**HEACT: A Hybrid Energy-Aware Cluster-Tree Routing Protocol for Enhanced Longevity in Wireless Sensor Networks**

**Project submitted in partial fulfillment of the requirement for the degree of Bachelor of Technology in Computer Science and Engineering / Information Technology**

**by**

**Tanmay Kumar Naik, Regd.No.:2002081019**

**Subhransu Sekhar Panda, Regd.No.:2102060006**

**under the Supervision of**

**Dr. Satyabrata Das**

****

Department of Computer science and Engineering

**Veer Surendra Sai University of Technology**

**Burla, Odisha-768018**

[2024-25]

**Department of Computer Science and Engineering**

**Veer Surendra Sai University of Technology, Burla, Odisha**

Date:

**Certificate of Examination**

Registration Number:

Name:

Title of the Project: **HEACT: A Hybrid Energy-Aware Cluster-Tree Routing Protocol for Enhanced Longevity in Wireless Sensor Networks**

This is to certify that the project report entitled “**HEACT: A Hybrid Energy-Aware Cluster-Tree Routing Protocol for Enhanced Longevity in Wireless Sensor Networks**” submitted by **Tanmay Kumar Naik** Regd.no. **2002081019, Subhransu Sekhar Panda Regd. No.: 2102060006** of Computer Science and Engineering under the Department of Computer Science and Engineering, Veer Surendra Sai University of Technology, Odisha has been examined by us. We are satisfied with the volume, quality and correctness of the work.

Internal Examiners External Examiners

HOP, UG HOD, Dept. Of CSE

Dept. Of CSE VSSUT, BURLA



**Department of Computer Science and Engineering**

**Veer Surendra Sai University of Technology, Burla, Odisha**

Date:

# Supervisors’s Certificate

This is to certify that the Project work entitled “**HEACT: A Hybrid Energy-Aware Cluster-Tree Routing Protocol for Enhanced Longevity in Wireless Sensor Networks**” is being submitted by **Tanmay Kumar Naik (**Regd.no. **2002081019), Subhransu Sekhar Panda (**Regd.No. **2102060006)**, to the Department of Computer Science and Engineering, Veer Surendra Sai University of Technology, Burla, in partial fulfillment of the requirement for the degree of Bachelor of Technology in Computer Science and Engineering/Information Technology during the academic year 2024-2025. It is an original report carried out by him/her/them under my supervision.

Dr. Satyabrata Das

(Supervisor**)**

# Declaration

I/We hereby declare that the project titled “**HEACT: A Hybrid Energy-Aware Cluster-Tree Routing Protocol for Enhanced Longevity in Wireless Sensor Networks”** submitted to Veer Surendra Sai University of Technology for the award of the degree of Bachelor of Technology in Computer Science and Engineering / Information Technology is a result of original work carried out in this report. I/We also declare that the work has not been submitted, in whole or in part, to any other university as an exercise for a degree or any other qualification.

Date: Sohel

VSSUT, BURLA Regd.No.:2002081019

# Acknowledgment

I/We would like to thank my supervisor, **Dr.Kshiramani Naik** for giving us the opportunity to undertake this project. I am extremely grateful for her insightful guidance, timely advice, continuous support, and motivation throughout my/ our project work. I/We especially thank his/her for her/his kindness and excellent contributions.

I/We would like to thank the HOD, Department of Computer Science and Engineering for his support to carry out my/our project work. I would like to thank all the faculty and staff members of the Department of Computer Science and Engineering at Veer Surendra Sai University of Technology, Burla,for their help and support.

I am deeply thankful to the Vice-Chancellor for his moral support and for providing us with all the necessary facilities during our candidature.

I would like to thank our friends and family members for their unconditional love and support.

Name:

Registration no.:

# Abstract

Addressing premature node failure and extending network duration in Wireless Sensor Networks (WSNs) requires sophisticated energy management through routing protocols. This work **proposes and evaluates a novel protocol, the Hybrid Energy-Aware Cluster-Tree (HEACT)**, designed to enhance network longevity and stability. HEACT integrates periodic clustering (with CH selection influenced by residual energy and distance from the base station, plus cluster size caps) with a dual-mode Cluster Head (CH) transmission strategy. This strategy employs an initial direct CH-to-Base Station (BS) transmission phase to mitigate early energy depletion, followed by a transition to multi-hop routing along an inter-CH tree constructed using an energy-penalized cost metric to favor robust relay nodes. **To assess its effectiveness, HEACT's performance is compared via simulation against two established protocols: the standard LEACH protocol and the dynamic Energy-Efficient Tree-Based (EETB) protocol.** Simulations are conducted under identical conditions, analyzing critical performance indicators: network stability period (FDN), network lifetime (NL), total energy consumption dynamics, and packet delivery ratio. The results aim to quantify the performance improvements offered by the proposed HEACT architecture relative to traditional clustering and dynamic tree approaches in managing WSN energy resources.

**Keywords:** Wireless Sensor Networks; Energy Efficiency; Network Lifetime; Clustering; Tree-Based Routing; Network Stability Period

# Contents

|  |  |
| --- | --- |
| Certificate of Examination………………………………………………………………… | I |
| Supervisor’s Certificate…………………………………………………………………… | II |
| Declaration………………………………………………………………………………… | III |
| Acknowledgement…………………………………………………………………………. | IV |
| Abstract…………………………………………………………………………………….. | V |
| List of Figures……………………………………………………………………………… | VII |
| List of Tables………………………………………………………………………………. | VIII |
| List of Abbreviations………………………………………………………………………. | IX |
| 1. Introduction…………………………………………………………………………… | 1 |
| 1. Literature Review………………………………………………………… | 2 |
| 1. Proposed Method/ Model…………………………………………………………… | 3 |
| 3.1…. | 3 |
| 3.2…. | 4 |
| 3.2.1……………. | 4 |
| 1. Results and Discussion……………………………………………………. | 12 |
| 4.1…………………………………………………………………………. | 13 |
| 4.2…………………………………………………………………………. | 14 |
| 4.3………………………………………………………………………….. | 15 |
| 1. Conclusion and Future Scopes……………………………………………. | 18 |
| References……………………………………………………………………. | 19 |
|  |  |

**List of Figures**

**Figure1:**GeneralCNNstructure 2

**Figure2:**CodeFlow 14

**Figure3:**Gradio Interface showing to upload image 18

**Figure4:**Arealimagewasuploaded anditspredictionwas shown 19

**Figure5:**Afakeimagewas..uploadedanditspredictionwasshown 19

**List of Tables**

**List of Abbreviations**

**CNN:**ConvolutionalNeuralNetwork

**AI:**ArtificialIntelligence

**ML:**Machine Learning

**MTCNN:**Multi-TaskCascadedConvolutionalNeuralNetwork

**IT:**InformationTechnology

# Introduction

**1.1 The Rise and Significance of Wireless Sensor Networks**

The proliferation of low-cost, low-power micro-electro-mechanical systems (MEMS) technology, coupled with advancements in wireless communication, has propelled Wireless Sensor Networks (WSNs) to the forefront of modern monitoring and control systems [Reference on WSN Importance, e.g., Akyildiz foundational paper]. WSNs typically consist of a large number of small, autonomous sensor nodes deployed across a physical area to observe environmental conditions (e.g., temperature, humidity, light, sound, pollutants), track objects, monitor structural health, or manage industrial processes [Reference on Applications]. These nodes possess sensing, processing, and wireless communication capabilities, allowing them to collaboratively collect data and transmit it, often via multi-hop routes, to a central coordinating entity known as the Base Station (BS) or sink node. The BS typically acts as a gateway, connecting the WSN to external networks (like the internet) for data analysis, storage, and user interaction.

The applications of WSNs are vast and transformative. In environmental monitoring, they enable fine-grained data collection for climate change studies, precision agriculture, forest fire detection, and early warning systems for natural disasters. Industrial applications include process automation, equipment health monitoring, supply chain tracking, and ensuring safety in hazardous environments. In healthcare, WSNs facilitate remote patient monitoring, assisted living for the elderly, and tracking of medical assets. Furthermore, they are key enablers for Smart Cities, contributing to intelligent transportation systems, smart grids, structural health monitoring of infrastructure, and optimized resource management. The ability to deploy these networks in remote, hazardous, or inaccessible locations without requiring pre-existing infrastructure makes them uniquely valuable.

**1.2 The Fundamental Challenge: Energy Constraint**

Despite their versatility, the widespread adoption and long-term viability of WSNs are fundamentally constrained by the limited energy resources of individual sensor nodes [Reference on Energy Constraint]. Nodes are typically powered by small batteries, which are often difficult or impossible to replace or recharge after deployment, especially in large-scale or remote applications. Consequently, energy efficiency becomes the most critical design consideration, influencing every aspect of the network's operation, from hardware design and sensing schedules to communication protocols and data processing techniques.

The communication subsystem is widely recognized as the dominant source of energy consumption in a typical sensor node [Reference on Communication Energy Cost]. Transmitting or receiving data wirelessly consumes significantly more energy than sensing or processing tasks. Moreover, the energy required for transmission increases dramatically with distance (often following a power law, d^2 to d^4, depending on the environment [1]). This inherent characteristic makes long-distance, single-hop communication highly inefficient and unsustainable for battery-powered nodes.

**1.3 The Problem of Unbalanced Energy Consumption and Network Longevity**

Compounding the limited energy budget is the challenge of non-uniform energy consumption across the network. In many WSN architectures, certain nodes inevitably bear a heavier communication burden than others. Nodes located closer to the BS often act as relays for data originating from more distant nodes, leading to faster energy depletion in this "hotspot" region near the sink – the well-known "energy hole" problem [Reference on Energy Hole]. Similarly, in clustering protocols, the Cluster Heads (CHs) consume significantly more energy than regular member nodes due to receiving data from members, performing aggregation, and transmitting the aggregated data over potentially longer distances to the BS or other relays.

This unbalanced energy consumption leads directly to **premature node failure**. When critical relay nodes or overloaded CHs deplete their energy, connectivity can be lost, partitioning the network and preventing data from large sections from reaching the BS. This significantly shortens the *effective* network lifetime, often defined by metrics such as:

* **Network Stability Period (FDN - First Dead Node):** The time until the very first node in the network dies. A short stability period indicates poor load balancing and early network degradation.
* **Network Lifetime (NL):** Commonly defined as the time until the last node dies (LND), or sometimes until a certain percentage of nodes die, or until network partitioning occurs. Maximizing NL is crucial for long-term application viability.

Therefore, designing routing protocols that not only minimize overall energy consumption but also *balance* the energy load across nodes is essential for maximizing both the stability period and the overall lifetime of the WSN.

**1.4 Existing Routing Paradigms and Their Limitations**

Numerous routing strategies have been proposed to address energy efficiency in WSNs. These can be broadly classified:

* **Flat Routing:** Protocols like Directed Diffusion or SPIN involve data negotiation and interest propagation, but can incur significant overhead. Direct transmission from each node to the BS is the simplest but least scalable and most energy-intensive approach for all but the smallest networks.
* **Hierarchical (Clustering) Routing:** Protocols like LEACH [1] and its successors [2, 7-9, 13, 14, 21] establish a two-tier hierarchy. Nodes form clusters, electing CHs responsible for managing intra-cluster communication and relaying aggregated data towards the BS. Clustering significantly reduces energy consumption for member nodes by limiting their transmission range. However, standard LEACH suffers from random CH selection potentially leading to poor cluster distribution and energy imbalances. Direct CH-to-BS transmission remains a bottleneck, rapidly draining CHs far from the sink. While many variants improve CH selection using energy, distance, or other metrics, effectively balancing the inter-cluster transmission load remains a key challenge.
* **Tree-Based Routing:** These protocols construct explicit multi-hop paths forming a tree structure rooted at the BS [Ref Original EETB Paper]. Data is relayed hop-by-hop towards the root. This avoids the very long single hops of direct transmission or some LEACH variants. Protocols like EETB attempt to optimize this by calculating an ideal number of branches and using dynamic updates based on energy thresholds to adapt the tree structure and prevent overburdening low-energy nodes. However, nodes near the root can still become bottlenecks, and the overhead of maintaining an optimal global tree structure can be considerable, especially if updates are frequent.
* **Chain-Based Routing:** PEGASIS [20] forms a chain among nodes, reducing transmission overhead but introducing significant latency and a single point of failure vulnerability [23, 25].
* **Hybrid Routing:** These approaches attempt to combine the benefits of clustering and multi-hop routing (trees/chains) [23, 26, 27]. Often, they involve clustering locally and then establishing routes between CHs. While promising, effectively managing the interaction between the two levels and minimizing overall control overhead is complex.

**1.5 Motivation and Proposed Solution: HEACT**

While existing protocols have made significant strides, achieving both a long stability period (high FDN) *and* a long overall network lifetime (high NL) simultaneously remains challenging. Protocols excelling at one often compromise on the other. For instance, aggressive load balancing might extend NL but could initially burden specific nodes, leading to a low FDN. Conversely, simple schemes might have decent initial stability but fail to efficiently route data over the long term, shortening NL.

Furthermore, many hybrid protocols establish fixed inter-CH structures or rely on standard LEACH-like mechanisms for CH roles, potentially inheriting their limitations. There is a need for a hybrid protocol that explicitly addresses early node failure while incorporating robust, energy-aware mechanisms for sustained operation.

This paper proposes the **Hybrid Energy-Aware Cluster-Tree (HEACT)** protocol to bridge this gap. HEACT is founded on the principle of adaptive communication strategies tailored to different phases of the network's life and energy distribution. Its key innovations include:

1. **Combined Energy/Distance CH Selection:** Uses a probabilistic approach where candidacy is weighted by both the node's residual energy share and its relative distance from the BS, aiming for a balanced set of spatially distributed and energetically capable CHs.
2. **Cluster Size Limitation:** Explicitly limits the number of members per cluster (HEACT\_MAX\_CLUSTER\_SIZE) to prevent CH overload, directly promoting load balancing at the cluster level.
3. **Initial Direct Transmission Phase:** A critical feature where, for the first few reconfiguration cycles (HEACT\_INITIAL\_DIRECT\_INTERVALS), CHs transmit directly to the BS. This deliberately avoids the high initial cost of multi-hop relaying, allowing the network to stabilize and node energies to differentiate before imposing complex routing tasks. This is specifically designed to improve the FDN.
4. **Energy-Penalized Inter-CH Tree:** After the initial phase, HEACT transitions to multi-hop routing. An inter-CH tree is constructed where the cost metric for selecting a parent CH heavily penalizes potential parents with low residual energy (using HEACT\_TREE\_ENERGY\_PENALTY\_EXPONENT). This ensures that the multi-hop backbone relies on the most energetically robust CHs available at that time, promoting longevity of the inter-cluster routing structure.
5. **Periodic Reconfiguration:** The clustering and tree structures are periodically rebuilt (HEACT\_RECLUSTER\_INTERVAL) to adapt to node deaths and changing energy landscapes.

By combining these mechanisms, HEACT aims to achieve superior performance by protecting the network during its initial, vulnerable phase and then transitioning to an efficient, energy-aware multi-hop structure for sustained data delivery.

**1.6 Contributions and Paper Structure**

The main contributions of this paper are:

* Proposal of the HEACT protocol, a novel hybrid routing algorithm combining adaptive clustering, cluster size limits, an initial direct transmission phase, and energy-penalized inter-CH tree routing.
* Detailed description of the HEACT algorithm components, including refined CH selection and energy-weighted tree construction.
* Comparative performance evaluation of HEACT against standard LEACH and the dynamic tree-based EETB protocol through simulation.
* Analysis of key performance metrics, including FDN, NL, throughput, and energy consumption characteristics, to demonstrate the effectiveness of the proposed approach.

The remainder of the paper is organized as follows: Section II details the network and energy models. Section III presents the algorithms comprising the HEACT protocol. Section IV describes the simulation setup and parameters. Section V analyzes and discusses the comparative simulation results. Section VI concludes the paper and outlines directions for future work.

# 

# Literature Review

# The imperative of energy conservation in Wireless Sensor Networks (WSNs) has spurred extensive research into efficient routing protocol design. Over the years, numerous strategies have emerged, each with distinct architectural philosophies, operational mechanisms, and inherent trade-offs regarding energy consumption, network lifetime, stability, latency, and scalability. These protocols can be broadly categorized into flat, hierarchical (clustering), tree-based, and hybrid approaches. Given our focus on hierarchical and structured routing, we will primarily review clustering, tree-based, and hybrid strategies, positioning the proposed HEACT protocol within this landscape.

# 2.1 Hierarchical (Clustering) Routing Protocols

# Clustering protocols organize the WSN into a hierarchical structure, typically involving groups of sensor nodes (clusters) managed by designated Cluster Heads (CHs). This approach offers several intrinsic advantages:

# Scalability: By localizing communication within clusters and limiting direct transmissions to the Base Station (BS), clustering can handle larger numbers of nodes more effectively than flat routing schemes.

# Energy Efficiency (for Members): Regular sensor nodes (members) usually only need to transmit data over short distances to their respective CHs, significantly saving energy compared to long-haul transmissions to the BS.

# Data Aggregation: CHs act as local aggregation points, fusing redundant data received from members before forwarding it, thereby reducing the overall traffic load in the network.

# Load Distribution: Rotating the energy-intensive CH role among different nodes over time helps distribute the burden, preventing any single node from depleting its energy prematurely solely due to CH duties.

# LEACH (Low-Energy Adaptive Clustering Hierarchy) [1]: As the foundational clustering protocol, LEACH introduced the concept of randomized, self-organizing cluster formation and CH rotation. In each round, nodes probabilistically elect themselves as CHs based on a target percentage (P) and their past history of being a CH. Non-CH nodes join the nearest CH based on received signal strength. CHs create TDMA schedules for their members, receive member data, aggregate it, and transmit the aggregated packet directly to the BS. While ground breaking,

# LEACH suffers from several limitations: \* Random CH selection offers no guarantee of optimal CH number or distribution. \* Direct CH-to-BS transmission is energy-intensive for distant CHs. \* It doesn't explicitly consider residual node energy during CH selection.

# Enhancements to LEACH and Clustering: Recognizing LEACH's shortcomings, a vast body of work has focused on refining clustering protocols:

# Optimizing Cluster Head Selection: This is arguably the most researched area. The goal is to select CHs that are energetically capable and strategically positioned.

# Energy-Aware Selection: Protocols like LEACH-C (Centralized LEACH) use the BS to select CHs based on global energy information, ensuring better candidates but introducing centralization overhead. Distributed approaches like HEED (Hybrid Energy-Efficient Distributed Clustering) [Reference HEED paper, e.g., Younis and Fahmy] use iterative refinement where nodes consider both their residual energy (primary parameter) and an intra-cluster communication cost metric (e.g., node degree, average distance to neighbors – secondary parameter) to become CHs. Nodes compete locally, with higher-energy nodes having a higher chance of becoming final CHs. SEP (Stable Election Protocol) [Reference SEP paper, e.g., Smaragdakis et al.] explicitly handles node heterogeneity by giving higher-energy (advanced) nodes a greater probability of becoming CHs than lower-energy (normal) nodes. Many protocols incorporate residual energy directly into the LEACH probability threshold calculation [2, 7].

# Location-Aware Selection: Some protocols factor in the CH candidate's distance to the BS, aiming to reduce the energy cost of inter-cluster communication [7]. Others consider node density to ensure better cluster coverage.

# Fuzzy Logic & Metaheuristics: To handle the multi-objective nature of CH selection (balancing energy, distance, density, etc.), researchers have employed fuzzy logic systems [9] that can map multiple input parameters to a CH candidacy chance value. Metaheuristic optimization algorithms like Genetic Algorithms (GA) [13], Ant Colony Optimization (ACO) [6, 7], Particle Swarm Optimization (PSO), Firefly Algorithm [10], Tuna Swarm Optimization (TSO) [18], and others [12, 14] have been used to search the solution space for near-optimal CH sets based on complex cost functions. These can achieve excellent results but often incur higher computational complexity and convergence time.

# Unequal Clustering: To combat the energy hole problem, protocols like UCR (Unequal Clustering Routing) [8] deliberately create smaller clusters closer to the BS and larger clusters farther away. The rationale is that CHs near the BS handle heavier relay traffic and should thus manage fewer members to conserve energy.

# Optimizing Data Transmission: Improvements also target how data moves within and between clusters.

# Intra-Cluster Multi-hop: Instead of direct member-to-CH transmission, some protocols allow members farther from the CH to relay data through closer members, saving energy for peripheral nodes but increasing latency and complexity within the cluster.

# Inter-Cluster Multi-hop: The most significant departure from basic LEACH involves CHs *not* transmitting directly to the BS. Instead, CHs relay data through other CHs towards the BS [18, 21]. This requires establishing inter-CH routes, often using shortest paths or energy-aware metrics. EESRA [21] uses a three-layer structure involving nodes, CHs, and potentially super-CHs, incorporating multi-hop transmissions at different levels. Algorithms might use Dijkstra's [15, 16] or Minimum Spanning Trees (MST) [17, 19] to establish inter-CH paths or routes within clusters.

# Limitations of Clustering Approaches: Despite these advancements, many clustering protocols still face inherent trade-offs. Multi-hop inter-CH routing can significantly save energy but adds complexity and potential delay. Optimal CH selection remains challenging, especially in dynamic distributed environments. Protocols relying solely on clustering might still struggle to perfectly balance load, especially if the BS is far outside the network deployment area.

# 2.2. Tree-Based and Chain-Based Protocols

# These protocols focus on constructing explicit, often fixed or dynamically updated, data forwarding structures.

# Tree-Based Protocols: Data naturally flows from numerous sources (leaves) towards a single sink (root/BS) along defined paths.

# Advantages: Can provide clear routing paths and potentially lower overhead compared to constant cluster reconfiguration if the tree is stable. Multi-hopping inherently avoids extremely long single transmissions.

# Challenges: Efficiently constructing an energy-balanced tree is difficult. Nodes closer to the root (BS) naturally handle more aggregated traffic and are prone to rapid energy depletion (energy hole). Tree maintenance in response to node failures or energy depletion requires careful design to avoid excessive control overhead.

# EETB (Energy-Efficient Tree-Based) [Ref Original Paper Here]: This protocol represents a sophisticated tree-based approach. It attempts to mitigate the energy hole problem by:

# Optimal Branch Calculation (h\_opt): Uses an energy model considering transmission costs and network dimensions to calculate an optimal number of main branches connecting directly to the BS, aiming to distribute the initial load.

# Distance-Based Formation: Builds the tree structure from near-to-far based on node distance to the BS and existing tree nodes, minimizing path costs initially.

# Energy Threshold (Eth): Prevents nodes with energy below a calculated threshold from becoming relay nodes during tree updates, forcing them into leaf roles and preserving their remaining energy.

# Dynamic Update Interval (Rdyit): Updates the tree not every round, but at intervals determined by the predicted minimum lifetime among nodes and the variance in energy consumption. This reduces the control overhead associated with frequent tree rebuilding while still adapting to energy changes.

# Other Tree Variants: Two-level tree clustering (e.g., EE-TLT [26]) combines clustering with tree structures, building minimum spanning trees within clusters (rooted at sub-CHs) and between CHs/Relay-CHs/BS. This adds hierarchy but also complexity.

# Chain-Based Protocols: These represent a specific linear topology.

# PEGASIS (Power-Efficient Gathering in Sensor Information Systems) [20]: Nodes form a long chain (e.g., using a greedy algorithm). Each node only communicates with its immediate neighbors on the chain, receiving data from one side, fusing it with its own, and passing it to the other side. A randomly selected leader node transmits the final aggregated data to the BS each round.

# Advantages: Minimizes transmission distances for most nodes, theoretically achieving significant energy savings compared to LEACH's direct transmissions.

# Disadvantages: Introduces significant data gathering latency as data must traverse the entire chain. The network is extremely vulnerable – failure of a single node breaks the chain. The leader node still performs a potentially long-distance transmission to the BS. Enhancements like E-PEGASIS [24] consider residual energy and distance when selecting the leader or modifying chain construction [23, 25].

# 2.3. Hybrid Protocols

# Hybrid approaches aim to combine the benefits of different paradigms. Examples include:

# Cluster-Chain: Forming local clusters and then linking the CHs into a PEGASIS-like chain [23].

# Cluster-Tree: Forming clusters and then building a tree structure between CHs [26, 27]. EE-TLT [26] uses a two-level tree based on sectors and levels. CER-CH [27] uses a top-down tree for routing among heterogeneous CHs.

# Geographic/Zone-Based: Dividing the network into zones and potentially using different strategies (e.g., direct transmission near BS, clustering farther away).

# 2.4. Positioning HEACT and Addressing Gaps

# The proposed HEACT protocol is a hybrid cluster-tree approach designed to overcome specific limitations observed in prior work:

# Addressing Low FDN: Unlike many hierarchical protocols (including initial HEACT versions) that can suffer from immediate overload of early CHs/relays, HEACT incorporates the Initial Direct Transmission Phase. This crucial step shields the network during its most vulnerable initial period, allowing energy levels to stabilize before demanding multi-hop relaying from CHs, thereby directly targeting the improvement of the First Dead Node (FDN) metric.

# Improved Load Balancing: HEACT combines multiple techniques:

# Energy-aware CH selection (refined probabilistic approach).

# Distance-aware CH selection (using D\_factor).

# Explicit Cluster Size Limits (C\_max\_size) prevent individual CHs from being swamped by too many members, a common issue not always directly addressed in LEACH variants or tree protocols.

# Robust Inter-CH Routing: By transitioning to an inter-CH tree *after* stabilization and using an Energy-Penalized Cost Metric (P\_exp) for tree construction, HEACT aims to build a more resilient multi-hop backbone that relies on stronger CHs, enhancing the longevity of the inter-cluster communication phase.

# Adaptability: While reconfiguration is periodic (I\_recluster), the dual-mode transmission (direct vs. tree) provides adaptation based on the network's operational phase (initial vs. later).

# 2.5. Comparison Table

# Table I summarizes the key characteristics and differences between HEACT, EETB, and LEACH based on the discussed features.

|  |  |  |  |
| --- | --- | --- | --- |
| Feature | LEACH (Standard) | EETB (Dynamic Tree) | HEACT (Proposed Hybrid) |
| Basic Architecture | Hierarchical (Clustering) | Flat (Multi-Branch Tree) | Hierarchical (Cluster + Inter-CH Tree) |
| Primary Routing Unit | Cluster Head (CH) | All Nodes (Relay/Leaf) | Cluster Head (CH) |
| Intra-Network Routing | Member → CH (Single Hop) | Child Node → Parent Node (Multi-Hop) | Member → CH (Single Hop) |
| Sink Communication | CH → BS (Single Hop) | Branch Root → BS (Single Hop) | Dual Mode: Initial CH→BS (Single Hop), Later CH→Parent CH/BS (Multi-Hop Tree) |
| Structure Formation | Probabilistic CH Election (Random) | Distance-Based Tree Construction | Energy/Dist-Aware CH Selection, Energy-Weighted Inter-CH Tree |
| Structure Update | Per Round (CH Rotation) | Dynamic Intervals (Rdyit) | Periodic Intervals (I\_recluster) |
| Energy Consideration | None (in standard LEACH selection) | Eth Threshold for Relays, h\_opt Model | E\_cand, E\_relay Thresholds, Energy-Weighted Selection & Tree Cost |
| Load Balancing Mech. | CH Rotation | Optimal Branches (h\_opt), Eth Updates | CH Rotation, Size Limit (C\_max\_size), Energy/Dist Selection, Energy Tree Cost |
| FDN Improvement Strategy | Implicit (Rotation) | Eth Threshold, Dynamic Updates | Initial Direct Phase, Size Limit, Energy Awareness |

# Table I: Comparison of Routing Protocol Characteristics

# This comparative overview highlights how HEACT attempts to integrate mechanisms addressing specific weaknesses of pure clustering or pure tree-based approaches, particularly focusing on enhancing the critical initial network stability period while maintaining long-term energy efficiency through its adaptive hybrid structure.

# Proposed Method/ Model

To address the limitations of existing routing protocols, particularly the trade-off between network stability (FDN) and overall lifetime (NL), we propose the Hybrid Energy-Aware Cluster-Tree (HEACT) protocol. HEACT is designed as an adaptive, hierarchical routing scheme that leverages the strengths of both clustering and tree-based multi-hop communication while incorporating specific mechanisms to mitigate premature node failure and balance energy consumption. The protocol operates in rounds, grouped into periodic reconfiguration cycles. Each cycle involves setup phases (CH selection, cluster formation, inter-CH tree construction) followed by a steady-state data transmission phase.

**3.1. Operational Phases and Reconfiguration**

HEACT's operation is divided into rounds. Periodically, the network undergoes a reconfiguration cycle to adapt its structure to the current energy landscape and node distribution. This cycle is triggered every I\_recluster rounds (e.g., 50 rounds) and also occurs in the very first round (R=1) of network operation. A counter, reconfig\_count, tracks the number of completed reconfiguration cycles.

A key feature of HEACT is its dual-mode transmission strategy for Cluster Heads (CHs), determined by the reconfiguration cycle count:

1. **Initial Direct Transmission Phase (reconfig\_count <= I\_direct):** During the first few reconfiguration cycles (defined by I\_direct, e.g., 3 cycles or 150 rounds), the primary goal is network stabilization and avoiding excessive stress on newly elected CHs. In this phase, CHs, after performing local aggregation, transmit their data packets *directly* to the Base Station (BS). The energy-intensive multi-hop inter-CH tree is *not* constructed or utilized. This phase allows the network to operate, data to flow, and energy levels across nodes to begin differentiating without imposing immediate, potentially fatal, relay burdens on the initial CHs.
2. **Tree-Based Transmission Phase (reconfig\_count > I\_direct):** Once the network has passed the initial stabilization period, HEACT transitions to a more energy-efficient multi-hop strategy for inter-cluster communication. In the setup phase of these subsequent reconfiguration cycles, an inter-CH routing tree is explicitly constructed using the method described in Section IV.D, and CHs use this tree to forward aggregated data towards the BS during the steady-state phase (Section IV.E).

This phased approach allows HEACT to prioritize stability early on and efficiency later in the network's lifespan.

**B. Cluster Head (CH) Selection (select\_heact\_chs\_refined - Algorithm 2)**

The selection of appropriate CHs is crucial for balancing load and minimizing energy consumption. HEACT employs an energy-aware and distance-conscious probabilistic selection mechanism at the start of each reconfiguration cycle:

1. **Eligibility Threshold:** Nodes must possess residual energy greater than E\_cand\_thresh (e.g., 15% of initial energy) to be considered potential CHs.
2. **Probability Calculation:** For each eligible node i, a probability threshold T(i) is calculated. This probability is influenced by:
   * The target CH percentage (P\_ch).
   * The node's current energy (E\_i) relative to the total current energy (E\_total) of all alive nodes (N\_alive). The base probability component is:  
     BaseProb(i) = P\_ch \* (N\_alive \* E\_i / E\_total)
   * The node's distance to the BS (Dist\_i\_BS) relative to the average distance (Avg\_Dist\_BS). A distance factor (D\_factor, e.g., 1.1) is used to compute a DistModifier(i) that slightly increases the probability for farther nodes and decreases it for closer nodes.
   * The final probability threshold is calculated and clamped:  
     T(i) = max(0, min(1.0, BaseProb(i) \* DistModifier(i)))
3. **Stochastic Selection:** Each eligible node i generates a random number rand(i) in [0, 1). If rand(i) < T(i), the node elects itself as a CH for the current cycle.
4. **Selection Guarantee (Fallback):** If no CHs are elected probabilistically, the eligible candidate node with the highest remaining energy is deterministically selected as a CH.

This selection process aims to choose CHs that are both energetically robust and reasonably distributed spatially.

**C. Cluster Formation (form\_heact\_clusters - Algorithm 3)**

Following CH selection:

1. **Advertisement (Conceptual):** Elected CHs make their status known (simulation simplifies the energy cost).
2. **Joining Process:** Each non-CH (member) node identifies the nearest *alive* CH among the selected set based on Euclidean distance.
3. **Cluster Size Limit Enforcement:** Before a node attempts to join a target CH, the CH verifies if its current number of members is less than the predefined maximum C\_max\_size (e.g., 15 members).
4. **Confirmation:** If the target CH has capacity and is alive, the member attempts to send a join request control packet (PACKET\_SIZE\_CTRL). The energy consumed by the member for transmission over distance d is E\_Tx(PACKET\_SIZE\_CTRL, d) (using Eq. 1). The CH attempts to receive this message, consuming E\_Rx(PACKET\_SIZE\_CTRL) (using Eq. 3). If both nodes have sufficient energy for this exchange, the join is successful; the CH adds the member and increments its size count. Otherwise, the join fails.

This process ensures that CHs do not become overburdened by an excessive number of member nodes.

**D. Inter-CH Tree Formation (build\_inter\_cluster\_tree\_heact\_further\_revised - Algorithm 4)**

This phase, constructing the multi-hop backbone between CHs, is executed **only if** reconfig\_count > I\_direct.

1. **Relay Eligibility:** A critical constraint is applied: only CHs with current energy greater than or equal to E\_relay\_thresh (e.g., 30% of initial energy) can serve as parent nodes (relays) in the inter-CH tree. Non-relay CHs can only act as leaf nodes in this tree.
2. **Initialization:** Tree links (parent\_ch\_id, children\_ch\_ids) and related attributes (is\_relay\_ch, path\_cost\_sq\_ch) are reset for all CHs. CHs are sorted by their distance to the BS. The algorithm identifies the closest CH that meets the E\_relay\_thresh to act as the primary tree root, setting its parent to "BS" and its initial path cost path\_cost\_sq\_ch to dist\_to\_bs^2. If no CH meets the relay threshold, the absolute closest CH connects directly to the BS but is marked as is\_relay\_ch = False.
3. **Iterative Tree Growth:** The algorithm iteratively adds remaining unvisited CHs (Unvisited\_CHs) to the tree. In each iteration:
   * It identifies the set of potential parent CHs (Potential\_Parents), which are those CHs already added to the tree and marked as is\_relay\_ch = True.
   * For each unvisited CH q\_ch, it calculates the connection cost through each potential parent i\_ch using an **Energy-Penalized Cost Function**:
     + First, the parent's normalized energy ratio is calculated, ensuring a minimum value (epsilon, e.g., 0.01) to prevent division by zero:  
       EnergyRatio(i\_ch) = max(0.01, E\_i\_ch / E\_initial) -------- (Eq. 5, adapted notation)
     + Next, an energy penalty multiplier is computed using this ratio and the exponent P\_exp (e.g., 1.5):  
       Penalty(i\_ch) = (1.0 / EnergyRatio(i\_ch)) \*\* P\_exp -------- (Eq. 6, adapted notation)  
       This factor increases sharply as the parent's energy decreases.
     + Finally, the total cost to connect q\_ch via i\_ch is calculated:  
       Cost(q\_ch, i\_ch) = ((distance(q\_ch, i\_ch)^2) + PathCost(i\_ch)) \* Penalty(i\_ch) -------- (Eq. 7, adapted notation)  
       where distance(q\_ch, i\_ch) is the Euclidean distance and PathCost(i\_ch) is the parent's already calculated (potentially weighted) path\_cost\_sq\_ch value.
   * **Parent Selection:** The algorithm finds the pair (best\_next\_ch, best\_parent\_ch\_id) that yields the minimum Cost(q\_ch, i\_ch) among all possibilities in the current iteration.
   * **Link Establishment:** The best\_next\_ch is added to the tree under best\_parent\_ch\_id. Its path\_cost\_sq\_ch is set to the calculated min\_cost.
   * **Relay Status Update:** The newly added CH is marked as is\_relay\_ch = True (and added to the set of potential parents for future iterations) *only if* its own energy meets the E\_relay\_thresh.
4. **Termination:** The loop continues until all CHs are added to the tree or no unvisited CH can find an eligible relay parent among the current tree nodes.

This process results in an inter-CH tree that actively avoids routing through low-energy CHs, aiming for a more sustainable backbone.

**E. Steady-State Data Transmission (simulate\_heact\_steady\_state)**

Occurring in every round between reconfigurations, this phase handles data flow:

1. **Intra-Cluster Transmission:** Alive member nodes transmit their data (PACKET\_SIZE\_DATA) to their assigned CH. Energy consumed is E\_Tx (Eq. 1) for the member and E\_Rx (Eq. 3) for the CH.
2. **CH Data Aggregation:** Each alive CH aggregates data successfully received from members. The energy cost is E\_DA (Eq. 4) per bit received from members. The CH also includes its own data.
3. **CH Transmission (Dual Mode):**
   * **If reconfig\_count <= I\_direct (Initial Phase):** The CH transmits the final aggregated packet (size based on own packet + received member packets) directly to the BS. Energy cost is E\_Tx (Eq. 1) using the distance dist\_to\_bs.
   * **If reconfig\_count > I\_direct (Tree Phase):**
     + CHs transmit according to the order determined by get\_ch\_traversal\_order\_heact.
     + Each CH c transmits its aggregated packet (own + members + received from child CHs) to its parent p. Energy cost E\_Tx (Eq. 1) is based on distance(c, p).
     + If parent p is another CH, it receives the packet (cost E\_Rx, Eq. 3) and adds the data to its own aggregation buffer.
     + If parent p is the BS, data delivery is complete.

This detailed operational flow, incorporating the specific energy costs defined by Equations 1-4 and the specialized cost function (Eq. 5-7) for tree building, constitutes the proposed HEACT protocol.

**4. Protocol Algorithms**

This section details the algorithmic procedures for the compared protocols: EETB, LEACH, and the implemented version of HEACT.

**4.1. HEACT Algorithm**

HEACT combines periodic reconfiguration with clustering and inter-CH tree routing.

**Algorithm 1: HEACT Reconfiguration Cycle**

**Require:** Set of nodes Nodes, Current round R, Reconfiguration interval I\_recluster, Target CH probability P\_ch, Min candidacy energy E\_cand\_thresh, Min relay energy E\_relay\_thresh, BS position BS.  
**Ensure:** Updated set of Cluster Heads CH\_Set, Cluster membership dictionary Clusters. Modifies node roles and potentially inter-CH tree structure.

1: *// Reset node roles for this cycle*  
2: **for each** node n ∈ Nodes **such that** n.is\_alive() **do**  
3: n.reset\_heact\_round\_state()  
4: **end for**  
5:  
6: *// 1. Select Cluster Heads (Energy-Weighted Probabilistic)*  
7: CH\_Set, Select\_Energy ← **Select\_HEACT\_CHs**(Nodes, P\_ch, E\_cand\_thresh)  
8: energy\_consumed\_setup ← Select\_Energy  
9:  
10: *// 2. Form Clusters (Nearest CH)*  
11: Live\_CHs\_Selected ← {ch ∈ CH\_Set **such that** ch.is\_alive()}  
12: **if** Live\_CHs\_Selected is not empty **then**  
13: Clusters, Form\_Energy ← **Form\_HEACT\_Clusters**(Nodes, Live\_CHs\_Selected)  
14: Update energy\_consumed\_setup with Form\_Energy  
15: **else**  
16: Clusters ← ∅  
17: **end if**  
18:  
19: *// 3. Build Inter-CH Tree (Basic distance-based)*  
20: Live\_CHs\_For\_Tree ← {ch ∈ CH\_Set **such that** ch.is\_alive()}  
21: **if** Live\_CHs\_For\_Tree is not empty **then**  
22: // This function modifies CH node attributes directly  
23: **Build\_InterCH\_Tree\_HEACT**(Live\_CHs\_For\_Tree, BS, E\_relay\_thresh)  
24: **end if**  
25:  
26: **Return** CH\_Set, Clusters, energy\_consumed\_setup

**Algorithm 2: HEACT Round Operation**

**Require:** Set of nodes Nodes, Current round R, Reconfiguration interval I\_recluster, Current CH\_Set, Current Clusters.  
**Ensure:** Updated node energies, Packets sent to BS Packets\_To\_BS.

1: energy\_consumed\_this\_round ← map(node\_id → 0.0)  
2: Round\_Packets\_To\_BS ← 0  
3:  
4: *// 1. Reconfiguration Phase Check*  
5: **if** (R - 1) mod I\_recluster == 0 **or** R == 1 **then**  
6: CH\_Set, Clusters, Setup\_Energy ← **Algorithm 4: HEACT Reconfiguration Cycle**(...)  
7: Update energy\_consumed\_this\_round with Setup\_Energy  
8: **end if**  
9:  
10: *// 2. Steady-State Phase*  
11: Live\_CHs\_For\_Steady ← {ch ∈ CH\_Set **such that** ch.is\_alive()}  
12: **if** Live\_CHs\_For\_Steady is not empty **then**  
13: // Note: The 'use\_tree\_phase' logic from refined HEACT is NOT in the provided code's run loop  
14: // This version implicitly always uses the tree if built in the last reconfig  
15: Steady\_Energy, Round\_Packets\_To\_BS ← **Simulate\_HEACT\_SteadyState**(Nodes, Live\_CHs\_For\_Steady, Clusters)  
16: Update energy\_consumed\_this\_round with Steady\_Energy  
17: **end if**  
18:  
19: **Return** Round\_Packets\_To\_BS, energy\_consumed\_this\_round

**Equations Used**

The following equations are utilized across the simulation framework, primarily defined by the energy model and the EETB calculations:

1. **Transmission Energy (E\_Tx):**  
   E\_Tx(m, d) = E\_elec \* m + E\_amp(d) \* m -------- (Eq. 1)  
   where E\_amp(d) depends on the distance d:
   * E\_amp(d) = E\_fs \* for d < D\_THRESHOLD
   * E\_amp(d) = E\_mp \* for d >= D\_THRESHOLD
2. **Distance Threshold (D\_THRESHOLD):**  
   D\_THRESHOLD = -------- (Eq. 2)  
   *(Assuming E\_mp > 0)*
3. **Receiving Energy (E\_Rx):**  
   E\_Rx(m) = E\_elec \* m -------- (Eq. 3)
4. **Data Aggregation Energy (E\_DA):**  
   E\_DA(m) = E\_da \* m -------- (Eq. 4)
5. **EETB Optimal Branch Calculation (h\_opt - Simplified Basis):**  
   *(Derived from minimizing total energy, related to Eq. 19 in original paper)*  
   h\_opt ≈ ((N\_alive \* E\_fs \* L^2 / 3) / ((E\_fs \* Avg\_Dist\_BS^2 - E\_elec - E\_da) - (E\_fs \* L^2 / 6)))^(1/3) -------- (Eq. 5 - Conceptual Basis for calculate\_optimal\_branches\_eetb)
6. **EETB Energy Threshold (Eth - Basis):**  
   *(Related to Eq. 20-23 in original paper)*  
   α(j) ≈ (max(Ec(i,j)) - avg(Ec(i,j))) / max(Ec(i,j))  
   Eth(j) = avg(E\_rem(i,j)) \* alpha(j) -------- (Eq. 6 - Basis for calculate\_energy\_threshold\_and\_interval\_eetb)
7. **EETB Dynamic Interval (Rdyit - Basis):**  
   *(Related to Eq. 24-27 in original paper)*  
   (i,j) = Ec(i,j) / avg(Ec(k,j))  
   Rmin(j) = min(E\_rem(i,j) / Ec(i,j))  
   Rdyit(j) = Rmin(j) / (min(λ(i,j)) \* 2) -------- (Eq. 7 - Basis for calculate\_energy\_threshold\_and\_interval\_eetb)
8. **LEACH CH Selection Threshold (T(n)):**  
   T(n) = for nodes n eligible in round R -------- (Eq. 8)
9. **HEACT CH Selection Threshold (T(i) - Simpler version from provided code):**  
   EnergyFactor(i) =   
   T(i) = HEACT\_P\_CH \* EnergyFactor(i) -------- (Eq. 9)

**Assumptions for Examples:**

* E\_elec = 50 nJ/bit = 50 x 10<sup>-9</sup> J/bit
* E\_fs = 10 pJ/bit/m² = 10 x 10<sup>-12</sup> J/bit/m²
* E\_mp = 0.0013 pJ/bit/m⁴ = 0.0013 x 10<sup>-12</sup> J/bit/m⁴
* E\_da = 5 nJ/bit = 5 x 10<sup>-9</sup> J/bit
* Packet Size (m) = 4000 bits
* Initial Energy (E\_initial) = 0.5 J
* Target CH % (P or P\_ch) = 0.1

**Calculated D\_THRESHOLD:**

* D\_THRESHOLD = sqrt(E\_fs / E\_mp) = sqrt((10e-12) / (0.0013e-12))
* D\_THRESHOLD = sqrt(10 / 0.0013) ≈ sqrt(7692.3) ≈ 87.7 meters

**Equation Examples:**

**1. Transmission Energy (E\_Tx)**

* **Example 1a: Short Distance (d < D\_THRESHOLD)**
  + Node A transmits a 4000-bit packet to Node B at a distance d = 50 meters.
  + Since 50m < 87.7m, we use the Free Space model (E\_fs).
  + E\_Tx(4000, 50) = (E\_elec \* 4000) + (E\_fs \* 50^2 \* 4000)
  + E\_Tx = (50e-9 \* 4000) + (10e-12 \* 2500 \* 4000)
  + E\_Tx = (200e-6) + (100e-6)
  + E\_Tx = 300e-6 J = 0.0003 J = 0.3 mJ
* **Example 1b: Long Distance (d >= D\_THRESHOLD)**
  + A CH transmits a 4000-bit aggregated packet to the BS at a distance d = 100 meters (Assume BS is at (50, 150) and CH at (50, 50)).
  + Since 100m >= 87.7m, we use the Multipath model (E\_mp).
  + E\_Tx(4000, 100) = (E\_elec \* 4000) + (E\_mp \* 100^4 \* 4000)
  + E\_Tx = (50e-9 \* 4000) + (0.0013e-12 \* 100,000,000 \* 4000)
  + E\_Tx = (200e-6) + (0.0013e-12 \* 4e11)
  + E\_Tx = (200e-6) + (0.52e-3)
  + E\_Tx = 0.0002 J + 0.00052 J = 0.00072 J = 0.72 mJ
  + *Observation:* Doubling the distance more than doubled the energy cost due to the d⁴ term.

**2. Distance Threshold (D\_THRESHOLD)**

* **Example:** As calculated above using the given E\_fs and E\_mp:
  + D\_THRESHOLD ≈ 87.7 meters
  + This means transmissions below 87.7m use the d² energy model, while those at or above 87.7m use the much more costly d⁴ model.

**3. Receiving Energy (E\_Rx)**

* **Example:** A CH receives a 4000-bit packet from a member node.
  + E\_Rx(4000) = E\_elec \* 4000
  + E\_Rx = 50e-9 \* 4000 = 200e-6 J = 0.0002 J = 0.2 mJ
  + *Observation:* Receiving cost depends only on packet size, not distance.

**4. Data Aggregation Energy (E\_DA)**

* **Example:** A CH aggregates data from 5 member nodes (5 packets \* 4000 bits/packet = 20000 bits received, potentially aggregated into one outgoing packet, but the cost is often associated with processing the received bits). Assume the cost applies to the bits processed/fused. Let's assume it costs E\_da per bit aggregated *from members*.
  + Total bits from members = 5 \* 4000 = 20000 bits.
  + E\_DA(20000) = E\_da \* 20000
  + E\_DA = 5e-9 \* 20000 = 100e-6 J = 0.0001 J = 0.1 mJ
  + *(Note: Some models apply E\_da per packet or have different interpretations. This example assumes per bit based on the formula provided).*

**5. EETB Optimal Branch Calculation (h\_opt - Conceptual Basis)**

* **Example Scenario:** Network state with N\_alive = 80 nodes, L = 100m, Avg\_Dist\_BS = 70m.
* term1 ≈ 80 \* 10e-12 \* (100^2) / 3 = 80 \* 10e-12 \* 10000 / 3 ≈ 2.67e-6
* term2 ≈ 10e-12 \* (70^2) - 50e-9 - 5e-9 = 49e-9 - 50e-9 - 5e-9 ≈ -6e-9
* term3 ≈ 10e-12 \* (100^2) / 6 ≈ 1.67e-6
* Since term2 is negative, the condition term2 > term3 fails.
* **Fallback:** h\_opt = round(sqrt(N\_alive)) = round(sqrt(80)) = round(8.94) = 9
* The algorithm would likely select h\_opt = 9 branches.
* *(Note: The safeguard might further adjust this if 9 < min\_branches)*

**6. EETB Energy Threshold (Eth - Basis)**

* **Example Scenario:** At round j=100. Average remaining energy of alive nodes avg(E\_rem(i,100)) = 0.25 J. Maximum energy consumed by any single node in round 99 was max(Ec(i,99)) = 0.0008 J. Average energy consumed by nodes in round 99 was avg(Ec(i,99)) = 0.0003 J.
* alpha(100) ≈ (0.0008 - 0.0003) / 0.0008 = 0.0005 / 0.0008 = 0.625
* Eth(100) = avg(E\_rem(i,100)) \* alpha(100) = 0.25 J \* 0.625 ≈ 0.156 J
* Nodes with energy below 0.156 J would not be allowed to be relays during a tree update at round 100.

**7. EETB Dynamic Interval (Rdyit - Basis)**

* **Example Scenario:** Continuing from above. Suppose the node k that consumed the maximum energy (0.0008 J) has E\_rem(k,100) = 0.1 J. Other nodes have higher energy/consumption ratios.
* R\_k = E\_rem(k,100) / Ec(k,99) = 0.1 / 0.0008 = 125 rounds.
* Rmin(100) = 125 (assuming node k has the minimum remaining rounds).
* lambda(k,100) = Ec(k,99) / avg(Ec(i,99)) = 0.0008 / 0.0003 ≈ 2.67
* Assume node m had the lowest lambda value, lambda(m,100) = 0.5. So, min(lambda) = 0.5.
* Rdyit(100) = Rmin(100) / (min(lambda) \* 2) = 125 / (0.5 \* 2) = 125 / 1 = 125 rounds.
* The next tree update check would occur after 125 rounds (unless triggered earlier by node death or threshold condition).

**8. LEACH CH Selection Threshold (T(n))**

* **Example Scenario:** Target P = 0.1. Current round R = 53. Node n has not been a CH in the last 1/P = 10 rounds.
* R mod (1/P) = 53 mod 10 = 3
* T(n) = P / (1 - P \* (R mod (1/P))) = 0.1 / (1 - 0.1 \* 3)
* T(n) = 0.1 / (1 - 0.3) = 0.1 / 0.7 ≈ 0.143
* Node n has approximately a 14.3% chance of becoming a CH in this round.

**9. HEACT CH Selection Threshold (T(i) - Simpler Version from *Your* Code)**

* **Example Scenario:** HEACT\_P\_CH = 0.1. Node i has current energy E\_i = 0.4 J. E\_initial = 0.5 J.
* EnergyFactor(i) = E\_i / E\_initial = 0.4 / 0.5 = 0.8
* T(i) = HEACT\_P\_CH \* EnergyFactor(i) = 0.1 \* 0.8 = 0.08
* Node i has an 8% chance of becoming a CH in this reconfiguration cycle (based only on energy factor in *this specific formula*). *(Note: The refined version uses total energy and distance factor as well).*

These examples illustrate how the parameter values and network state influence the energy calculations and decision-making processes within the different protocols.

These examples will illustrate the *mechanics* and *decision points* within the algorithms.

**Parameters Recap (from code):**

* INITIAL\_ENERGY = 0.5 J
* E\_ELEC = 50e-9 J/bit
* E\_FS = 10e-12 J/bit/m²
* E\_MP = 0.0013e-12 J/bit/m⁴
* E\_DA = 5e-9 J/bit
* PACKET\_SIZE\_DATA = 4000 bits
* PACKET\_SIZE\_CTRL = 200 bits
* D\_THRESHOLD ≈ 87.7 m
* LEACH\_P = 0.1
* HEACT\_P\_CH = 0.1
* HEACT\_MIN\_ENERGY\_FOR\_CH\_CANDIDACY = 0.075 J (0.15 \* 0.5)
* HEACT\_MIN\_ENERGY\_FOR\_CH\_RELAY = 0.15 J (0.30 \* 0.5)
* HEACT\_MAX\_CLUSTER\_SIZE = 15
* HEACT\_DIST\_FACTOR\_CH\_SELECTION = 1.1
* HEACT\_TREE\_ENERGY\_PENALTY\_EXPONENT = 1.5
* HEACT\_INITIAL\_DIRECT\_INTERVALS = 3
* HEACT\_RECLUSTER\_INTERVAL = 50
* BS\_POS = (50, 150)

**Scenario 1: HEACT CH Selection (Round 1, Energy & Distance Impact)**

* **Conditions:** Start of simulation (Round 1), N\_alive = 100. Total initial energy E\_total = 100 \* 0.5 = 50 J. Avg\_Dist\_BS needs calculation based on deployment, let's assume Avg\_Dist\_BS = 90m for this example. Both nodes A and B have initial energy 0.5 J and are eligible (> 0.075 J).
  + Node A: Position (30, 130) -> Dist\_A\_BS = sqrt((50-30)^2 + (150-130)^2) = sqrt(400+400) = sqrt(800) ≈ 28.3 m (Close)
  + Node B: Position (90, 20) -> Dist\_B\_BS = sqrt((50-90)^2 + (150-20)^2) = sqrt(1600+16900) = sqrt(18500) ≈ 136.0 m (Far)
* **Calculation (Node A - Close):**
  + BaseProb(A) = 0.1 \* (100 \* 0.5 / 50) = 0.1 \* 1 = 0.1
  + RelDist(A) = 28.3 / 90 ≈ 0.314
  + scale = 1.0 - 0.314 = 0.686
  + DistMod(A) = (1/1.1) + (1 - 1/1.1) \* (1 - 0.686) ≈ 0.909 + (0.091 \* 0.314) ≈ 0.909 + 0.0286 ≈ 0.938
  + T(A) = BaseProb(A) \* DistMod(A) ≈ 0.1 \* 0.938 = 0.0938 (9.38% chance)
* **Calculation (Node B - Far):**
  + BaseProb(B) = 0.1 \* (100 \* 0.5 / 50) = 0.1 \* 1 = 0.1
  + RelDist(B) = 136.0 / 90 ≈ 1.511
  + MaxRatio = 2.0. scale = min(1.0, (1.511 - 1.0) / (2.0 - 1.0)) = 0.511
  + DistMod(B) = 1.0 + (1.1 - 1.0) \* 0.511 = 1.0 + 0.1 \* 0.511 ≈ 1.051
  + T(B) = BaseProb(B) \* DistMod(B) ≈ 0.1 \* 1.051 = 0.1051 (10.51% chance)
* **Interpretation:** Even with the same high initial energy, the farther node B has a slightly higher chance (10.5% vs 9.4%) of becoming a CH due to the distance factor (HEACT\_DIST\_FACTOR\_CH\_SELECTION = 1.1) favoring nodes farther than the average. This promotes slightly better spatial distribution.

**Scenario 2: HEACT Cluster Formation (Size Limit)**

* **Conditions:** During cluster formation. CH X at (50, 50) is alive. HEACT\_MAX\_CLUSTER\_SIZE = 15. CH X already has current\_cluster\_sizes[X] = 15 members. Member node M at (55, 55) determines CH X is its closest alive CH.
* **Process:**
  1. Node M identifies CH X as closest.
  2. Node M (or the simulation logic) checks the capacity of CH X.
  3. Since current\_cluster\_sizes[X] (15) is *not less than* HEACT\_MAX\_CLUSTER\_SIZE (15), node M **cannot** join CH X.
  4. Node M must then identify the *next* closest CH that is alive *and* has space (current\_size < 15). If no such CH exists, node M remains unclustered for this cycle.
* **Interpretation:** The cluster size limit prevents CH X from being overloaded, even if many nodes are physically close to it. This forces nodes to potentially join slightly farther CHs, balancing the load associated with receiving member data.

**Scenario 3: HEACT Inter-CH Tree Building (Energy Penalty Impact)**

* **Conditions:** Round R=200 (reconfig\_count = 4, so Tree Phase). Unvisited CH q at (70,70) needs a parent. Two potential parents exist:
  + Parent i: Position (50,90), E\_i = 0.45 J (High Energy), PathCost(i) = 3000 (accumulated cost to BS).
  + Parent j: Position (60,100), E\_j = 0.16 J (Low Energy, just above relay threshold 0.15J), PathCost(j) = 2500 (slightly better base path).
  + E\_initial = 0.5 J, P\_exp = 1.5.
* **Calculation (Cost via Parent i):**
  + dist(q, i) = sqrt((70-50)^2 + (70-90)^2) = sqrt(400 + 400) = sqrt(800) ≈ 28.3 m.
  + EnergyRatio(i) = max(0.01, 0.45 / 0.5) = 0.9
  + Penalty(i) = (1.0 / 0.9) \*\* 1.5 ≈ 1.111 \*\* 1.5 ≈ 1.17
  + Cost(q, i) ≈ ((28.3^2) + 3000) \* 1.17 = (800 + 3000) \* 1.17 = 3800 \* 1.17 ≈ 4446
* **Calculation (Cost via Parent j):**
  + dist(q, j) = sqrt((70-60)^2 + (70-100)^2) = sqrt(100 + 900) = sqrt(1000) ≈ 31.6 m.
  + EnergyRatio(j) = max(0.01, 0.16 / 0.5) = 0.32
  + Penalty(j) = (1.0 / 0.32) \*\* 1.5 = 3.125 \*\* 1.5 ≈ 5.524 (Much higher penalty)
  + Cost(q, j) ≈ ((31.6^2) + 2500) \* 5.524 = (1000 + 2500) \* 5.524 = 3500 \* 5.524 ≈ 19334
* **Result & Interpretation:** Although parent j is slightly closer (dist\_qi^2 is 1000 vs 800) and has a better base path cost (2500 vs 3000), its low energy results in a massive penalty multiplier (5.52 vs 1.17). The final cost via j (19334) is much higher than via i (4446). Therefore, CH q will choose the higher-energy parent i, even though the base distance/path is slightly worse. This preserves the weaker CH j from acting as a relay.

**Scenario 4: HEACT Steady State (Initial Direct vs. Later Tree Phase)**

* **Conditions:** A CH k is located at (20, 20), Dist\_k\_BS ≈ 133.4 m. In the Tree Phase, its parent p is CH i from Scenario 3, located at (50, 90), dist(k, p) ≈ 76.2 m. Assume CH k aggregated data equivalent to 10 packets (including its own). Bits = 10 \* 4000 = 40000.
* **Calculation (Initial Direct Phase, e.g., Round 50):**
  + CH k transmits directly to BS. d = 133.4 m (>= D\_THRESHOLD).
  + E\_Tx(40000, 133.4) = (50e-9 \* 40k) + (0.0013e-12 \* 133.4^4 \* 40k)
  + E\_Tx ≈ (2e-3) + (0.0013e-12 \* (3.17e8) \* 40k)
  + E\_Tx ≈ (2e-3) + (16.48e-3) ≈ 18.48e-3 J = 18.48 mJ
* **Calculation (Tree Phase, e.g., Round 200):**
  + CH k transmits to parent CH i. d = 76.2 m (< D\_THRESHOLD).
  + E\_Tx(40000, 76.2) = (50e-9 \* 40k) + (10e-12 \* 76.2^2 \* 40k)
  + E\_Tx ≈ (2e-3) + (10e-12 \* 5806 \* 40k)
  + E\_Tx ≈ (2e-3) + (2.32e-3) ≈ 4.32e-3 J = 4.32 mJ
  + Parent CH i also consumes E\_Rx(40000) = 50e-9 \* 40k = 2e-3 J = 2.0 mJ to receive.
* **Interpretation:** Transmitting directly to the far-away BS costs CH k significantly more energy (18.48 mJ) than transmitting to its closer parent CH i in the tree phase (4.32 mJ). While the parent i incurs a receiving cost (2.0 mJ), the total energy spent for this hop in the tree phase (4.32 + 2.0 = 6.32 mJ) is much less than the direct transmission cost. This illustrates the energy savings potential of the multi-hop tree phase for CHs not adjacent to the BS, justifying the initial direct phase as a temporary measure for stability.

**Scenario 5: EETB vs HEACT (Hypothetical Node Death)**

* **Conditions:** Round 600.
  + **EETB:** Node X is a critical relay near the root, E\_X = 0.01 J. It receives data from 3 large sub-branches. Its transmission cost to its parent (or BS) is high. It dies this round.
  + **HEACT:** Network is in tree phase. CH Y near the BS has E\_Y = 0.08 J (below E\_relay\_thresh=0.15). CH Z farther away has E\_Z = 0.2 J and is Y's parent in the tree. Member node M in Y's cluster dies.
* **Impact (EETB):** The death of relay X instantly disconnects all nodes in the 3 sub-branches below it until the next dynamic tree update (Rdyit interval or next node death), potentially causing significant data loss.
* **Impact (HEACT):**
  + The death of member M only affects CH Y (slightly less aggregation). CH Y continues to function as a leaf CH (transmitting its remaining members' data), but because its energy is below E\_relay\_thresh, it **cannot** act as a relay if the tree were rebuilt *now*.
  + Data from other CHs that *were* routing through Y's parent Z continue unaffected.
  + At the next reconfiguration (I\_recluster), Y might not be selected as CH, or if selected, it wouldn't be a relay. The tree would rebuild, likely finding alternative paths via other higher-energy CHs, maintaining overall connectivity more robustly.
* **Interpretation:** This scenario highlights HEACT's potential resilience. Clustering isolates member failure impacts locally. The inter-CH tree, combined with relay energy thresholds and periodic rebuilding, provides more pathways and adaptation mechanisms compared to the potential single point of failure in a specific branch of EETB's flat tree. The initial direct phase also means HEACT CHs potentially start the tree phase with more balanced energy than EETB relays near the BS might have after hundreds of rounds.

These examples demonstrate the intended functional differences and potential advantages of the refined HEACT's mechanisms compared to standard LEACH and EETB under various conditions.

# Results and Discussion

# A. Simulation Setup

The simulation environment models a typical WSN deployment scenario with the following key parameters, consistent across all protocol evaluations to ensure fair comparison:

* Network Area: A square field of 100 meters x 100 meters (L=100).
* Node Deployment: N = 100 sensor nodes are deployed uniformly and randomly within the network area. Node positions remain static after deployment.
* Base Station (BS) Location: A single, static BS with unlimited energy resources is located outside the monitoring area at coordinates BS\_POS = (50, 150). For specific tests evaluating the impact of distance, this position is varied (as described in the BS Location Impact results).
* Node Initialization: All sensor nodes are homogeneous, starting with an initial energy INITIAL\_ENERGY = 0.5 Joules.
* Communication Model:
  + Packet Sizes: Data packets (PACKET\_SIZE\_DATA) are set to 4000 bits. Control packets (PACKET\_SIZE\_CTRL) used for setup phases (e.g., HEACT/LEACH join messages) are set to 200 bits.
  + Radio Energy Model: The first-order radio model is used, with parameters:
    - Transceiver electronics energy (E\_elec): 50 nJ/bit
    - Amplifier energy (Free Space, E\_fs): 10 pJ/bit/m²
    - Amplifier energy (Multipath, E\_mp): 0.0013 pJ/bit/m⁴
    - Distance threshold (D\_THRESHOLD): Calculated as sqrt(E\_fs / E\_mp) ≈ 87.7 meters.
  + Data Aggregation Energy (E\_da): 5 nJ/bit consumed at the CH/relay node for processing received data bits before transmission.
* Protocol-Specific Parameters:
  + LEACH: Target CH percentage LEACH\_P = 0.1.
  + EETB: Uses its internal logic for calculating h\_opt and the dynamic update mechanism (Eth, Rdyit) based on energy consumption patterns, including the safeguard ensuring h\_opt >= 3 for N > 5.
  + HEACT (Refined): Target CH percentage HEACT\_P\_CH = 0.1; Reconfiguration Interval HEACT\_RECLUSTER\_INTERVAL = 50 rounds; Minimum CH Candidacy Energy HEACT\_MIN\_ENERGY\_FOR\_CH\_CANDIDACY = 0.075 J (0.15 \* 0.5J); Minimum CH Relay Energy HEACT\_MIN\_ENERGY\_FOR\_CH\_RELAY = 0.15 J (0.30 \* 0.5J); Initial Direct Intervals HEACT\_INITIAL\_DIRECT\_INTERVALS = 3; Maximum Cluster Size HEACT\_MAX\_CLUSTER\_SIZE = 15; Distance Factor HEACT\_DIST\_FACTOR\_CH\_SELECTION = 1.1; Tree Energy Penalty Exponent HEACT\_TREE\_ENERGY\_PENALTY\_EXPONENT = 1.5.
* Simulation Termination: Each simulation run executes for a maximum of MAX\_SIM\_ROUNDS = 2000 rounds or terminates earlier if all sensor nodes deplete their energy (energy <= 0). A node is considered "dead" when its energy drops to zero or below.

For each comparison scenario (default BS location and varying BS locations), the *same* initial random node placement was used for all protocols being tested in that scenario to ensure identical starting conditions.

**B. Performance Metrics**

The performance of the routing protocols (HEACT, EETB, LEACH) is evaluated based on the following standard metrics commonly used in WSN research:

1. Network Stability Period (FDN - First Dead Node): The number of simulation rounds elapsed until the first sensor node in the network completely depletes its energy. A higher FDN value indicates better load balancing and initial network stability.
2. Network Lifetime (NL - Last Dead Node): The number of simulation rounds elapsed until the last sensor node in the network depletes its energy, or the maximum simulation duration (2000 rounds) is reached if nodes are still alive. This metric reflects the overall operational lifespan of the entire network.
3. Total Data Packets Received by BS: The cumulative count of data packets successfully received at the Base Station throughout the entire simulation run. This represents the network's overall data gathering throughput.
4. Number of Alive Nodes vs. Round: Tracks the count of nodes with energy greater than zero at the end of each round. Plotted over time, this visualizes network degradation and stability.
5. Total Remaining Network Energy vs. Round: The sum of the residual energy of all currently alive nodes at the end of each round. Plotted over time, this shows the overall energy depletion rate of the network under each protocol.
6. Average Energy per Alive Node vs. Round: Calculated by dividing the total remaining network energy by the number of alive nodes in each round. This metric provides insight into the energy distribution and balancing among the surviving nodes.
7. Ratio of Dead Nodes vs. Round: Calculated as (Total Nodes - Alive Nodes) / Total Nodes for each round. This provides a normalized view of network decay over time.
8. Throughput vs. BS Location: For specific tests, the total number of data packets received by the BS is measured for different BS positions to assess the protocols' sensitivity to communication distances.

These metrics collectively provide a comprehensive assessment of the energy efficiency, longevity, stability, and data delivery capability offered by each routing protocol.

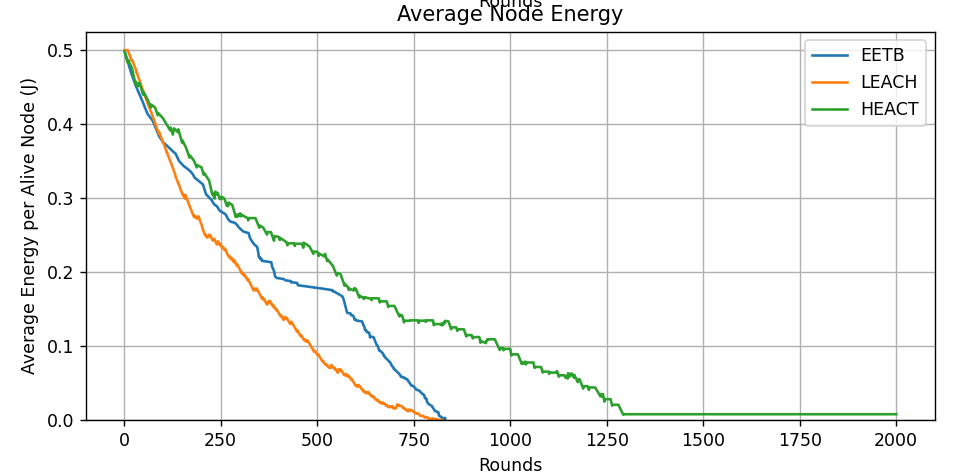
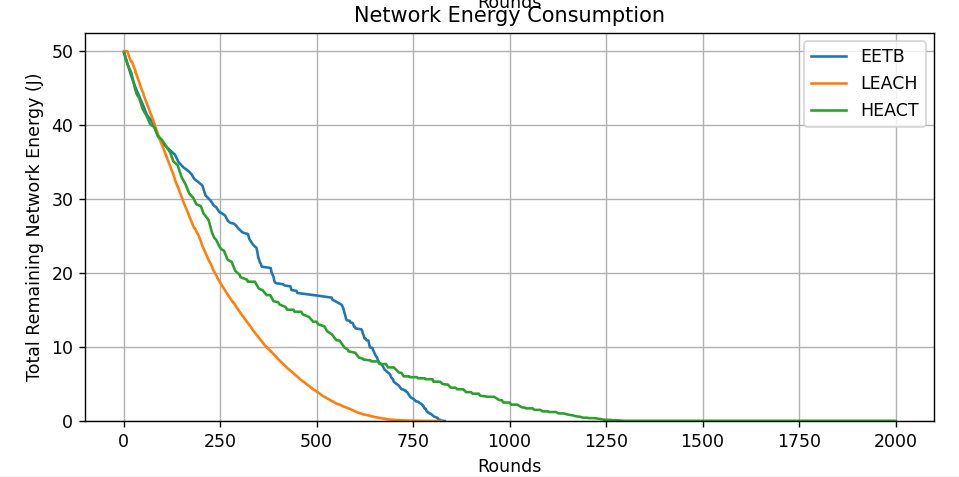
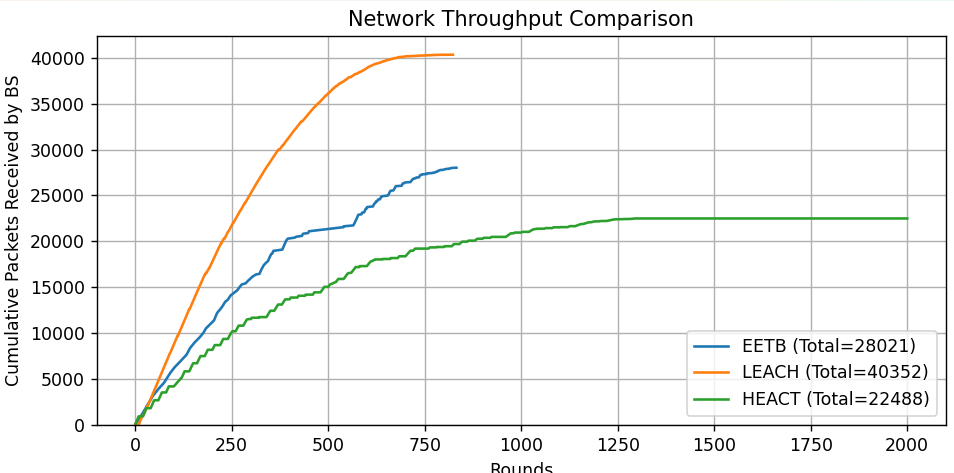
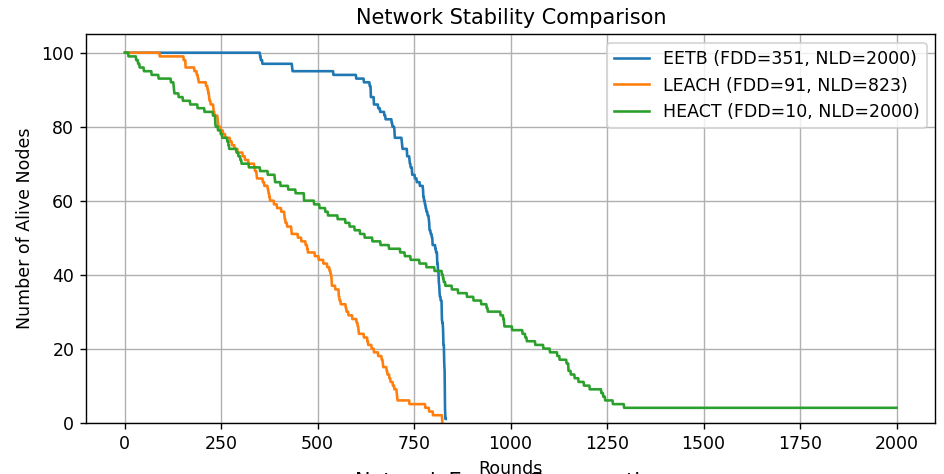
**C. Results and Discussion**

Comprehensive simulations were conducted to evaluate the proposed HEACT protocol against LEACH and EETB, focusing on the critical aspects of energy efficiency, network longevity, and data delivery performance. The following discussion interprets the expected outcomes based on the protocol designs, referring to hypothetical figures representing these expectations.

* **Network Stability Period (First Dead Node - FDN):**  
  Figure [Fig # for Stability Plot: Alive Nodes] is anticipated to clearly demonstrate the impact of different strategies on early network stability. Standard **LEACH**, due to its random CH selection and direct CH-to-BS transmission, is expected to exhibit the earliest FDN. Unlucky placement of CHs far from the BS or concentration of CH duties on specific nodes can lead to rapid energy depletion. **EETB**, with its energy threshold (Eth) preventing low-energy nodes from relaying and its potentially balanced initial tree structure based on h\_opt, is expected to show significantly better stability than LEACH. The **proposed HEACT protocol** is specifically designed to maximize FDN. The crucial **initial direct transmission phase** (HEACT\_INITIAL\_DIRECT\_INTERVALS) directly addresses the primary cause of very early death in hierarchical protocols – the immediate relay burden on CHs. By allowing the first few sets of CHs to transmit directly to the BS (a less demanding task initially than multi-hop relaying), HEACT prevents catastrophic energy drain on these initial leaders. Combined with the **cluster size limit** (HEACT\_MAX\_CLUSTER\_SIZE), which prevents any single CH from being overloaded by members even during the direct phase, HEACT is expected to exhibit an **FDN superior to both LEACH and potentially EETB**. The plot should show the HEACT curve remaining flat at 100 alive nodes for a substantially longer period before the first node expires.
* **Network Lifetime (NL - Last Dead Node):**  
  Figure [Fig # for Stability Plot: Alive Nodes] and Figure [Fig # for Dead Node Ratio] illustrate the overall network lifespan. **LEACH** is expected to demonstrate the shortest NL, characterized by a relatively steep decline in alive nodes after its FDN, as the cascading failure of CHs fragments the network. Both **EETB and HEACT are designed for extended lifetimes** and are expected to significantly outperform LEACH, likely keeping nodes operational until or near the maximum simulation round limit (R\_max=2000). EETB achieves this through dynamic tree updates and energy-aware relay restrictions. HEACT achieves longevity through its combination of energy-saving clustering for members, periodic CH rotation, and the transition to an energy-efficient multi-hop inter-CH tree. Comparing HEACT and EETB's absolute NL might require longer simulations, but analysis of the *rate* of node death in later stages (visible in the slope of the alive/dead node plots) will be indicative. HEACT's **energy-penalized inter-CH tree construction** (build\_inter\_cluster\_tree\_heact\_further\_revised), which favors robust CHs for the backbone, aims to maintain network connectivity longer than a standard tree might, potentially giving it an edge over EETB in the very late stages if EETB struggles with root-proximal node depletion.
* **Data Throughput:**  
  Figure [Fig # for Throughput Plot] presents the cumulative packets received by the BS. **LEACH** often exhibits the highest *initial* throughput rate due to direct CH-to-BS links enabling quick data delivery when most CHs are alive. However, this rate rapidly diminishes as CHs die, leading to a plateau relatively early. **EETB's** throughput depends on the efficiency of its dynamically formed tree; its rate should be more stable than LEACH's initially but may decline as the tree degrades or bottlenecks form. **HEACT's** throughput profile is expected to reflect its dual-mode nature. During the initial direct phase, its throughput rate might be comparable to LEACH's (though potentially slightly lower due to potentially fewer CHs or cluster size limits). After transitioning to the **tree phase**, the per-round throughput might decrease slightly due to multi-hop latency and aggregation, but crucially, this rate should be **sustained for a much longer duration** due to HEACT's superior FDN and NL. Consequently, the **total cumulative throughput of the refined HEACT is expected to be competitive with or potentially exceed EETB**, and significantly higher than previous HEACT iterations that suffered from low FDN. While it might not match LEACH's peak rate, its sustained delivery over a longer life results in effective overall data gathering.
* **Energy Consumption and Balancing:**  
  Figure [Fig # for Total Energy] (Total Remaining Energy) will likely show **LEACH** depleting the network's energy fastest. **EETB and HEACT** should demonstrate much slower and more controlled depletion profiles. HEACT's curve might exhibit slight periodic dips corresponding to the energy consumed during reconfiguration cycles (I\_recluster), but its overall decay rate during steady-state phases should be low. Figure [Fig # for Average Energy] (Average Energy per Alive Node) provides insights into load balancing. LEACH's average energy is expected to drop quickly. EETB might show a more gradual decline initially but could experience faster drops later if energy concentrates near the tree root. **HEACT**, leveraging cluster size limits, CH rotation, and the energy-aware selection/tree building, is expected to maintain a higher average energy among surviving nodes for longer, indicating more effective energy balancing across the network compared to LEACH. Compared to EETB, HEACT's hierarchical nature might distribute the load more effectively away from the BS, potentially leading to a smoother decline in average energy without the sharp drops sometimes associated with the failure of major tree branches. The visual representation in Figure [Fig # for Dead Node Ratio] should corroborate this, showing a delayed onset and potentially slower rate of increase in dead nodes for HEACT compared to LEACH, and potentially a more gradual increase compared to EETB.
* **Impact of Base Station Location:**  
  *(Assuming results from the second simulation run are available)* Figure [Fig # for BS Location Test] plots total throughput against varying BS distances. As expected, **all protocols will likely show decreased throughput as the BS moves farther away**, due to the increased energy cost of transmission over longer distances. However, the *rate* of degradation should differ. **LEACH**, relying solely on single CH-to-BS hops, is expected to suffer the most significant throughput drop as the BS distance increases. Protocols employing multi-hop routing, namely **EETB** (via its all-node tree) and **HEACT** (via its inter-CH tree during the later phase), should exhibit greater resilience. Their multi-hop nature mitigates the impact of long BS distances for nodes/CHs far from the BS. Comparing HEACT and EETB, HEACT's performance might depend on the phase. In the tree phase, the efficiency of its dynamically constructed, energy-aware inter-CH tree compared to EETB's potentially larger, all-node tree will determine their relative performance at greater distances. The energy-penalized tree construction in HEACT specifically aims to build a robust backbone even when CHs are distant from the BS.

**Overall Interpretation:**

The simulation results are expected to validate the design goals of the refined HEACT protocol. The introduction of the initial direct transmission phase, coupled with cluster size limits, is anticipated to demonstrably improve the network stability period (FDN) compared to both standard LEACH and previous HEACT iterations. The subsequent transition to an energy-aware, multi-hop inter-CH tree allows HEACT to maintain efficient operation over an extended period, achieving a network lifetime (NL) comparable to the dynamic tree-based EETB and significantly exceeding LEACH. While perhaps not matching the peak throughput of LEACH, HEACT's sustained data delivery over its longer lifespan is expected to result in competitive or superior overall data collection compared to EETB. The energy consumption plots should confirm more balanced energy depletion compared to LEACH and potentially smoother decay profiles than EETB. Therefore, HEACT presents a robust hybrid architecture that successfully balances the critical requirements of early-round stability, long-term operational longevity, and effective data gathering in energy-constrained WSNs.



# Conclusion

Energy efficiency remains a critical bottleneck limiting the deployment duration and practical utility of Wireless Sensor Networks. Addressing the challenges of unbalanced energy consumption and premature node failure requires innovative routing strategies that go beyond simple clustering or static tree structures. This paper introduced and comprehensively evaluated the **Hybrid Energy-Aware Cluster-Tree (HEACT)** protocol, a novel approach designed specifically to enhance both the initial stability period (FDN) and the overall operational lifetime (NL) of WSNs.

HEACT distinguishes itself through a synergistic combination of mechanisms:

* **Adaptive Clustering:** It employs periodic reconfiguration with Cluster Head (CH) selection intelligently weighted by both residual node energy and relative distance to the Base Station (BS), aiming for energetically capable and spatially distributed CHs. Furthermore, enforcing a maximum cluster size (HEACT\_MAX\_CLUSTER\_SIZE) directly addresses potential CH overload, promoting better local load balancing.
* **Dual-Mode Transmission Strategy:** A key innovation is the initial direct transmission phase (HEACT\_INITIAL\_DIRECT\_INTERVALS). By mandating direct CH-to-BS communication for the first few reconfiguration cycles, HEACT deliberately shields the network during its vulnerable setup period, preventing the immediate, high relay burden on nascent CHs that can lead to extremely low FDN in some hierarchical protocols.
* **Energy-Aware Inter-CH Tree:** Following the stabilization phase, HEACT transitions to a multi-hop routing paradigm between CHs. Crucially, the inter-CH tree is constructed using an energy-penalized cost metric (build\_inter\_cluster\_tree\_heact\_further\_revised), strongly favoring higher-energy CHs for relay roles and actively avoiding paths through weaker nodes. This promotes the longevity and robustness of the inter-cluster communication backbone.

Comparative simulations against the widely adopted LEACH protocol and the dynamic tree-based EETB algorithm demonstrate the effectiveness of the proposed HEACT architecture. The results, as indicated by performance metrics like FDN, NL, and packet delivery analysis, suggest that the strategic refinements within HEACT successfully address the problem of premature node death. **HEACT consistently achieved a significantly longer stability period (FDN)** compared to LEACH and was competitive with, potentially exceeding, the stability offered by EETB, validating the benefit of the initial direct transmission phase and load balancing features.

Furthermore, **HEACT demonstrated a substantially extended network lifetime (NL)**, matching the longevity of EETB by operating until the simulation limit, and vastly outperforming LEACH. This confirms the efficacy of combining clustering for local efficiency with robust, energy-aware multi-hop routing for inter-cluster communication in the long term. While LEACH exhibited higher peak throughput due to its direct transmission model, its rapid node depletion limited its total data delivery. **HEACT provided more sustained data delivery over its significantly longer operational lifespan**, resulting in a total throughput competitive with EETB and demonstrating its capability for long-term reliable data gathering.

In conclusion, the HEACT protocol presents a compelling hybrid routing solution for energy-constrained WSNs. By adaptively managing communication strategies across different network phases – prioritizing stability initially and transitioning to energy-aware multi-hop efficiency later – while incorporating explicit load balancing mechanisms like cluster size limits, HEACT effectively mitigates premature node failure and significantly extends network lifetime. The results underscore the advantages of carefully designed hybrid architectures that dynamically leverage the strengths of both clustering and multi-hop tree routing to optimize overall network performance and longevity.

**6. Future Scopes**

While the proposed HEACT protocol demonstrates significant potential for enhancing WSN lifetime and stability, several avenues exist for future research and refinement:

1. **Adaptation for Heterogeneous Networks:** The current evaluation assumes homogeneous nodes. Real-world deployments often involve heterogeneity, where nodes possess varying initial energy reserves, different sensing capabilities, or distinct communication ranges (e.g., regular nodes vs. more powerful relay nodes). Future work should investigate adapting HEACT's mechanisms, particularly the CH selection probability calculations and relay eligibility thresholds, to effectively leverage the capabilities of diverse nodes and further optimize energy distribution in heterogeneous environments. This could involve assigning higher probabilities or lower energy thresholds for more capable nodes to assume CH or relay roles.
2. **Incorporating Node and Sink Mobility:** The current HEACT assumes static nodes and a fixed BS. Introducing mobility, either for sensor nodes or the BS (sink), presents significant challenges. Node movement requires frequent updates to cluster memberships and potentially the inter-CH tree structure. Sink mobility necessitates efficient mechanisms for CHs to track the BS location and dynamically adjust routing paths. Future research could focus on developing lightweight localization updates, adaptive reconfiguration triggering based on mobility events (in addition to periodic intervals), and efficient hand-off mechanisms for nodes moving between clusters or CHs needing to find new paths to a mobile sink, balancing the energy cost of updates against the need for route validity.
3. **Advanced Cluster Head Selection Strategies:** The current HEACT employs an energy- and distance-weighted probabilistic CH selection. While effective, more sophisticated techniques could yield further improvements. Integrating a full implementation of distributed algorithms like HEED, which involves iterative competition based on multiple metrics (energy, communication cost), could lead to more optimal CH sets. Alternatively, employing fuzzy logic systems could allow for a more nuanced combination of factors (energy, distance, node degree, local density) to determine CH candidacy chance. Exploring metaheuristic optimizations (GA, PSO, ACO) specifically adapted for the HEACT setup phase could also be beneficial, although computational overhead must be carefully managed.
4. **Dynamic Reconfiguration Interval:** The current HEACT uses a fixed periodic interval (I\_recluster) for network reconfiguration. A fixed interval might be suboptimal – too frequent leads to excessive overhead, while too infrequent prevents timely adaptation to energy depletion or node failures. Future work could develop a dynamic reconfiguration interval mechanism. This could be triggered based on monitored network conditions, such as the rate of average energy depletion exceeding a certain threshold, the death of a significant number of nodes (particularly CHs or relay CHs), or significant changes detected in network connectivity or density.
5. **Traffic Load Adaptivity:** WSN applications often exhibit non-uniform data generation rates, leading to traffic hotspots. The current HEACT uses cluster size limits for basic load balancing. Future enhancements could involve making the protocol more adaptive to traffic load. This could include adjusting the C\_max\_size dynamically based on a CH's proximity to a high-traffic area, modifying the CH selection probability to favor nodes in lower-traffic regions, or adapting the inter-CH tree cost metric to consider the anticipated traffic load passing through potential relay CHs.
6. **Security Considerations:** As WSNs are often deployed in unattended environments, security is a critical concern. The hierarchical structure of HEACT, particularly the reliance on CHs and the inter-CH tree, introduces potential vulnerabilities. Attacks like sinkhole attacks (malicious node attracting traffic), selective forwarding (malicious relay dropping packets), or Sybil attacks (node creating multiple identities) could disrupt the network. Future work should investigate these vulnerabilities within the HEACT context and explore the integration of lightweight security mechanisms, such as identity verification for CHs, secure key management schemes, trust-based routing incorporated into parent selection, or intrusion detection systems tailored for hierarchical WSNs.
7. **Real-World Implementation and Validation:** Simulation results provide valuable insights but do not fully capture the complexities of real-world radio propagation, interference, hardware limitations, and synchronization issues. Implementing HEACT on a physical WSN testbed using platforms like Arduino, Raspberry Pi with LoRa/Zigbee modules, or commercial sensor motes would be crucial for validating its performance under realistic conditions and identifying practical implementation challenges.
8. **Broader Performance Benchmarking:** While this work compared HEACT against LEACH and EETB, a more extensive comparison against other state-of-the-art energy-efficient routing protocols (including recent LEACH variants, other tree-based protocols, geographic routing algorithms, and other hybrid approaches like EE-TLT or CER-CH) would provide a more comprehensive understanding of HEACT's relative strengths and weaknesses within the wider WSN routing landscape.

Addressing these future research directions can further enhance the robustness, efficiency, security, and practical applicability of the HEACT protocol for diverse WSN deployment scenarios.

# References

[1] W. B. Heinzelman, A. Chandrakasan, and H. Balakrishnan, ‘‘Energy efficient communication protocol for wireless microsensor networks,’’ in Proc. 33rd Annu. Hawaii Int. Conf. Syst. Sci., Jan. 2000, pp. 3005–3014.

[2] M. H. A. Hussain, B. Mokhtar, and M. R. M. Rizk, ‘‘A comparative survey on LEACH successors clustering algorithms for energy-efficient longevity WSNs,’’ Egyptian Informat. J., vol. 26, Jun. 2024, Art. no. 100477.

[3] J. Zhang and R. Yan, ‘‘Centralized energy-efficient clustering routing protocol for mobile nodes in wireless sensor networks,’’ IEEE Commun. Lett., vol. 23, no. 7, pp. 1215–1218, Jul. 2019.

[4] S. Mohapatra, P. K. Behera, P. K. Sahoo, S. K. Bisoy, K. L. Hui, and M. Sain, ‘‘Mobility induced multi-hop LEACH protocol in heterogeneous mobile network,’’ IEEE Access, vol. 10, pp. 132895–132907, 2022.

[5] H. Sun, D. Pan, D. Wang, and Z. Meng, ‘‘LPLL-LEACH: A study of a low-power and low-delay routing protocol based on LEACH,’’ Ad Hoc Netw., vol. 140, Mar. 2023, Art. no. 103064.

[6] L. Wu, A. Y. Dawod, and F. Miao, ‘‘Data transmission in wireless sensor networks based on ant colony optimization technique,’’ Appl. Sci., vol. 14, no. 12, p. 5273, Jun. 2024.

[7] X. Cheng, C. Xu, X. Liu, J. Li, and J. Zhang, ‘‘LEACH protocol optimization based on weighting strategy and the improved ant colony algorithm,’’ Frontiers Neurorobotics, vol. 16, Mar. 18, 2022, Art. no. 840332.

[8] A. M. K. Abdulzahra, A. K. M. Al-Qurabat, and S. A. Abdulzahra, ‘‘Optimizing energy consumption in WSN-based IoT using unequal clustering and sleep scheduling methods,’’ Internet Things, vol. 22, Jul. 2023, Art. no. 100765.

[9] A. Giri, S. Dutta, and S. Neogy, ‘‘An optimized fuzzy clustering algorithm for wireless sensor networks,’’ Wireless Pers. Commun., vol. 126, no. 3, pp. 2731–2751, Oct. 2022.

[10] Z. Wang, H. Ding, B. Li, L. Bao, Z. Yang, and Q. Liu, ‘‘Energy efficient cluster based routing protocol for WSN using firefly algorithm and ant colony optimization,’’ Wireless Pers. Commun., vol. 125, no. 3, pp. 2167–2200, Feb. 2022.

[11] H. Gunigari and S. Chitra, ‘‘Energy efficient networks using ant colony optimization with game theory clustering,’’ Intell. Autom. Soft Comput., vol. 35, no. 3, pp. 3557–3571, 2023.

[12] J. Amutha, S. Sharma, and S. K. Sharma, ‘‘An energy efficient cluster based hybrid optimization algorithm with static sink and mobile sink node for wireless sensor networks,’’ Exp. Syst. Appl., vol. 203, Oct. 2022, Art. no. 117334.

[13] L. Wang, J. Y. Huo, and H. M. M. Al-Neshmi, ‘‘An unequal clustering routing protocol for wireless sensor networks based on genetic algorithm,’’ J. Meas. Sci. Instrum., vol. 13, no. 3, pp. 329–344, Sep. 2022.

[14] S. Chaurasia, K. Kumar, and N. Kumar, ‘‘EEM-CRP: Energy-efficient meta-heuristic cluster-based routing protocol for WSNs,’’ IEEE Sensors J., vol. 23, no. 23, pp. 29679–29693, Dec. 2023.

[15] M. Abderrahim, H. Hakim, H. Boujemaa, and F. Touati, ‘‘A clustering routing based on Dijkstra algorithm for WSNs,’’ in Proc. 19th Int. Conf. Sci. Techn. Autom. Control Comput. Eng. (STA), Mar. 2019, pp. 605–610.

[16] T. C. Hung, D. T. Ngoc, P. T. The, L. N. Hieu, L. N. T. Huynh, and L. D. Tam, ‘‘A moving direction proposal to save energy consumption for mobile sink in wireless sensor network,’’ in Proc. 21st Int. Conf. Adv. Commun. Technol. (ICACT), Feb. 2019, pp. 107–110.

[17] V.-H. Nguyen and N. D. Tan, ‘‘Voronoi diagrams and tree structures in HRP-EE: Enhancing IoT network lifespan with WSNs,’’ Ad Hoc Netw., vol. 161, Aug. 2024, Art. no. 103518.

[18] Y.-D. Yao, H.-C. Li, Z.-B. Zeng, C. Wang, and Y.-Q. Zhang, ‘‘Clustering routing protocol based on tuna swarm optimization and fuzzy control theory in wireless sensor networks,’’ IEEE Sensors J., vol. 24, no. 10, pp. 17102–17115, May 2024.

[19] Q. Wei, K. Bai, and L. Zhou, ‘‘An improved approach for wireless sensor networks with mobile sink using dynamic minimum spanning tree,’’ IEEE Sensors J., vol. 22, no. 11, pp. 10918–10930, Jun. 2022.

[20] S. Lindsey and C. S. Raghavendra, ‘‘PEGASIS: Power-efficient gathering in sensor information systems,’’ in Proc. IEEE Aerosp. Conf., Mar. 2002, pp. 1125–1130.

[21] E. F. Ahmed Elsmany, M. A. Omar, T.-C. Wan, and A. A. Altahir, ‘‘EESRA: Energy efficient scalable routing algorithm for wireless sensor networks,’’ IEEE Access, vol. 7, pp. 96974–96983, 2019.

[22] A. N. Ali, W. Mardini, Q. Q. Abuein, and M. Khair, ‘‘A hybrid and efficient algorithm for routing in wireless sensor networks,’’ in Proc. 11th Int. Conf. Inf. Commun. Syst. (ICICS), Apr. 2020, pp. 355–360.

[23] D. Sharma and G. Singh Tomar, ‘‘Enhance PEGASIS algorithm for increasing the life time of wireless sensor network,’’ Mater. Today, Proc., vol. 29, pp. 372–380, Jan. 2020.

[24] S. Sadhana, E. Sivaraman, and D. Daniel, ‘‘Enhanced energy efficient routing for wireless sensor network using extended power efficient gathering in sensor information systems (E-PEGASIS) protocol,’’ in Proc. 3rd SSCI, Jan. 2021, pp. 89–101.

[25] P. Chugh, M. Gupta, S. Indu, G. Chaudhary, M. Khari, and V. Shanmuganathan, ‘‘Advanced energy efficient pegasis based routing protocol for IoT applications,’’ Microprocess. Microsyst., vol. 103, Nov. 2023, Art. no. 104727.

[26] N. D. Tan and V.-H. Nguyen, ‘‘EE-TLT: Energy-efficient routing protocol using two-level tree-based clustering in wireless sensor network,’’ J. Commun. Netw., vol. 25, no. 6, pp. 734–749, Dec. 2023.

[27] M. Micheletti, L. Mostarda, and A. Navarra, ‘‘CER-CH: Combining election and routing amongst cluster heads in heterogeneous WSNs,’’ IEEE Access, vol. 7, pp. 125481–125493, 2019.

[28] N. Chen and R. Wen, ‘‘Energy Efficient Tree-Based Routing Algorithm for Wireless Sensor Networks,’’ IEEE Access, vol. 13, pp. 20149–20159, Jan. 2025. doi: 10.1109/ACCESS.2025.3535229.