{pJ/bit/m^4}$ (multipath model)
- $d_0 = \sqrt{\frac{\epsilon_{fs}}{\epsilon_{mp}}}$ is the critical distance threshold

**Reception energy:**

$$E_{rx}(k) = E_{elec} \cdot k$$

**Aggregation energy:**

$$E_{DA}(k) = E_{DA} \cdot k$$

Where:

- $E_{DA} = 5\,\mathrm{nJ/bit}$ is the data aggregation cost.

These equations guide energy-aware decisions throughout the protocol, ensuring optimal trade-offs between communication cost and network longevity.

## 3.3 Detailed description of EAGLE protocol mechanisms

### 3.3.1 Network structuring and initialization

Our protocol begins with a one-time initialization phase in which the sensor field is logically partitioned into hierarchical regions based on both radial distance and angular orientation relative to the base station (BS). This spatial structuring serves as the foundation for the EAGLE protocol's layered communication and cluster-based decision-making.

The network area is divided into concentric circular **layers** and fixed-angle **zones**:

- **Layering (Radial Partitioning):** Each node computes its Euclidean distance $d_i$ from the BS. Using the minimum communication range $Tr_{min}$, the maximum number of layers is determined as:

$$L_{max} = \left\lceil \frac{\max(d)}{Tr_{min}} \right\rceil$$

Each node is then assigned to a layer $L_i$ based on:

$$L(i) = \min\left( \left\lfloor \frac{d_i}{Tr_{min}} \right\rfloor, L_{max} - 1 \right)$$

This approach ensures that nodes are grouped into concentric regions based on their communication range relative to the BS, allowing for tiered data transmission and scalable CH allocation.

**example calculation:**

Given:

- Base Station (BS) at $(100, 100)$
- Node at $(19.1, 59.3)$
- $Tr_{min} = 30$ meters
- Assume $\max(d) = 150$ meters (network radius)

1. Compute Euclidean distance:

$$\begin{aligned} d_i &= \sqrt{(19.1 - 100)^2 + (59.3 - 100)^2} \\ &= \sqrt{(-80.9)^2 + (-40.7)^2} \\ &= \sqrt{6544.81 + 1656.49} \\ &= \sqrt{8201.3} \\ &\approx 90.56 \text{ meters} \end{aligned}$$

2. Determine maximum layers:

$$L_{max} = \left\lceil \frac{150}{30} \right\rceil = 5 \text{ layers}$$

3. Assign layer:

$$L(i) = \min\left(\left\lfloor \frac{90.56}{30} \right\rfloor, 5 - 1\right)$$
$$= \min(3, 4)$$
$$= 3$$

$$\boxed{\text{Node } (19.1, 59.3) \text{ is assigned to Layer 3}}$$

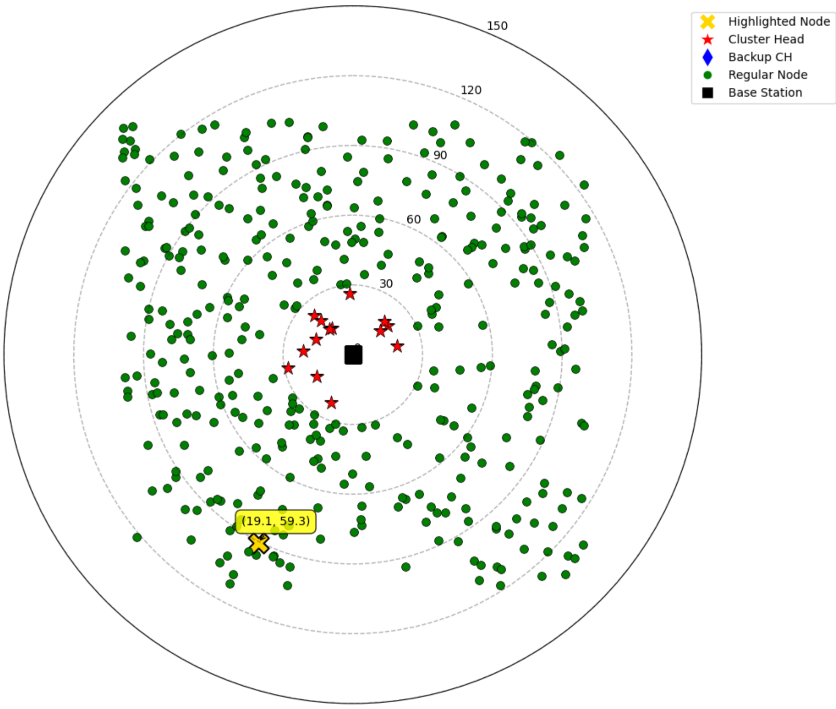


Figure 3.1: EAGLE protocol network structure (LAYERS)

- **Zoning (Angular Partitioning):** While angular zoning has been conceptually applied in previous WSN designs (for example : according to recent developments in zonal clustering [33, 34]), the proposed approach introduces a mathematically explicit and precise formulation for zone assignment, enhancing structural regularity and enabling more localized decision-making.

  We proposed that each node calculates its angular coordinate $\theta_i$ relative to the BS using the standard arctangent function:

  1. **Compute the angle $\theta_i$ (in radians)**

The angle $\theta_i$ is calculated using the two-argument arctangent function (`atan2`) to avoid ambiguity in quadrant determination:

$$\theta_i = \text{atan2}(y_i - y_{\text{BS}}, x_i - x_{\text{BS}}) \tag{3.1}$$

**Input:**

- $(x_i, y_i)$: Coordinates of node $i$
- $(x_{\text{BS}}, y_{\text{BS}})$: Coordinates of the base station

**Output:**

- $\theta_i \in [-\pi, \pi]$ (angle in radians, where $-\pi$ corresponds to the negative x-axis, and $\pi$ wraps back around)

2. **Normalize $\theta_i$ to $[0, 2\pi]$**

   Since `atan2` returns values in $[-\pi, \pi]$, we shift the range to $[0, 2\pi]$ for easier zone mapping:

$$\theta_{\text{norm}} = (\theta_i + \pi) \bmod 2\pi \tag{3.2}$$

   **Purpose:** Ensures all angles are positive and continuous (0° to 360°).

3. **Convert to degrees and assign zone number**

   The normalized angle is converted to degrees and divided into 12 zones, each spanning 30°:

$$Z_i = \left\lfloor \frac{(\theta_i + \pi) \cdot 180}{30 \cdot \pi} \right\rfloor \tag{3.3}$$

   **Explanation:**

   - $(\theta_i + \pi) \cdot 180/\pi$: Converts radians to degrees
   - Division by 30°: Each zone covers 30° (since $360°/12 = 30°$)
   - Floor function ($\lfloor \cdot \rfloor$): Ensures discrete zone assignment:
     * $0° \leq \theta < 30° \Rightarrow$ Zone 0
     * $30° \leq \theta < 60° \Rightarrow$ Zone 1
     * $\vdots$
     * $330° \leq \theta < 360° \Rightarrow$ Zone 11

**Result:**

- $Z_i \in \{0, 1, 2, \ldots, 11\}$, where each $Z_i$ represents a unique angular sector

This equation divides the full 360-degree space around the BS into 12 equal sectors, each spanning 30 degrees. The use of the floor function $\lfloor \cdot \rfloor$ ensures that every node is assigned to a unique angular zone $Z_i \in \{0, 1, \ldots, 11\}$.

The logic behind this division is to localize clustering and routing decisions within bounded angular sectors, which improves energy balancing, reduces redundant transmissions, and simplifies path planning.

Each node is uniquely identified by a pair $(L_i, Z_i)$, which precisely defines its radial and angular placement in the network. This dual-layered structuring significantly enhances scalability, zone-based fault management, and spatially optimized cluster head election.

**example calculation:**

Given:

- Base Station (BS) at $(100, 100)$
- Node at $(19.1, 59.3)$
- 12 zones ($30^\circ$ each)

1. Compute angle:

$$\begin{aligned}
\theta_i &= \text{atan2}(59.3 - 100, 19.1 - 100) \\
&= \text{atan2}(-40.7, -80.9) \\
&\approx -2.678 \text{ radians} \approx -153.43^\circ
\end{aligned}$$

2. Normalize angle:

$$\begin{aligned}
\theta_{norm} &= (-2.678 + 2\pi) \mod 2\pi \\
&= (-2.678 + 6.2832) \mod 6.2832 \\
&= 3.6052 \text{ radians} \approx 206.57^\circ
\end{aligned}$$

3. Assign zone:

$$Z_i = \left\lfloor \frac{206.57^\circ}{30^\circ} \right\rfloor$$
$$= \lfloor 6.886 \rfloor$$
$$= 6$$

$$\boxed{\text{Node } (19.1, 59.3) \text{ is assigned to Zone 6 } (180^\circ - 210^\circ)}$$

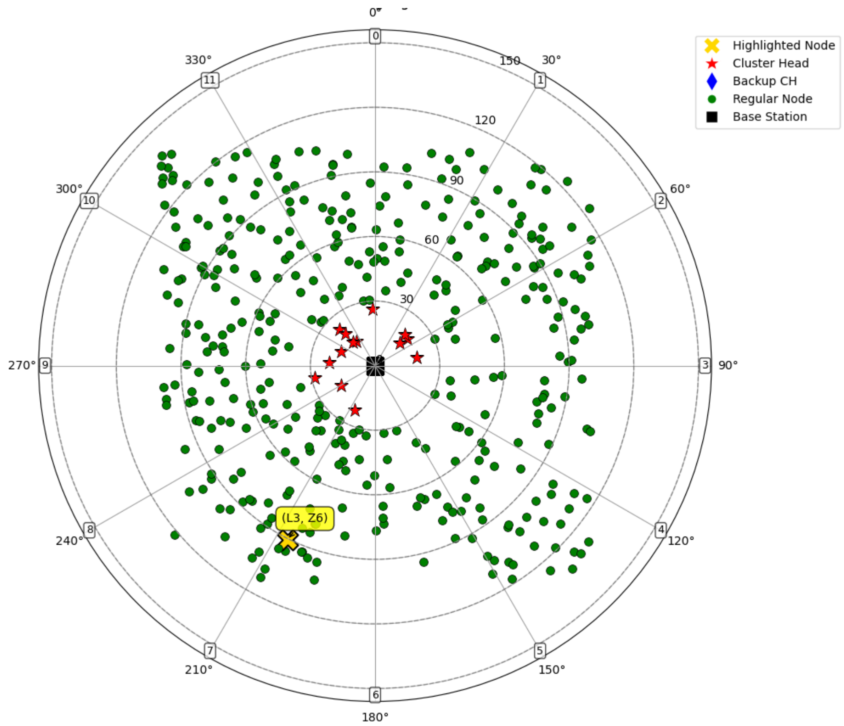


Figure 3.2: EAGLE protocol network structure (LAYERS / ZONES)

The combination of these mathematical partitioning strategies contributes to EAGLE's robust, adaptive, and energy-aware protocol behavior in dynamically deployed sensor networks.

### 3.3.2 Cluster Head and Backup election strategy

The selection of CHs and backup CHs within each zone of the network is a critical component of the protocol's energy management and communication reliability. In our proposed architecture, CH election is governed by a Genetic Algorithm (GA), a population-based evolutionary optimization method inspired by the mechanisms of natural selection, reproduction, and survival of the fittest.

The GA begins by filtering the list of candidate nodes to include only those that are alive and possess sufficient residual energy. A population of candidate pairs $(CH, BCH)$ is then initialized, where each individual consists of two distinct, randomly chosen alive nodes representing a potential CH and its backup. The initial population size is defined as a constant $(P = 20)$.

Each candidate pair is evaluated using a domain-specific fitness function that captures the trade-off between energy availability and spatial centrality. The fitness $F$ of a candidate is given by:

$$F(CH, BCH) = \frac{E_{CH} + E_{BCH}}{d_{CH} + d_{BCH} + \varepsilon}$$

Where:

- $E_{CH}$ and $E_{BCH}$ are the residual energies of the candidate CH and backup CH,
- $d_{CH}$ and $d_{BCH}$ are their respective Euclidean distances to the centroid of the zone, calculated from the positions of all alive nodes in that zone,
- $\varepsilon = 10^{-6}$ is a small constant added to prevent division by zero.

This fitness function favors pairs with high energy and central positions, ensuring that the CH and backup CH are both capable and optimally located within their zone.

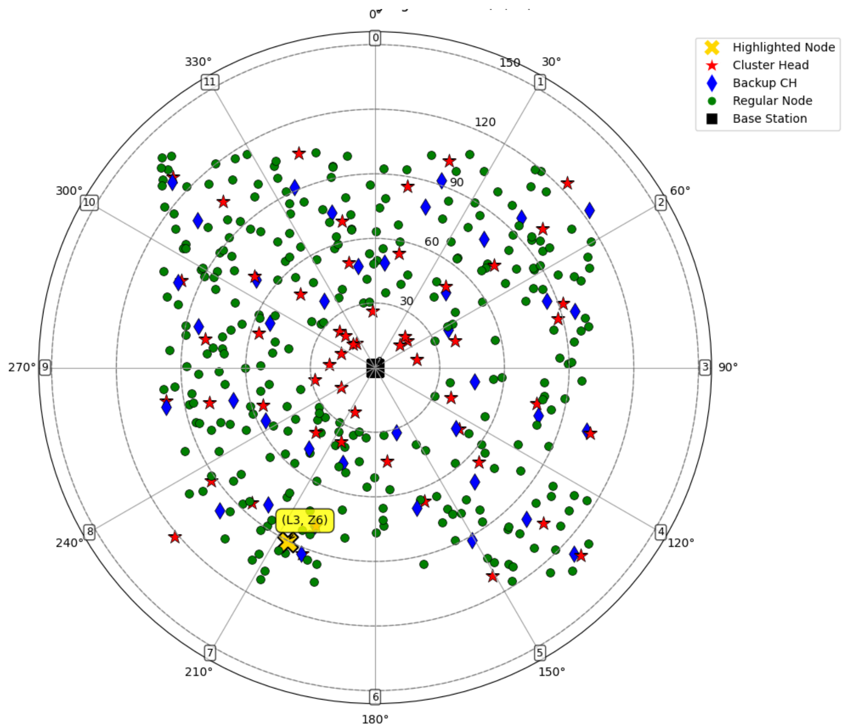


Figure 3.3: EAGLE protocol network structure (CH / Backup)

The evolutionary loop consists of a fixed number of generations ($G = 10$), where each generation comprises the following phases:

1. **Fitness-based selection**: The current population is sorted in descending order of fitness. The top 50% of the population (10 individuals) is retained for reproduction.

2. **Crossover (Reproduction)**: New individuals are generated by randomly selecting pairs from the elite population. Each child is formed by copying either parent1 or parent2 with equal probability (0.5). This retains high-quality traits while enabling recombination of different node combinations.

3. **Mutation**: With a low mutation probability ($\mu = 0.01$), a newly formed individual is replaced by a completely random new pair of alive nodes. This introduces genetic diversity and helps prevent premature convergence.

4. **Population update**: The newly formed offspring are added to the elite set to replenish the full population size (20 individuals). This updated population proceeds to the next generation.

After all generations are completed, the pair with the highest fitness score is selected.

The first node is assigned the role of the primary CH, and the second becomes the backup CH.

This GA-based CH election strategy is computationally lightweight yet highly effective. It ensures dynamic adaptability to real-time network changes, equitable rotation of CH responsibilities, and energy balancing across the network. Because the algorithm simultaneously considers both energy and geometry, it naturally leads to CH placements that reduce intra-zone transmission distances and delay, improving overall energy efficiency.

**Remark.** *In accordance with the hierarchical design of the protocol, all nodes located in the first layer (closest to the base station) are automatically designated as cluster heads. These nodes are exempted from the GA-based election process. Instead, they directly transmit their data to the base station without undergoing further routing or aggregation. This direct transmission approach in the first layer minimizes delay and energy consumption, leveraging the proximity of these nodes to the base station for optimal communication efficiency.*

This intelligent and decentralized mechanism serves as the foundation for the higher layers of adaptive routing and fault-tolerant operations implemented by the protocol.

**Algorithm 1:** Genetic Algorithm for CH and Backup CH Selection in EAGLE

**Input:** Set of alive nodes in a zone: *zone_nodes*

**Output:** Optimal Cluster Head (CH) and Backup CH

**if** *number of alive nodes* $< 2$ **then**
  **return** *(single alive node as CH if available, None)*

Initialize population $P$ with $N$ random pairs $(CH, BackupCH)$ from *zone_nodes*;

**for** $gen = 1$ **to** $GA\_GENERATIONS$ **do**
  Evaluate fitness for each individual $(CH, BackupCH)$ in $P$ using:

$$fitness = \frac{E_{CH} + E_{BackupCH}}{d_{CH} + d_{BackupCH} + \varepsilon}$$

  where $d$ is the distance to the zone centroid;
  Select top 50% of population $P$ as survivors;
  **while** *size of new generation* $< N$ **do**
    Randomly select two parents from top 50%;
    Perform crossover: randomly choose one parent as child;
    **if** *random chance* $< GA\_MUTATION\_RATE$ **then**
      Replace child with new random $(CH, BackupCH)$ pair;
    Add child to new generation;
  Update population $P$ with new generation;

Return the individual $(CH, BackupCH)$ with highest fitness from $P$;

### 3.3.3 Energy-aware CH validation and recovery mechanisms

A crucial component of the EAGLE protocol is its ability to maintain the operational health of CHs and ensure uninterrupted communication within the network. Each CH is responsible not only for receiving and aggregating data from its member nodes but also for forwarding the aggregated data to other CHs or directly to the BS). To fulfill these duties, the CH must possess sufficient residual energy. The validation of CH energy sufficiency is conducted at the beginning of every communication round.

### Energy requirements calculation

The total energy requirement for a CH to operate within a round includes three primary components:

1. **Reception energy**: The CH must be able to receive packets from all its member nodes. For a CH with $n$ members, the total reception energy is:

$$E_{\text{rx-total}} = n \cdot E_{\text{elec}} \cdot k \tag{3.4}$$

   where $E_{\text{elec}}$ is the energy per bit to run the receiver circuitry and $k$ is the size of each packet in bits.

2. **Aggregation energy**: The CH aggregates $n$ received packets, consuming:

$$E_{\text{agg}} = n \cdot E_{\text{DA}} \cdot k \tag{3.5}$$

   where $E_{\text{DA}}$ is the energy cost of data aggregation per bit.

3. **Transmission energy**: The CH then transmits the compressed, aggregated packet to the next hop. Assuming the average transmission distance is $d$, the energy required is:

$$E_{\text{tx}} = E_{\text{elec}} \cdot (k \cdot \beta) + \begin{cases} \epsilon_{\text{fs}} \cdot (k \cdot \beta) \cdot d^2, & \text{if } d < d_0 \\ \epsilon_{\text{mp}} \cdot (k \cdot \beta) \cdot d^4, & \text{otherwise} \end{cases} \tag{3.6}$$

   where $\beta$ is the compression factor, $\epsilon_{\text{fs}}$ and $\epsilon_{\text{mp}}$ are the amplifier constants, and $d_0$ is the threshold distance between free space and multipath propagation models.

### Safety margin requirement

To ensure reliability, the CH is required to have a safety margin of at least 10% more than the calculated total energy for the round:

$$E_{\text{required-total}} = 1.1 \cdot (E_{\text{rx-total}} + E_{\text{agg}} + E_{\text{tx}}) \tag{3.7}$$

### Recovery mechanism

If a CH fails this energy sufficiency check or is otherwise non-functional, the protocol initiates a recovery mechanism:

- **Backup CH promotion**: If a backup CH is available in the same zone and passes the energy sufficiency check, it is promoted to the primary CH role.
- **Emergency CH selection**: If the backup CH is not viable, the protocol selects a new CH from the zone's alive nodes. The node with the highest remaining energy that can fulfill the communication duties is promoted.
- **Zone reconfiguration**: If no suitable CH can be identified, the zone may be temporarily excluded from communication for the current round, reducing overall reliability but preserving energy.

This dynamic and intelligent CH validation and recovery mechanism ensures that the network continues to function optimally under variable energy conditions and node failures, supporting the protocol's goal of maximizing WSN longevity.

This mechanism enhances protocol resilience and reduces communication interruptions due to node failures or energy depletion.

### 3.3.4 Learning-based routing optimization

The routing strategy in EAGLE protocol is governed by a reinforcement learning framework, specifically implemented using Q-learning. This model enables each cluster head (CH) to autonomously and adaptively determine the optimal next-hop node for forwarding aggregated data packets toward the base station (BS), optimizing both energy consumption and reliability over time.

#### Q-Table structure

Each CH maintains a Q-table of dimensions $N \times N$, where $N$ is the number of nodes in the network. Each entry $Q(i, j)$ in this table represents the estimated cumulative reward of selecting node $j$ as the next hop from node $i$. At every decision point, the CH consults this Q-table to choose its next-hop node. However, in each round, the CH is restricted to selecting only the available CHs located in the next hierarchical layer, ensuring energy-efficient and layer-consistent routing.

#### Action selection policy

The action selection process is governed by an $\epsilon$-greedy policy:

- With probability $\epsilon$, the CH randomly selects an available neighbor to explore new routes.
- With probability $1-\epsilon$, it chooses the node with the highest Q-value, thus exploiting the current knowledge.

### Reward calculation

In the reinforcement learning framework implemented within **EAGLE** protocol, the reward assigned to a successful packet transmission is defined as the negative square of the Euclidean distance between the transmitting node and the selected next-hop node. Formally, the reward $r$ is computed as:

$$r = -\|\mathbf{pos}_i - \mathbf{pos}_j\|^2$$

This reward structure is closely aligned with the energy efficiency objectives inherent in wireless sensor networks (WSNs). It reflects the empirical reality that transmission energy consumption grows quadratically with distance under free-space propagation and even more steeply under multipath conditions. As a result, this negative reward formulation naturally biases the learning agent—typically a cluster head (CH)—toward selecting routes that minimize transmission distance and thus conserve energy.

Although this reward function is not novel and has been previously utilized in similar Q-learning-based routing schemes for WSNs, its effectiveness in combination with a hybrid GA and reinforcement learning framework remains substantial. The reward structure used in this work is consistent with formulations found in earlier studies, such as: [35]

### Q-Value update

The Q-value is then updated using the standard temporal difference formula:

$$Q(i,j) \leftarrow Q(i,j) + \alpha \left[ r + \gamma \cdot \max_k Q(j,k) - Q(i,j) \right] \tag{3.8}$$

Where:

- $\alpha$ is the learning rate ($0 < \alpha \leq 1$)
- $\gamma$ is the discount factor ($0 \leq \gamma < 1$)

- $\max_k Q(j,k)$ is the best future reward from node $j$

**example calculation:** The reward is calculated based on the squared Euclidean distance between nodes For instance:

$$\text{Distance from Node 0 to Node 1} = (20-10)^2 + (10-10)^2 = 10^2 = 100 \Rightarrow r = -100$$

$$\text{Distance from Node 0 to Node 3} = (20-10)^2 + (20-10)^2 = 100+100 = 200 \Rightarrow r = -200$$

Table 3.1: Example Q-Table with Rewards Based on Squared Euclidean Distance

| **Q(i, j)** | **Node 0** | **Node 1** | **Node 2** | **Node 3** | **Node 4** |
|---|---|---|---|---|---|
| **Node 0** | – | -100 | -100 | -200 | -400 |
| **Node 1** | -100 | – | -200 | -100 | -100 |
| **Node 2** | -100 | -200 | – | -100 | -500 |
| **Node 3** | -200 | -100 | -100 | – | -200 |
| **Node 4** | -400 | -100 | -500 | -200 | – |

Through repeated learning and refinement, CHs continuously adapt to topological changes and energy dynamics, enabling EAGLE protocol to perform robust, energy-aware, and intelligent routing under dynamic network conditions.

### Q-learning based routing paths

The blue arrows in the network visualization represent the routing decisions made by cluster heads (CHs) based on their learned Q-tables. Each CH selects its next-hop forwarding node by evaluating expected reward values, which are computed as the negative square of the Euclidean distance between the CH and its candidate neighbor.

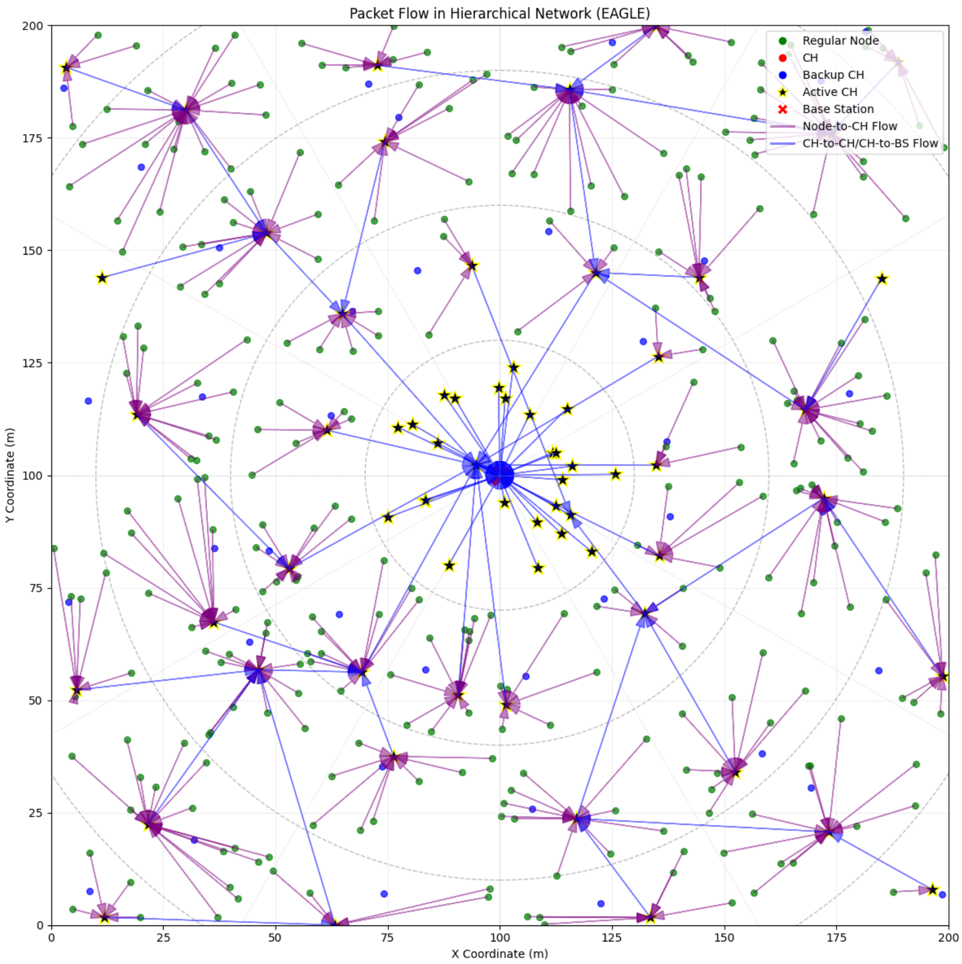


Figure 3.4: EAGLE protocol packet flow

The reward formulation naturally encourages the selection of routing links that are shorter in distance, thereby reducing energy consumption and enhancing link reliability.

Moreover, the observed routing pattern is non-uniform and dynamic, reflecting the protocol's adaptive behavior. Instead of relying on fixed routes, the CHs continuously update their Q-tables based on environmental changes such as node failures or energy depletion. This learning mechanism enables CHs to avoid overused or low-energy neighbors, promoting balanced energy consumption across the network and extending the overall system lifetime.

**Algorithm 2:** Q-Learning-Based Routing for Cluster Heads in EAGLE

**Input:** Current CH ID: *current_ch*;

Available parent CHs in lower layers: $available\_CHs$

**Output:** Selected next-hop parent CH or BS

```
if random() < ϵ then
    Select a random CH from available_CHs (exploration);
else
    For each parent in available_CHs, retrieve Q[current_ch][parent];
    Select CH with highest Q-value (exploitation);
return selected parent CH
```

**Algorithm 3:** Q-Value Update Rule (called after successful transmission)

**Input:** CH ID: *current_ch*, previous parent ID: *old_parent*, new parent ID: *new_parent*, reward: $r$

**Data:** Learning rate: $\alpha$, Discount factor: $\gamma$, Q-table: $Q$

```
oldQ ← Q[current_ch][old_parent];
maxFutureQ ← max_a Q[new_parent][a];
Compute updated Q-value:
```

$$Q[current\_ch][old\_parent] \leftarrow oldQ + \alpha \cdot (r + \gamma \cdot maxFutureQ - oldQ)$$

### 3.3.5 System maintenance and performance monitoring

EAGLE protocol incorporates a robust system maintenance and performance monitoring framework designed to ensure long-term network stability, responsiveness, and adaptiveness in the face of dynamic environmental and operational changes. This subsystem operates in parallel with the core communication and routing procedures and serves to preserve the functional integrity of the network over time.

#### Maintenance phase operations

At the conclusion of each communication round, the protocol executes a structured maintenance phase consisting of several interlinked tasks:

1. **Node role resetting**:

- All nodes are reset to their default states
- Previous designations (CH, backup CH, cluster member) are cleared
- Ensures new elections are based on current conditions

2. **CH re-election**:
   - New CHs and backup CHs selected per zone
   - GA-based optimization considers updated energy levels
   - Maintains balanced energy distribution

3. **Link quality validation**:
   - Nodes verify communication links
   - Check for failures, unavailability, or signal degradation
   - Q-learning selects alternative routes if needed

4. **Energy recalculation and logging**:
   - Remaining energy updated based on round consumption
   - Both round-specific and cumulative energy logged

5. **Failure event handling**:
   - **Dead node detection**:
     - Nodes below minimum threshold marked dead
     - Excluded from subsequent rounds
   - **CH failure logging**:
     - Failed CHs recorded
     - Affected zones flagged for scrutiny
   - **Transmission failures**:
     - Failed attempts logged with reasons
     - Used for Q-learning path optimization

6. **Metric archiving and analysis**: Collected performance indicators include:

- Number of alive nodes
- Packets successfully delivered to BS
- Total energy consumed (round and cumulative)
- CH failures count
- Transmission failures count
- Cumulative packets received at BS

7. **Network health milestones**: EAGLE protocol monitors key lifecycle events:
   - **First Node Death (FND)**: Round when first node depletes energy
   - **Half Node Death (HND)**: Round when 50% nodes are dead
   - **Last Node Death (LND)**: Final network failure round

**Framework benefits**

Through this layered monitoring framework, protocol EAGLE ensures:

- High degree of network self-awareness
- Combination of statistical logging and proactive checks
- Adaptive recovery measures
- Enhanced resilience, efficiency, and sustainability

These mechanisms collectively contribute to the protocol's ability to maintain optimal performance throughout the WSN deployment lifecycle.

## 3.4 Key technical merits of EAGLE

- **Hierarchical modularity**: The combination of layers and zones enables scalable network organization with minimal interdependencies.
- **Evolutionary optimization**: The use of Genetic Algorithms ensures that CHs are both optimally placed and energy-efficient.

- **Intelligent routing**: Reinforcement learning allows CHs to adapt routing behavior in response to environmental and topological changes.
- **High fault tolerance**: The proactive designation of backup CHs and energy-aware recovery strategies ensure continuous operation.
- **Dynamic monitoring**: Integrated performance tracking enables informed protocol evolution and long-term adaptability.

## 3.5 Conclusion

Our protocol EAGLE is a multifaceted, AI-driven protocol tailored to the energy constraints and dynamic nature of WSNs. Its layered design, evolutionary optimization of CHs, and reinforcement learning-based routing converge to offer a resilient, efficient, and adaptive communication framework. Through rigorous validation of energy resources, localized decision-making, and intelligent data forwarding, EAGLE protocol effectively addresses the core limitations of traditional WSN routing schemes. The following chapter we will present empirical evidence evaluating the protocol's performance under diverse simulation conditions, further substantiating its effectiveness and applicability.

Chapter 4

# Simulations and Results

## 4.1 Introduction

This chapter presents a detailed simulation-based evaluation of the proposed EAGLE protocol (Energy-Aware Genetic and Learning-based Architecture) for WSNs. The performance of EAGLE is compared against The CLCP "K-means-based clustering method" to demonstrate its effectiveness in extending network lifetime, optimizing energy usage, and enhancing routing reliability.

The simulations are conducted to assess a comprehensive set of performance metrics including node survivability, energy consumption, packet transmission efficiency, and routing behavior. Each result is thoroughly analyzed, with visual illustrations supporting the observations and detailed discussions explaining the underlying reasons for the observed performance differences.

## 4.2 Simulation environment

### 4.2.1 Platform and Tools

The simulations were executed using **Google Colab**, a cloud-based platform that supports Python-based computational notebooks. The Python 3 runtime environment was used, and the simulations leveraged libraries such as NumPy, Matplotlib, and Math.

**NumPy** [36]**:** A fundamental package for scientific computing in Python, providing support for large, multi-dimensional arrays and matrices, along with mathematical functions to operate on them. Used for vectorized distance calculations and energy computations.

**Matplotlib** [37]**:** A comprehensive 2D plotting library for Python, employed for visualizing network topologies, energy consumption trends, and comparative protocol performance metrics.

**scikit-learn** [38]**:** A machine learning library featuring the K-means clustering algorithm, used for baseline protocol implementation and comparative analysis.

**random:** Python's built-in pseudo-random number generator, utilized for node placement initialization and genetic algorithm operations.

**math:** Python's standard mathematical library, providing essential functions (e.g., trigonometric calculations for zone partitioning).

### 4.2.2 Hardware and system specifications

- **Execution platform**: Google Colab
  - **Python version**: Python 3
  - **System RAM**: 12.7 GB
  - **GPU**: T4 GPU (NVIDIA Tesla T4, 15.0 GB GPU RAM)
  - **Disk storage**: 112.6 GB
- **Processor (Local development)**: 11th Gen Intel(R) Core(TM) i5-1135G7 @ 2.40GHz (2.42 GHz)
- **RAM (Local)**: 8 GB
- **Operating System (Local)**: Windows 11 Pro

## 4.3 Simulation setup

### 4.3.1 Network configuration

- **Network area**: 200 m $\times$ 200 m

- **Number of nodes**: 400 sensor nodes
- **Initial node energy**: 0.5 Joules per node
- **Transmission range**: 30 meters
- **Base Station position**: Centrally located at (100, 100)
- **Packet size**: 4000 bits
- **Number of simulation rounds**: 2000

### 4.3.2 Energy model parameters

The simulations use a realistic energy model, with transmission and reception energy consumption calculated based on distance and packet size. Aggregation and compression factors are also considered.

## 4.4 Performance evaluation and results

### 4.4.1 Alive nodes per round

This figure illustrates the number of alive sensor nodes over time for both EAGLE and CLCP protocols. EAGLE protocol maintains node survivability significantly longer:

- **First Node Dies (FND)**: EAGLE at ∼513 rounds vs. CLCP at ∼172 rounds
- **Half Nodes Dead (HND)**: EAGLE at ∼1362 rounds vs. CLCP at ∼753 rounds

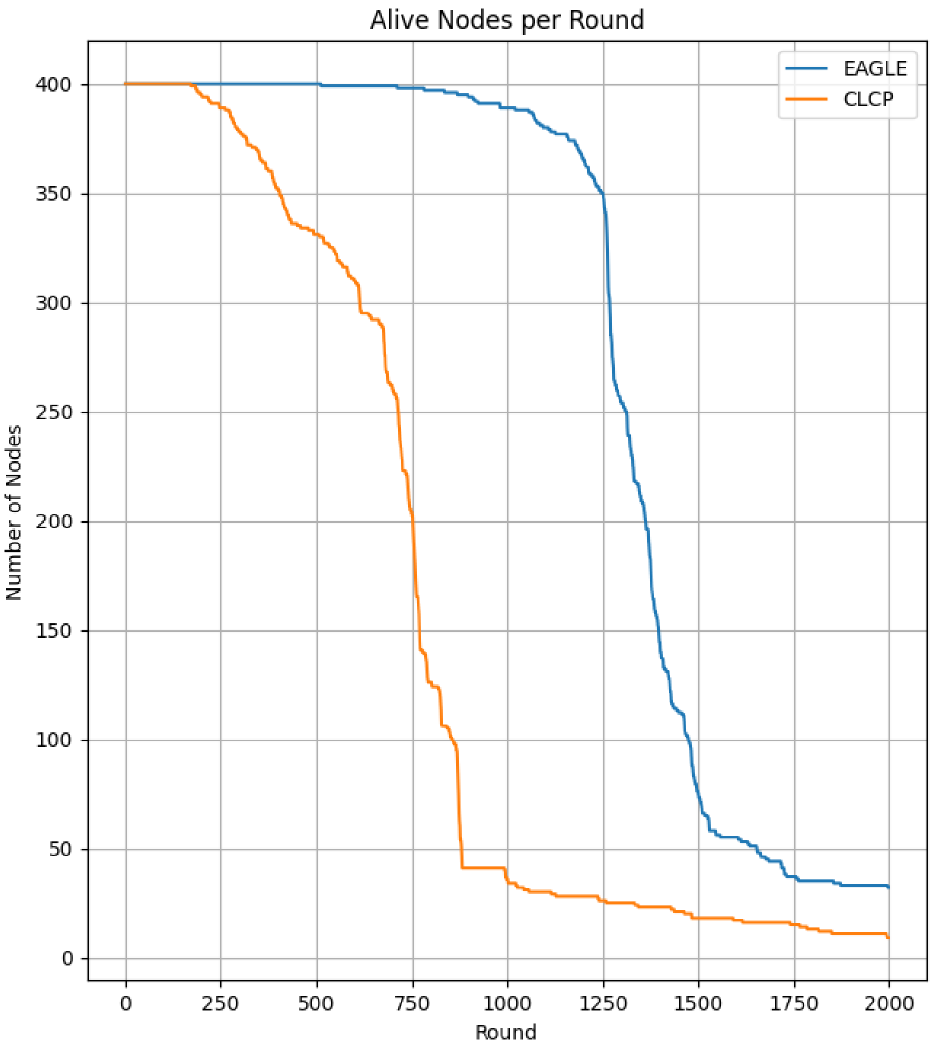


Figure 4.1: Number of alive nodes per round for EAGLE and CLCP protocols

**Discussion:** The slower death rate in EAGLE is primarily due to its intelligent cluster head (CH) rotation via Genetic Algorithms and adaptive routing via Q-learning. The centralized CH placement in CLCP protocol often overburdens specific nodes, leading to quicker energy depletion. In contrast, EAGLE's dynamic energy-aware strategy ensures better load balancing and node longevity.

### 4.4.2 Statistical comparison (FND & HND)

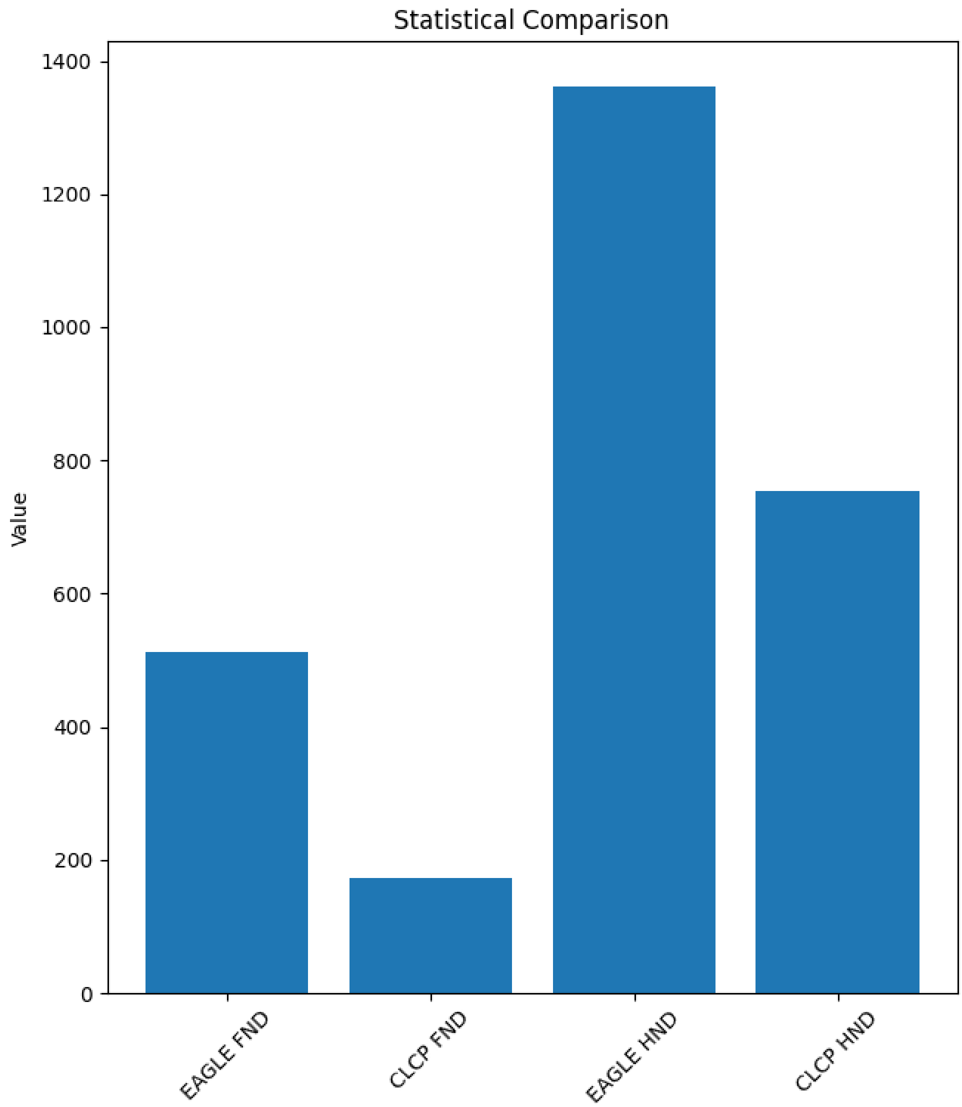


Figure 4.2: Statistical comparison of FND and HND

This bar chart compares First Node Death (FND) and Half Node Death (HND) for both protocols.

**Discussion:** EAGLE significantly outperforms CLCP, indicating better stability and reliability. This metric confirms that EAGLE can operate longer without critical performance loss. GA-based CH rotations prevent node overuse, and Q-learning adapts routing as the topology evolves.

### 4.4.3 Cumulative packet transmission

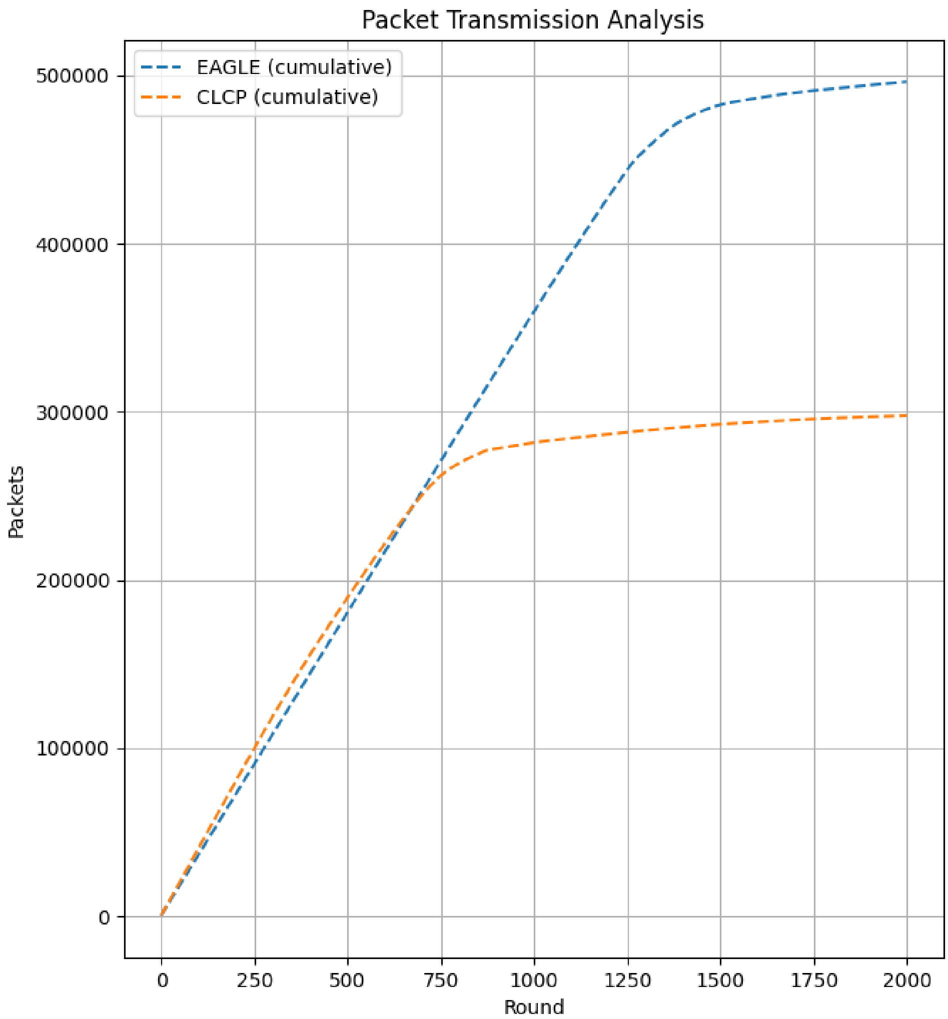


Figure 4.3: Packet transmission analysis

This graph displays the total number of packets successfully transmitted to the base station. EAGLE surpasses 500,000 packets, while CLCP plateaus around 290,000.

**Discussion:** EAGLE's superior packet throughput is a result of prolonged network operation and fewer node failures. Its hierarchical structure with backup CHs and intelligent routing ensures consistent packet delivery even in later rounds. CLCP protocol suffers from early CH deaths and unreliable paths, limiting throughput.

### 4.4.4 Remaining energy per round

This graph shows the total residual energy of all nodes per round.

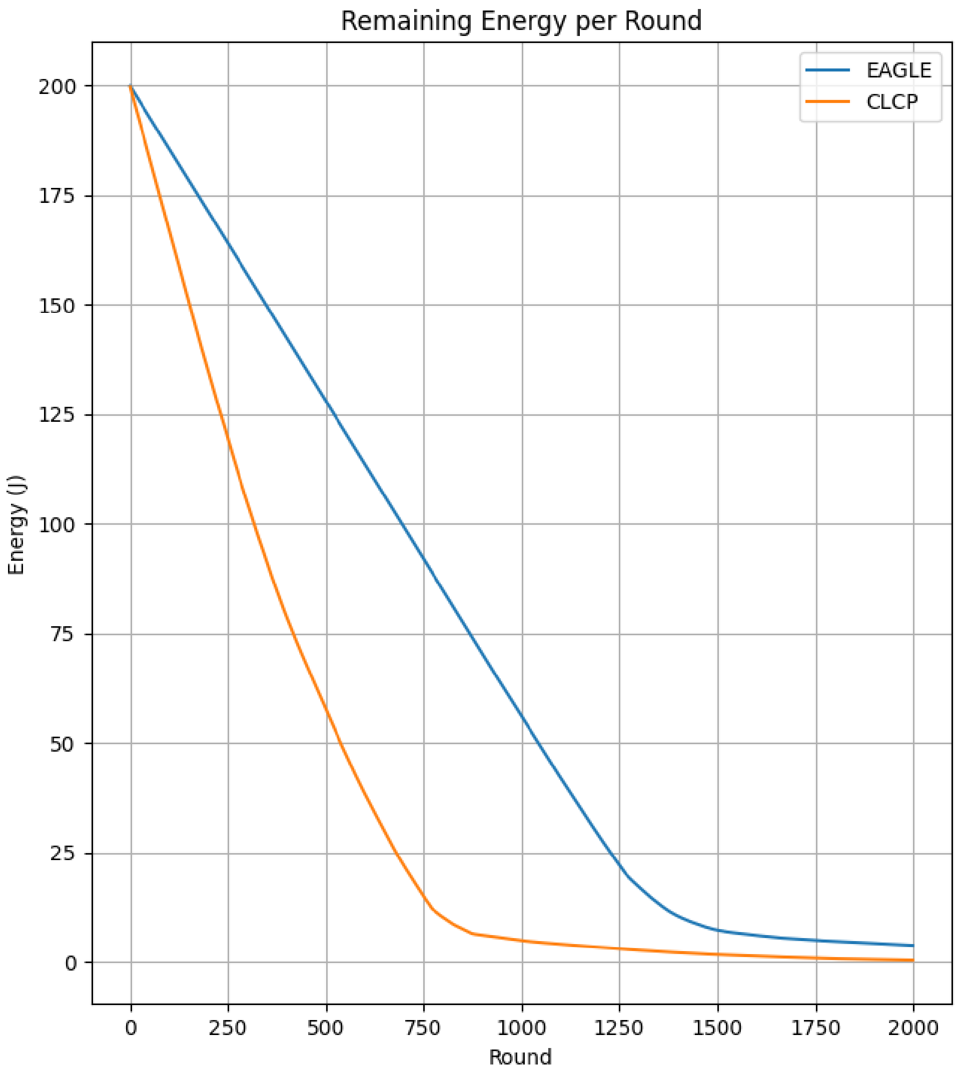


Figure 4.4: Remaining energy per round

**Discussion:** EAGLE protocol conserves energy more effectively due to balanced load distribution and adaptive decision-making. CLCP protocol drains energy faster because it repeatedly selects similar CHs and lacks route adaptivity. The smoother decline in EAGLE's curve reflects controlled energy use and resilience to node failures.

### 4.4.5 Energy efficiency per packet

The graph shows energy consumed per packet transmitted. EAGLE protocol consistently consumes less energy per packet across all simulation rounds.

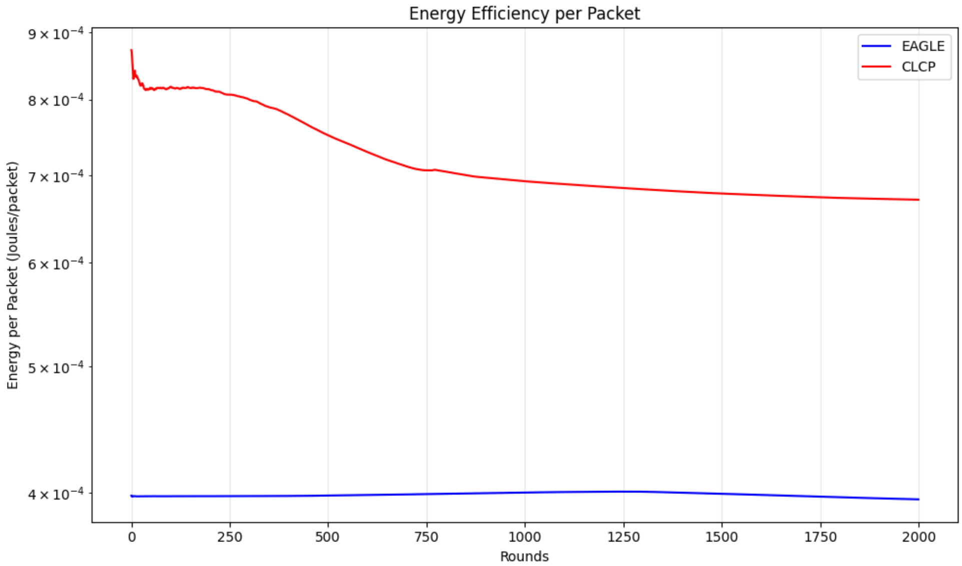


Figure 4.5: Energy consumption per packet comparison

**Discussion:** This efficiency stems from three factors:

- Optimal CH distribution minimizes transmission distance
- Data compression reduces payload size
- Q-learning routes avoid long-distance or congested paths

CLCP protocol cannot adapt to these conditions, resulting in higher per-packet energy costs.

### 4.4.6 Energy Distribution Fairness (CDF)

This cumulative distribution function (CDF) shows the remaining energy distribution across nodes. EAGLE protocol exhibits a more gradual curve compared to the steep CLCP curve.

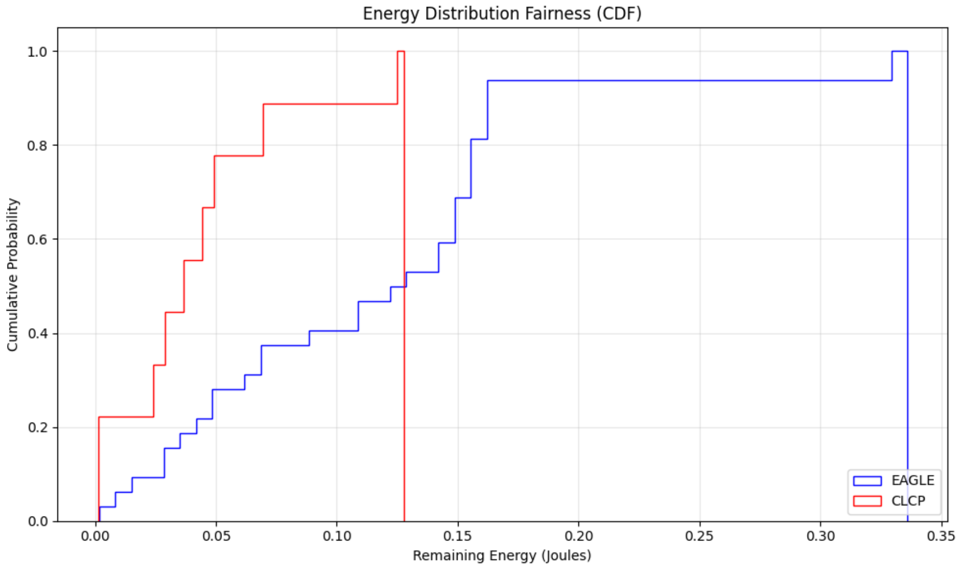


Figure 4.6: Energy distribution fairness comparison

**Discussion:** This implies that in EAGLE protocol, energy consumption is more uniformly distributed among nodes. The fairness results from:

- GA selecting CHs based on residual energy and central location
- Q-learning routing paths avoiding overused links

CLCP protocol lacks these adaptive mechanisms, leading to hotspots and uneven energy drains.

### 4.4.7 Total energy consumed per round

This figure plots cumulative energy consumed over time.

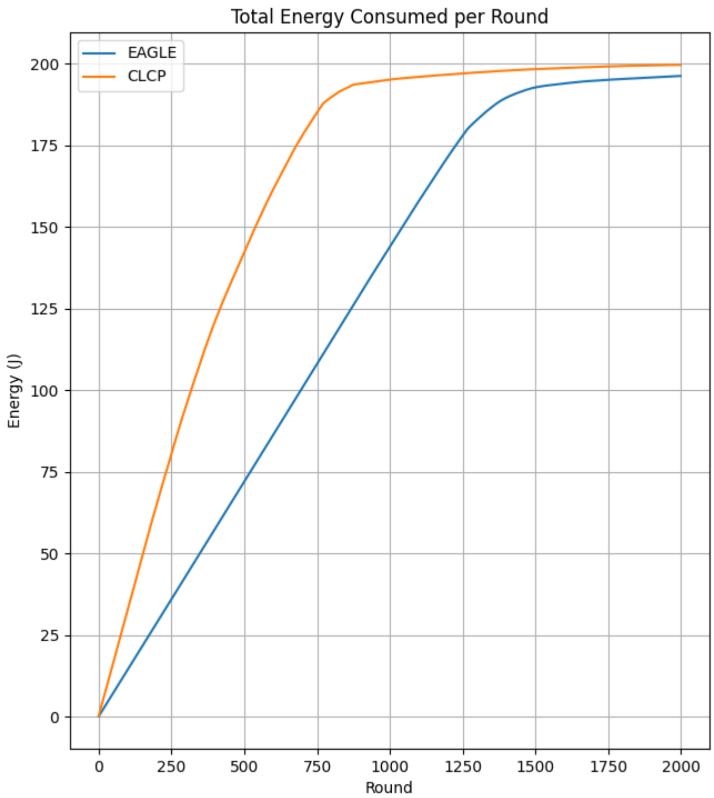


Figure 4.7: Total energy consumed per round

**Discussion:** CLCP protocol reaches energy saturation quickly, consuming nearly all energy before 1000 rounds. EAGLE protocol spreads energy usage more evenly, extending the functional network life. The slower rise in energy consumption corresponds to adaptive task allocation and energy-aware routing decisions.

## 4.5 Summary of results

Table 4.1: Performance comparison summary

| **Metric** | **EAGLE** | **CLCP** | **Advantage** |
|---|---|---|---|
| FND (Round) | ∼514 | ∼150 | EAGLE delayed node death |
| HND (Round) | ∼1376 | ∼706 | EAGLE prolonged network |
| Energy/Packet | Lower | Higher | EAGLE more efficient |
| Packets delivered | >500,000 | ∼290,000 | EAGLE transmits more data |
| Energy fairness | High | Low | EAGLE avoids energy hotspots |

## 4.6 Conclusion

The simulation outputs presented in this chapter are good indicators of the improved performance of the our novel proposal EAGLE protocol compared to CLCP in WSNs. By leveraging the capability of Genetic Algorithms for CH selection in conducting global optimizations, combined with the flexible decision-making capability of Q-learning for routing, EAGLE protocol demonstrates significant improvements in key performance indicators like energy efficiency, network fairness, data throughput, and operational lifespan.

These consistent increases under various conditions highlight the protocol's robustness, scalability, and intelligence. In addition, the findings confirm the suitability of EAGLE's implementation in real-life, energy-constrained, and changing WSN environments, where resourcefulness and flexibility are crucial.

In this context, our proposal EAGLE protocol is an excellent contribution towards further enhancing intelligent and energy-aware communication policies in sensor networks. The following chapter presents general concluding remarks as well as outlines probable future research avenues.

# General conclusion and Perspectives

In our thesis, we have presented a novel hybrid routing architecture for WSNs, named **EAGLE**. This protocol is built upon the synergistic integration of two complementary artificial intelligence techniques: **GA** for the optimized election of cluster heads, and **Q-learning**, a reinforcement learning approach, for adaptive inter-cluster routing optimization.

EAGLE protocol was designed with the goal of addressing the primary limitations inherent to WSNs, including **energy efficiency**, **fault tolerance**, and **scalability**. By introducing a layered and zoned hierarchical structure, intelligent validation of cluster heads, and an adaptive routing strategy driven by learning, our approach demonstrates its ability to extend network lifetime, balance energy consumption across nodes, and maintain high performance in terms of data transmission reliability.

Comparative simulations conducted against protocol CLCP (a traditional K-means-based clustering method) revealed significant improvements across several key performance indicators, including:

- Increased total number of packets successfully transmitted to the base station.
- Extended time before the first node dies (FND) and before half the network nodes fail (HND).
- More equitable distribution of residual energy throughout the network.
- Reduced average energy consumption per packet.

These results validate that the combined use of evolutionary algorithms and reinforcement learning provides a robust and scalable framework for modern WSNs. The effectiveness of Q-learning, particularly through the use of reward functions based on the squared Euclidean distance, has proven especially relevant in promoting energy-efficient and resilient routing. Similarly, the GA-based cluster head selection process ensures balanced choices based on both spatial centrality and residual energy.

**Future work**

This work opens up several promising avenues for future research and improvement:

- **Extension to mobile WSNs:** Adapting EAGLE protocol to networks with mobile sensor nodes would require incorporating trajectory prediction and dynamic cluster maintenance mechanisms.
- **Multi-objective optimization:** By integrating additional criteria such as latency, link reliability, and traffic load, EAGLE protocol could evolve into a multi-objective optimization framework.
- **Security and anomaly detection:** The addition of trust management or supervised learning modules for detecting malicious behavior could further enhance the robustness of the protocol.
- **Hardware deployment:** Implementing the EAGLE protocol on physical hardware platforms (e.g., Arduino, Raspberry Pi, TelosB) would allow validation under real-world conditions.
- **Comparison with other AI techniques:** Exploring variants based on deep learning methods, such as Deep Q-Networks (DQN), could improve learning accuracy and responsiveness to changing environments.

In summary, this thesis contributes to the design of intelligent protocols for WSNs. The proposed approach combines algorithmic rigor, dynamic adaptability, and energy awareness—key attributes for meeting the challenges of next-generation wireless sensor deployments.

# Bibliography


[1] K. Asha and H. S. Anupama, "Factors influencing the use of wireless sensor networks," *International Journal of Advanced Computer Science and Applications (IJACSA)*, vol. 12, no. 3, pp. 708–717, 2021. [Online]. Available: https://thesai.org/Downloads/Volume12No3/Paper_82-Factors_Influencing_the_use_of_Wireless_Sensor_Networks.pdf

[2] M. Macharla, "Basics of wireless sensor networks, topologies and application," *Sensors*, Jun 2021.

[3] A. Abbas, N. Oukas, S. Haboussi *et al.*, "A novel energy-efficient hierarchical routing protocol (eehrp) for iot networks using k-means machine learning," in *2023 International Conference on Computer and Applications (ICCA)*. IEEE, 2023, pp. 1–6.

[4] J. Yick, B. Mukherjee, and D. Ghosal, "Wireless sensor network survey," *Computer Networks*, vol. 52, pp. 2292–2330, 2008.

[5] V. Prakash, S. Pandey, and A. K. Singh, "Basic introduction of wireless sensor network," in *Proceedings of 2nd International Conference on Advanced Computing and Software Engineering (ICACSE)*, 2019.

[6] J. Zheng and A. Jamalipour, *Wireless Sensor Networks: A Networking Perspective*, 2009.

[7] S. Kalantari and S. Taghipour, "A survey on architectures, protocols, applications, and management in wireless sensor networks," *Journal of Advanced Computer Science Technology*, vol. 3, 2014.

[8] U. Nazir, "Analysis of energy efficient hierarchical routing protocols in wireless sensor networks," Ph.D. dissertation, COMSATS Institute of Information Technology, Islamabad, 2012.

[9] I. F. Akyildiz and E. P. Stuntebeck, "Wireless underground sensor networks: Research challenges," *Ad Hoc Networks*, vol. 4, no. 6, pp. 669–686, 2006.

[10] K. Soundarapandian and A. Kumar, "Challenges of wireless sensor networks and issues associated with time synchronization," 2015.

[11] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci, "Wireless sensor networks: a survey," *Computer networks*, vol. 38, no. 4, pp. 393–422, 2002.

[12] D. Xie, W. Wei, Y. Wang, and H. Zhu, "Tradeoff between throughput and energy consumption in multi-rate wireless sensor networks," *Sensors Journal, IEEE*, vol. 13, pp. 3667–3676, 2013.

[13] X. Liu, M. Li, Y. Zhang, X. Zhu, and Y. Guan, "Self-powered wireless sensor node based on rf energy harvesting and management combined design," *Energy Conversion and Management*, vol. 292, p. 117393, 2023.

[14] X. Liu, "Clustering routing algorithms in wireless sensor networks: An overview," *KSII Transactions on Internet and Information Systems*, vol. 6, 2012.

[15] S. Waware, N. Sarwade, and P. Gangurde, "A review of power efficient hierarchical routing protocols in wireless sensor networks," *International Journal of Engineering Research and Applications (IJERA)*, vol. 2, no. 2, pp. 1096–1102, 2012.

[16] K. Malik and M. M. Yaqoob, "An analytical survey on routing protocols for wireless sensor network (wsn)," *International Journal of Computer Applications*, vol. 107, pp. 40–45, 2014.

[17] J. H. Holland, *Adaptation in Natural and Artificial Systems.* University of Michigan Press, 1975.

[18] C. J. C. H. Watkins and P. Dayan, "Q-learning," *Machine Learning*, vol. 8, no. 3-4, pp. 279–292, 1992.

[19] R. Abdesselam and A. Belouatek, “Leach protocol review,” *Journal of Sensor Networks*, 2013.

[20] K. Palan *et al.*, “Firefly algorithm-based leach for wireless sensor networks,” *Procedia Computer Science*, 2017.

[21] B. Sanhaji *et al.*, “Nn-leach: Neural networks for cluster head selection,” *International Journal of Wireless Communications*, 2019.

[22] O. Younis and S. Fahmy, “Heed: A hybrid energy-efficient distributed clustering approach for ad hoc sensor networks,” *IEEE Transactions on Mobile Computing*, 2004.

[23] S. Yahoui, “Energy-efficient hierarchical routing protocol for wireless sensor networks,” Master’s thesis, University of Bouira, 2024.

[24] Y. Elghazi *et al.*, “Artificial intelligence-based cluster routing for wsns using soms,” *Sensors and Systems*, 2023.

[25] F. Hussain *et al.*, “Hybrid ga-anfis clustering in wireless sensor networks,” *Applied Soft Computing*, 2007.

[26] M. Riham and H. Yagoub, “Pso for hierarchical clustering in wsns,” *Wireless Sensor Systems Journal*, 2014.

[27] B. Sahoo *et al.*, “Bio-inspired secure clustering for wsns,” *Journal of Network Security*, 2015.

[28] M. Dahane *et al.*, “Bs-wca: Balanced and secure clustering for wsns,” *Ad Hoc Networks*, 2015.

[29] S. Benahmed *et al.*, “Sd-ca: Distributed anomaly detection in wsns,” *International Journal of Distributed Sensor Networks*, 2012.

[30] M. Chatterjee *et al.*, “Wca for mobile ad hoc networks,” *Cluster Computing*, 2002.

[31] Y. Choi and S. Woo, “Dwcar: Distributed clustering in manets,” *Wireless Communications and Mobile Computing*, 2006.

[32] W. B. Heinzelman, A. P. Chandrakasan, and H. Balakrishnan, "An application-specific protocol architecture for wireless microsensor networks," *IEEE Transactions on wireless communications*, vol. 1, no. 4, pp. 660–670, 2002.

[33] M. Gholipour, Z. Asadi, M. Mahmoud, and M. Shamsi, "Arzsep: An angle-based regional zoning stable election protocol for energy-efficient clustering in wireless sensor networks," *Wireless Personal Communications*, 2024. [Online]. Available: https://link.springer.com/article/10.1007/s11277-024-11651-w

[34] A. Gautam, D. Mohapatra, and S. K. Swain, "Zsep-e: Zonal stable election protocol with enhanced approach for wireless sensor networks," *International Journal of Applied Engineering Research*, vol. 10, no. 9, pp. 23 057–23 062, 2020. [Online]. Available: https://www.ripublication.com/ijaer10/ijaerv10n9_194.pdf

[35] R. Ghasemi and A. Khalaj, "Q-learning-based energy-efficient routing in wireless sensor networks," *Ad Hoc Networks*, vol. 85, pp. 113–123, 2019.

[36] C. R. Harris, K. J. Millman, S. J. van der Walt, R. Gommers, P. Virtanen, D. Cournapeau, E. Wieser, J. Taylor, S. Berg, N. J. Smith, R. Kern, M. Picus, S. Hoyer, M. H. van Kerkwijk, M. Brett, A. Haldane, J. F. del Río, M. Wiebe, P. Peterson, P. Gérard-Marchant, K. Sheppard, T. Reddy, W. Weckesser, H. Abbasi, C. Gohlke, and T. E. Oliphant, "Array programming with NumPy," *Nature*, vol. 585, pp. 357–362, 2020.

[37] J. D. Hunter, "Matplotlib: A 2d graphics environment," *Computing in Science & Engineering*, vol. 9, no. 3, pp. 90–95, 2007.

[38] F. Pedregosa, G. Varoquaux, A. Gramfort, V. Michel, B. Thirion, O. Grisel, M. Blondel, P. Prettenhofer, R. Weiss, V. Dubourg, J. Vanderplas, A. Passos, D. Cournapeau, M. Brucher, M. Perrot, and Duchesnay, "Scikit-learn: Machine learning in Python," *Journal of Machine Learning Research*, vol. 12, pp. 2825–2830, 2011. [Online]. Available: http://jmlr.org/papers/v12/pedregosa11a.html