Design and Evaluation of a Hybrid Wireless Sensor Network Combining Cluster-Tree and Mesh Routing

Nicholas Palmer
Dept. of Electrical and Computer Engineering
San Diego State University
San Diego, CA, USA
npalmer2267@sdsu.edu

***Abstract*—Wireless sensor networks (WSNs) are self-organizing networks of sensor nodes deployed to monitor the environment. These networks are typically battery-powered with limited processing and storage capabilities, requiring energy-efficient communication protocols and resilience to node and link failures. To address these challenges, we propose a hybrid WSN that combines cluster-tree and mesh routing to enhance network performance. We also implement router-bridge nodes to improve energy efficiency between cluster heads. We evaluate our hybrid WSN using a Python simulation tool and compare it with existing WSNs.**

***Index Terms*—Wireless Sensor Network, AODV, Cluster tree**

I. INTRODUCTION

Wireless sensor networks (WSNs) are used in a variety of applications, such as environmental monitoring, industrial automation, healthcare, and increasingly, in consumer devices. The necessity for ease of deployment and lower cost hardware have made WSNs popular in recent years, but they also pose significant challenges in terms of self-organization, energy efficiency, and robustness.

A. Self-Organization

Self-organization is a crucial advantage of WSNs, allowing deployments in remote or inaccessible locations, making it impractical to pre-configure infrastructure. Self-organization allows nodes to form a network topology and select routing paths dynamically based on the network conditions and the environment. Building an optimal topology presents many challenges given the dynamic nature of the environment, limited computational resources, and energy resources of most low-cost hardware. Most networks use one of two topologies: a hierarchical cluster-tree or a mesh topology.

- **Cluster-Tree**
  - Hierarchical network with a single root node and multiple cluster-heads
  - Cluster-heads are responsible for routing data to and from their cluster, only they must maintain routing information for their child members and child networks below them
  - Centralized routing allows for greater energy efficiency as the routing can be optimized for the entire network
- **Mesh**
  - Peer-to-peer network with no central nodes or defined hierarchy
  - Each node is responsible for routing data to and from its neighbors, maintaining routing information for all its n-hop neighbors
  - Decentralized routing allows for greater robustness to node and link failures as the routing can adapt on-demand to changes in the network topology

Despite these advantages, both topologies have their limitations. Cluster-tree topologies can be fragile, as the failure of a cluster-head can isolate an entire cluster from the network causing a cascading failure. While mesh topologies can incur significant overhead due to the need for each node to maintain routing information for all its neighbors, leading to increased overhead and energy use. These shortcomings motivate the need for a hybrid approach that combines the strengths of both topologies.

B. Hybrid WSNs

To address these limitations, hybrid WSNs that incorporate both cluster-tree and mesh routing have been proposed. In these approaches, nodes are organized into a cluster-tree topology, but they can also use mesh links to route data between clusters. This allows for greater robustness to node failures, as nodes can use mesh links to route bypass failed cluster-heads. Additionally, the cluster-tree structure allows for more efficient routing within clusters, reducing the need for each node to maintain routing information for all its neighbors.

C. Contributions

In this paper, we present the design and evaluation of a hybrid WSN that combines cluster-tree and mesh routing. We first describe the design of the hybrid topology formation and routing protocol, including the use of router-bridge nodes to improve energy efficiency between cluster heads. We then implement the protocol via a Python simulation and evaluate its performance in terms of average join time, connectivity, packet delivery, and energy use, under node and link losses. Finally, we compare the performance of our hybrid WSN with existing WSNs.

II. RELATED WORK

We briefly review prior works on cluster-tree and mesh routing protocols in WSNs, as well as existing hybrid approaches.

A. Cluster-Tree Protocols

B. Mesh Routing Protocols

C. Hybrid Approaches

D. Summary

- **Summarize relevant literature:**
  - Describe existing cluster-tree protocols (e.g., ZigBee-like approaches).
  - Describe mesh routing protocols in sensor/IoT networks.
  - Mention any known hybrid or hierarchical approaches that combine tree and mesh ideas.
- **Highlight limitations of prior work:**
  - Explain what existing solutions do not address (e.g., limited robustness to node failures, no analysis of join time, no energy-aware design).
- **Position the current work:**
  - Clearly state how the proposed hybrid design differs from and/or extends prior work and what aspects it focuses on (e.g., join time, connectivity under failures, impact of energy depletion).

III. SYSTEM MODEL AND ASSUMPTIONS

We describe the system model and assumptions used in our design and evaluation.

A. Network Model

We consider a network consisting of N sensor nodes deployed uniformly at random in a defined deployment area. The network includes one designated root node that initiates network formation, and the remaining nodes self-organize into a hybrid cluster-tree and mesh topology. Nodes communicate via a single radio channel with a fixed transmission range, and we assume static node placement throughout the simulation.

B. Traffic Model

To simulate a typical WSN use-case, we consider periodic data collection where sensor nodes report measurements to the sink. Once a node has joined the network and is in the REGISTERED, CLUSTER_HEAD, or ROUTER state, we begin generating traffic. Each node generates periodic data packets destined for the root node (sink), simulating a many-to-one traffic pattern typical of WSNs. The packet generation interval is set to a fixed value of 0.1 seconds, resulting in a constant data rate per node.

C. Energy Model

Each node is initialized with a finite energy budget and consumes energy based on its radio state. We model energy consumption for transmit (TX) and receive (RX) states, with power draw values derived from typical CC2420 radio specifications. Nodes that deplete their energy budget become permanently disabled and cease all communication.

- **Network model:**
  - Number and types of nodes (e.g., one sink/root, cluster heads, regular sensor nodes).
  - Radio abstraction (single channel, common transmission range, interference model at a high level).
  - Whether nodes are static or mobile (typically static).
- **Traffic model:**
  - What traffic is generated (e.g., periodic sensing towards the sink, control only, or both).
  - Which nodes generate data and at what rate.
- **Energy model:**
  - Briefly describe the radio states (TX/RX/IDLE/SLEEP) and that energy is consumed based on time spent in each state.
  - State that nodes start with a finite energy budget and shut down when energy reaches zero.
- **Additional assumptions and scope:**
  - Assumptions on initial energy equality, absence of obstacles, simplified propagation or MAC model, etc.
  - What is deliberately *not* modeled (e.g., no detailed interference modeling, no mobility).

IV. HYBRID CLUSTER-TREE + MESH NETWORK DESIGN

This section describes the proposed protocol in detail.

A. Overall Architecture

We propose a hybrid cluster-tree mesh topology where nodes are organized into a hierarchical cluster-tree structure, but they can also use mesh links to route data between clusters. This hybrid approach leverages the strengths of both cluster-tree and mesh routing.

Our architecture includes the following Node types, CLUSTER$_{M}EMBER, CLUSTER_{H}EAD, ROUTER, GATEWAY, c$

- Describe the roles of different nodes:
  - Sink/root node.
  - Cluster heads (parents).
  - Regular/leaf nodes.
  - Local neighborhood and mesh neighbors
- Explain how the cluster-tree and mesh components coexist:
  - Cluster-tree provides hierarchical structure and basic connectivity.
  - Discuss the formation of cluster tree, the role of routers, cluster overlaps

figures/wsn_arch.png

Fig. 1: A high-level architecture of the Hybrid Cluster Tree Mesh Network

- Mesh provides improved robustness in the local neighborhood.

A high-level figure illustrating the architecture (tree backbone with some cross-layer mesh links) if given in Figure **??**.

B. Energy Consumption Model

Each node is equipped with a CC2420-like radio transceiver. We model the energy consumption of the radio using four main states: transmit (TX), receive (RX), idle listening (IDLE), and sleep (SLEEP). Each state s is associated with a constant power draw P_s, derived from the typical CC2420 current consumption and supply voltage.

Let $E_i(t)$ denote the remaining energy of node i at time t. Over a time interval $[t, t+\Delta t]$, the energy is updated according to

$$E_i(t+\Delta t) = E_i(t) - \sum_{s\in\{\mathrm{TX,RX,I,S}\}} P_s\, T_{i,s}(t, t+\Delta t). \quad (1)$$

where $T_{i,s}(t, t+\Delta t)$ is the time spent by node i in state s during $[t, t+\Delta t]$.

Your simulator tracks the radio state (you have only implemented RX and TX states I believe. Also mention the other radio states of each node and discuss how your simulator integrates the corresponding energy consumption over time.

B. Network Formation Procedure

Describe, step by step, how the network forms from an initial empty state:

- **Initial conditions:**
  - Describe how the root starts the network and begins advertising its presence.
- **Discovery and joining:**
  - How new nodes detect neighbors or beacons.
  - How a node chooses a parent or cluster head (e.g., based on signal strength, hop count, or node ID).
  - The message exchange for joining .
- **Cluster formation rules:**
  - Conditions for becoming a cluster head versus joining as a child.
  - Any limits on the number of children per cluster head.
- **Timing and re-tries:**
  - When and how often nodes attempt to join or re-join if the first attempt fails.

C. Routing Strategy

This subsection explains how packets are forwarded once the network is formed:

- **Intra-cluster routing:**
  - How children send data to their cluster head and how the cluster head forwards data within the cluster (if needed).
- **Inter-cluster/global routing:**
  - Default tree-based paths towards the sink (parent-to-parent).
  - When and how mesh links are used .
- **Route selection:**
  - The metric used to choose routes (e.g., hop count, link quality, or combined metrics).
  - What routing information each node stores (parent, cluster ID, mesh neighbors, distance to sink, etc.).

D. Self-Organization and Maintenance

Describe how the network adapts over time:

- **Handling node failures:**
  - How children detect that their parent or cluster head has failed (e.g., missed heartbeats or timeouts).
  - How nodes attempt to re-join the network or select a new parent.
- **Handling changing link quality:**
  - How link quality is monitored and how bad links lead to route changes or parent switches.
- **Mesh adaptation:**
  - Rules for creating and removing mesh links between nodes.
  - Any periodic neighbor discovery or link probing.
  - The role of routers to provide communication between clusters
- **Energy-related aspects (if applicable):**
  - Any mechanisms to reduce energy usage (e.g., sleep cycles, rotation of cluster-head roles).
  - Adaptive Tx Power to reduce energy consumption
  - Recovery from node failures due to nodes running out of energy – any proactive algorithmic solution you can offer

V. SIMULATION PLATFORM AND EXPERIMENTAL SETUP

A. Simulation Platform

Describe where and how the protocol is implemented:

- Briefly explain the main modules:
  - Node and radio abstraction.
  - Topology and neighbor management.
  - Protocol logic (formation, routing, maintenance).
  - Logging and statistics collection.
- Note any simplifications:

B. Scenario Configuration

This subsection defines the experimental scenarios in detail:

- **Topology:**
  - Number of nodes (e.g., 25, 50, 100, 200).
  - Deployment area size (e.g., $100\,\mathrm{m} \times 100\,\mathrm{m}$).
  - Node placement (random uniform, grid, etc.).
- **Radio parameters:**
  - Communication range or path-loss model. Simplify the discussion here , radio access is based on the distance between nodes.
  - Assumptions about interference (we did not have any interference, what else).
- **Traffic configuration:**
  - Packet generation rate (low/medium/high traffic).
  - Packet size and destinations (root or any destination).
- **Energy configuration:**
  - Initial energy per node.
  - Power levels used in the energy model (TX/RX/IDLE/SLEEP). [You are spending energy only when you receive and Transmit. Ideally you should have 4 states at least]
- **Failure scenarios (if studied):**
  - Patterns and percentages of node failures (random vs targeted).
  - Link loss or packet loss probabilities, the impact on the network formation and run.

C. Metrics

Define clearly what do you measure and how: Add a plot where it is necessary to show experimental results.

- **Join-related metrics:**
  - Per-node join time and average join time.
  - Join success rate (fraction of nodes that join within a time limit).
- **Connectivity metrics:**
  - Fraction of nodes with a valid route to the sink.
  - Average hop count from connected nodes to the sink.
- **Routing performance:**
  - Packet delivery ratio (PDR) under various link and node conditions.
  - End-to-end delay (mean, possibly percentiles) under various link and node conditions.

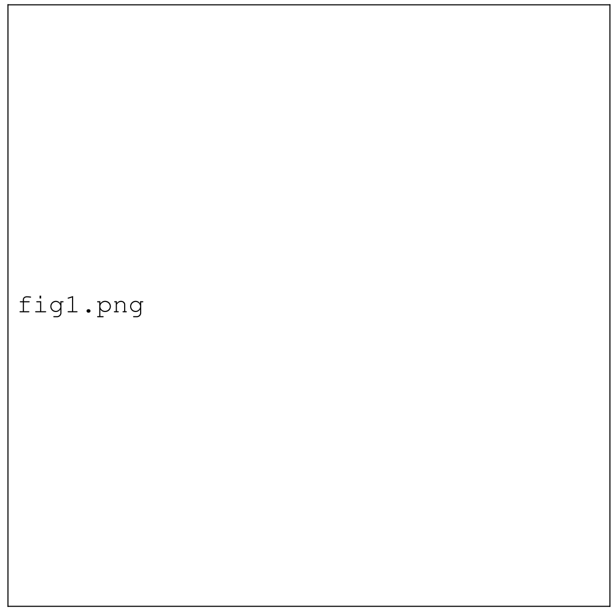

Fig. 2: Average join time versus network size.

- **Energy metrics:** At the beginning of each experiment, all nodes are initialized with a finite energy budget $E_i(0) = E_0$. When the remaining energy of node i reaches zero or becomes negative, i.e., $E_i(t) \leq 0$, the node is permanently disabled. A disabled node turns off its radio and ceases all communication and protocol participation. As a result, energy depletion induces node losses that may fragment the topology and degrade connectivity.
  In addition to per-node energy, the simulator records the cumulative time that each node spends in TX, RX, IDLE, and SLEEP. This will enable a fine-grained analysis of the energy cost incurred by different network roles (e.g., cluster heads, mesh forwarders, routers, and leaf nodes).
  - Average remaining energy over time.
  - Node lifetime (time until individual nodes die).
  - Network lifetime (time until a threshold fraction of nodes is no longer connected).

D. Experiment Design

Finally, you should explain how experiments are organized:

- **Baseline experiments:**
  - Vary network size (e.g., 25–200 nodes) under normal conditions to study join time, routing performance, and basic scalability.
- **Failure experiments:**
  - Introduce node failures (random and/or targeted) and measure the effect on connectivity, routing, and join/re-join behavior.
  - Simulate a scenario where N nodes are killed after the simulation converged to a topology and nodes stay off for some time specified. Nodes are re-activated after being off for some time. Discuss

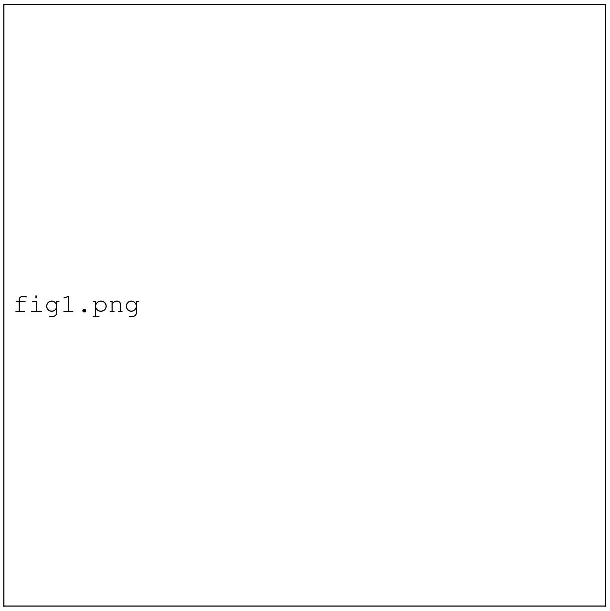

Fig. 3: Nodes killed versus number of nodes disconnected

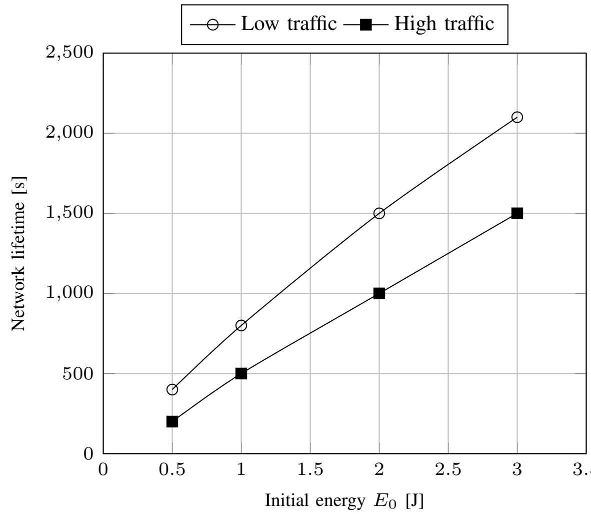

Fig. 4: Network lifetime as a function of the initial energy budget E_0 per node, for different traffic loads. Network lifetime is defined as the time until fewer than 80% of the nodes remain connected to the sink.

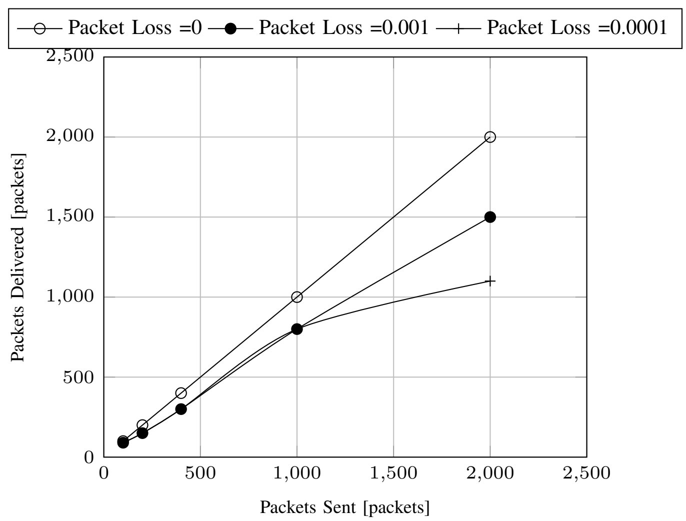

Fig. 5: Network lifetime as a function of the initial energy budget E_0 per node, for different traffic loads. Network lifetime is defined as the time until fewer than 80% of the nodes remain connected to the sink.

  - topology changes after the nodes are deactivated and reactivated.
  - Introduce packet/link loss and analyze PDR and delay.
- **Energy experiments:**
  - Vary initial energy budgets and traffic loads to study the trade-off between throughput and network lifetime.
- **Energy-Aware Experiment Design** We design several experiments to characterize the impact of energy consumption on network connectivity and performance.

  1) Network Lifetime vs. Initial Energy: In the first experiment set, we investigate how the initial energy budget affects the overall network lifetime. The number of nodes, node placement, and traffic pattern are kept constant, while the initial energy E_0 is varied across multiple scenarios (e.g., low, medium, and high battery capacity). For each scenario, we record:
  - the time of the first node death,
  - the time at which 10%, 50%, and 90% of the nodes have depleted their energy, and
  - the time at which the fraction of nodes with a valid route to the sink falls below a given connectivity threshold (e.g., 80%).

  We refer to the latter as the *network lifetime*, since it captures the time interval during which the network remains operational from the application point of view.

  2) Role-Based Energy Depletion: In the second experiment set, we analyze the energy consumption of different node roles in the hybrid cluster-tree and mesh topology. Cluster heads and mesh forwarders are expected to relay a larger fraction of the traffic and to transmit more control packets than leaf nodes. To quantify this effect, we log for each node its role, time of death (if any), and total time spent in each radio state. We then compare:
  - the average lifetime of cluster heads and mesh forwarders against that of leaf nodes, and
  - the average energy breakdown (TX/RX/IDLE/SLEEP) per role.

  This experiment highlights whether critical nodes (e.g., cluster heads) tend to deplete their batteries prematurely and how their failure impacts the remaining topology.

  3) Impact of Traffic Load on Energy Consumption: In the third experiment set, we study the trade-off between

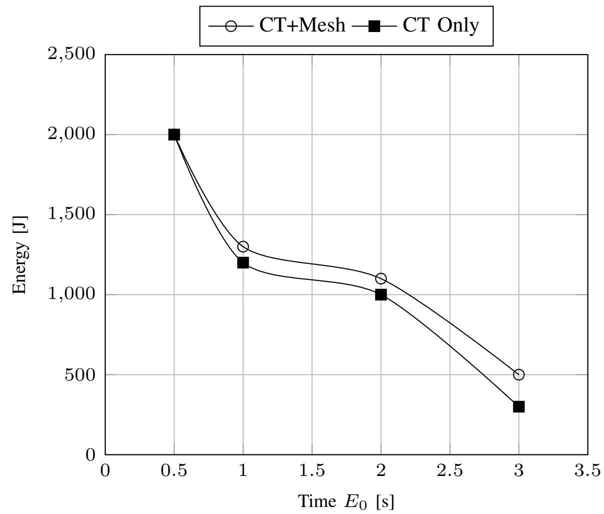

Fig. 6: Network lifetime as a function of the initial energy budget E_0 per node, for different traffic loads. Network lifetime is defined as the time until fewer than 80% of the nodes remain connected to the sink.

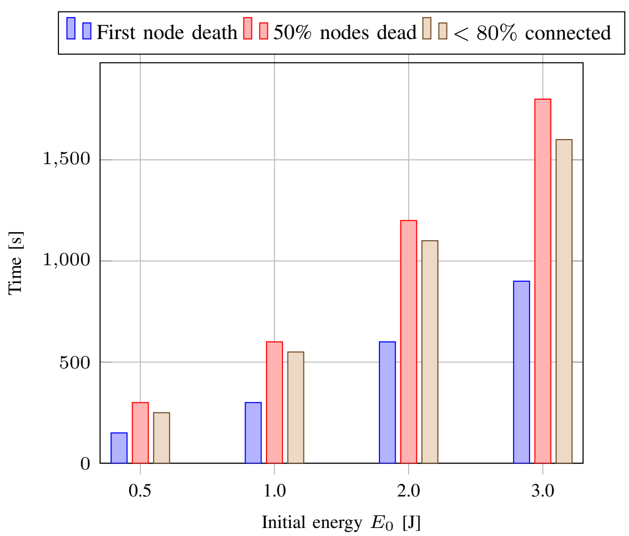

Fig. 7: Impact of the initial energy budget E_0 on different lifetime metrics: time of first node death, time at which 50% of the nodes are dead, and time at which fewer than 80% of the nodes remain connected to the sink.

traffic load and network lifetime. The topology and initial energy E_0 are kept fixed, while the application data rate is varied (e.g., low, medium, and high packet generation rates per node). For each traffic level, we monitor:

- the average remaining energy over time,
- the number and fraction of alive nodes,
- the number and fraction of nodes with a route to the sink, and
- the packet delivery ratio (PDR) computed over sliding time windows.

By comparing these metrics across traffic configurations, we evaluate how increased load accelerates energy depletion and reduces the time for which the network can sustain a target connectivity or reliability level.

4) Energy-Aware vs. Baseline Configuration (Optional): If applicable, we also consider an energy-aware configuration of the hybrid protocol, in which the cluster-head role is periodically rotated or the maximum number of children per cluster head is limited. We compare this configuration against a baseline with fixed cluster heads. The comparison focuses on node lifetime distributions, the fraction of prematurely dead cluster heads, and the resulting network lifetime.

- **Repetitions and statistics:**
  - Indicate how many runs are performed per scenario (e.g., 10 or 20 runs with different random seeds).
  - Explain how averages and confidence intervals are computed.

C. Discussion

D. Impact of Energy Consumption on Network Connectivity

This section presents the impact of energy consumption and energy-driven node failures on the hybrid cluster-tree and mesh network.

1) Evolution of Alive and Connected Nodes: Fig.**??** shows the fraction of alive nodes and the fraction of nodes that maintain a valid route to the sink as a function of time, for different initial energy budgets E_0. In all cases, the fraction of alive nodes decreases gradually as nodes deplete their batteries. However, the fraction of connected nodes typically decreases more abruptly when critical nodes (e.g., cluster heads or mesh forwarders) fail. For small E_0, the network rapidly loses connectivity after the first few cluster-head failures. Larger energy budgets postpone these critical failures and increase the time interval during which more than 80% of the nodes remain connected.

2) Node Lifetime Distribution and Role Effects: Fig. **??** depicts " Cluster heads exhibit significantly shorter lifetimes, confirming that their higher forwarding and control responsibilities translate into faster energy depletion. As a result, the first network partitions are typically triggered by the death of cluster heads. Leaf nodes, in contrast, often retain a substantial amount of energy when the network as a whole is already disconnected from the sink. This imbalance suggests that energy-aware role assignment or rotation mechanisms could substantially extend network lifetime.

3) Performance Degradation Due to Energy Depletion: Fig. **??** shows the packet delivery ratio (PDR) over time. We observe that PDR remains close to its nominal value while the majority of nodes are both alive and connected. Once a sufficient number of cluster heads and relay nodes have died, PDR drops sharply even though a non-negligible fraction of nodes are still powered. This indicates that from the application perspective, the effective end of the network occurs before the physical exhaustion of all nodes, reinforcing the importance of connectivity-based lifetime metrics.

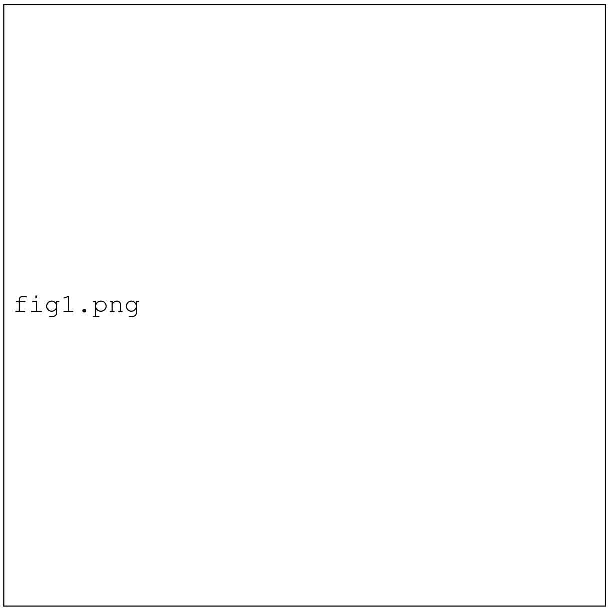

Fig. 8: Packet delivery ratio (PDR) over time.

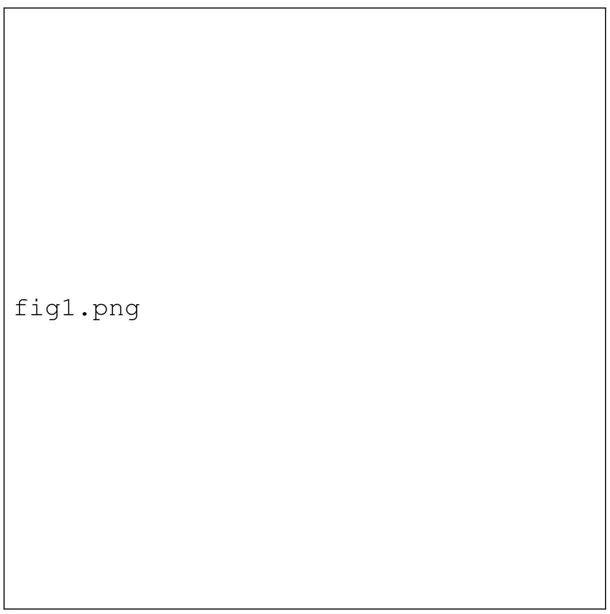

Fig. 9: Average remaining energy over time for different traffic loads (e.g., low, medium, and high packet generation rates per node). Higher traffic accelerates energy depletion and reduces the available energy budget of the nodes.

4) Impact of Traffic Load on Lifetime: Fig. **??** compares the average remaining energy and connectivity over time for different traffic loads. Higher packet generation rates lead to a faster reduction in the average remaining energy and a correspondingly earlier loss of connectivity. For high traffic, the network lifetime (time until fewer than 80% of nodes remain connected) can be reduced by more than half compared to the low-traffic scenario. This highlights a fundamental trade-off between throughput and lifetime in the proposed hybrid

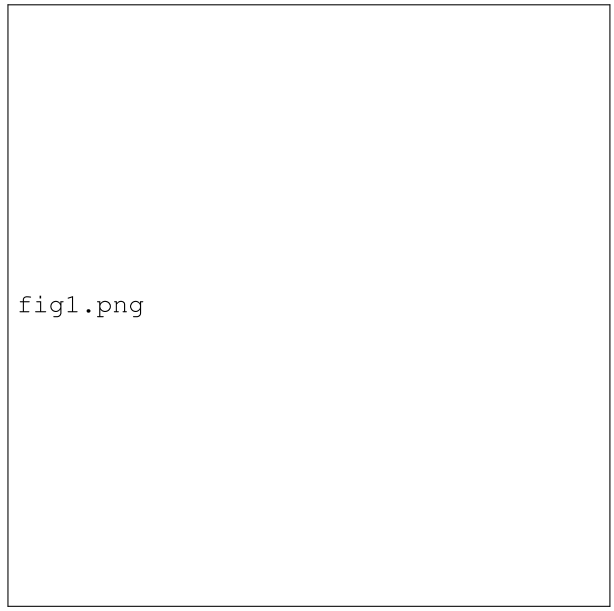

Fig. 10: Fraction of nodes that remain connected to the sink over time for different traffic loads. Under high traffic, connectivity drops significantly earlier, reducing the effective network lifetime.

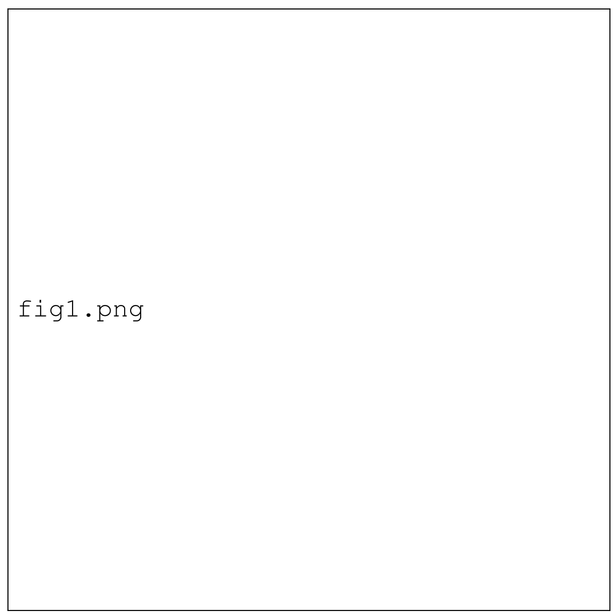

Fig. 11: The cumulative distribution function (CDF) of node lifetimes, separately for cluster heads and leaf nodes.

architecture.

5) Comparison with Energy-Aware Configuration (Optional): If an energy-aware configuration is enabled, in which cluster-head roles are periodically rotated or constrained by a maximum number of children, the lifetime results improve. In our simulations, the energy-aware configuration shifts the node lifetime CDF for cluster heads closer to that of leaf nodes and delays the onset of major connectivity losses. Consequently, the time at which PDR drops below the target threshold increases, demonstrating that simple energy-balancing mechanisms can significantly enhance the robustness of the hybrid cluster-tree and mesh network.

I. Conclusion and Future Work

References

[1] J. M. Wooten, D. M. Bevly, and J. Y. Hung, “Piezoelectric polymer-based collision detection sensor for robotic applications,” *Electronics*, vol. 4, pp. 204–220, 2015.

[2] X. Chao *et al.*, “Bioinspired twist-hyperbolic metamaterial for impact buffering and self-powered real-time sensing in uavs,” *Science Advances*, vol. 11, p. eadw6179, 2025.

[3] Q. He, Y. Zeng, L. Jiang *et al.*, “Growing recyclable and healable piezoelectric composites in 3d printed bioinspired structure for protective wearable sensor,” *Nature Communications*, vol. 14, p. 6477, 2023.

[4] L. Ambroziak, D. Ołdziej, and A. Koszewnik, “Multirotor motor failure detection with piezo sensor,” *Sensors*, vol. 23, no. 2, p. 1048, 2023.

[5] L. d. S. Goncalves, R. M. L. Pereira, R. S. Tyszler, M. C. A. M. Morais, and C. R. H. Barbosa, “Application of pvdf transducers for piezoelectric energy harvesting in unmanned aerial vehicles,” *Energies*, vol. 18, p. 4759, 2025.

[6] M. Akbar, A. R. Firdaus, S. Wibowo, and N. Wirawan, “Piezoelectric energy harvesting from a fixed-wing unmanned aerial vehicle,” in *2019 2nd International Conference on Applied Engineering (ICAE)*, 2019, pp. 1–4.

[7] A. Merei, H. Mcheick, A. Ghaddar, and D. Rebaine, “A survey on obstacle detection and avoidance methods for uavs,” *Drones*, vol. 9, p. 203, 2025.

[8] J. N. Yasin *et al.*, “Unmanned aerial vehicles (uavs): Collision avoidance systems and approaches,” *IEEE Access*, vol. 8, pp. 105 139–105 155, 2020.

[9] M. R. Rezaee, N. A. W. A. Hamid, M. Hussin, and Z. A. Zukarnain, “Comprehensive review of drones collision avoidance schemes: Challenges and open issues,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 25, no. 7, pp. 6397–6426, 2024.

[10] T. Feng and M. H. F. Aliabadi, “Structural integrity assessment of composite plates with embedded pzt transducers for structural health monitoring,” *Materials*, vol. 14, no. 20, p. 6148, 2021.

[11] R. Wang, Z. Guo, Y. Chen, X. Wang, and B. M. Chen, “Air bumper: A collision detection and reaction framework for autonomous mav navigation,” pp. 15 735–15 741, 2024.

[12] J. Verbeke, S. Debruyne, P. Sas, D. Moens, and A. VanDeWalle, “Vibration analysis of a uav multirotor frame,” 2016-09-20.

[13] L. E. Tuck, C. Samson, J. Laliberté, and M. Cunningham, “Magnetic interference mapping of four types of unmanned aircraft systems intended for aeromagnetic surveying,” *Geoscientific Instrumentation, Methods and Data Systems*, vol. 10, no. 1, pp. 101–112, 2021. [Online]. Available: https://gi.copernicus.org/articles/10/101/2021/

[14] E. Aldao, L. M. González-de Santos, and H. González-Jorge, “Lidar based detect and avoid system for uav navigation in uam corridors,” *Drones*, vol. 6, no. 8, 2022. [Online]. Available: https://www.mdpi.com/2504-446X/6/8/185

[15] C. Ma, Y. Fu, D. Wang, R. Guo, X. Zhao, and J. Fang, “Yolo-uav: Object detection method of unmanned aerial vehicle imagery based on efficient multi-scale feature fusion,” *IEEE Access*, vol. 11, pp. 126 857–126 878, 2023.

[16] R. Wang, Z. Guo, Y. Chen, X. Wang, and B. M. Chen, “Air bumper: A collision detection and reaction framework for autonomous mav navigation,” 2023. [Online]. Available: https://arxiv.org/abs/2307.06101

[17] M. Guanglei and P. Haibing, “The application of ultrasonic sensor in the obstacle avoidance of quad-rotor uav,” in *2016 IEEE Chinese Guidance, Navigation and Control Conference (CGNCC)*, 2016, pp. 976–981.

[18] N. Gageik, P. Benz, and S. Montenegro, “Obstacle detection and collision avoidance for a uav with complementary low-cost sensors,” *IEEE Access*, vol. 3, pp. 599–609, 2015.

[19] P. Hügler, F. Roos, M. Schartel, M. Geiger, and C. Waldschmidt, “Radar taking off: New capabilities for uavs,” *IEEE Microwave Magazine*, vol. 19, no. 7, pp. 43–53, 2018.

[20] A. Al-Kaff, F. García, D. Martín, A. De La Escalera, and J. M. Armingol, “Obstacle detection and avoidance system based on monocular camera and size expansion algorithm for uavs,” *Sensors*, vol. 17, no. 5, 2017. [Online]. Available: https://www.mdpi.com/1424-8220/17/5/1061

[21] Makeblock, “mbot mega,” 2021. [Online]. Available: https://support.makeblock.com/hc/en-us/sections/1500001152162-mBot-Mega

[22] Digilent, “Analog disvoery 3,” 2023. [Online]. Available: https://digilent.com/shop/analog-discovery-3/