Journal Publication of International Research for Engineering and Management (JOIREM)
Volume: 04 Issue: 5 | May-2026
ISSN (O) 3107-6696

Intelligent Routing to Optimize Energy Consumption in Wireless Sensor Networks: A Distributed Approach with Reinforcement Learning

Dias Abdrakhmanov

IITU, almaty, Kazakhstan

---***---

Abstract - Energy depletion heterogeneity — not total energy consumption — determines when wireless sensor networks fail: nodes at traffic hotspots die while peripheral nodes retain 60–80% residual charge, collapsing coverage prematurely. Existing clustering protocols such as LEACH and HEED address this by optimizing cluster head selection globally, yet their episodic reformation cycles (every 20 rounds) leave them structurally blind to topology changes between updates.

No published distributed protocol simultaneously optimizes residual energy, transmission distance, and node load using only one-hop information while adapting those weights online through reinforcement learning.

This paper presents DEAR (Distributed Energy-Aware Routing), a protocol that makes per-transmission forwarding decisions via a three-term cost function $C(n) = \alpha \cdot E\_r(n) + \beta \cdot d(n) + \gamma \cdot L(n)$, requiring no global state. DEAR-RL extends this by embedding Q-learning at cluster heads to update α, β, γ each round based on observed network conditions. Both protocols were evaluated in MATLAB R2023a across six competitors, four network scales ($N \in \{50, 100, 200, 300\}$), and five environmental scenarios, with 50 independent runs per configuration and two-sample t-tests for significance verification.

DEAR raises HND from 938 to 1,127 rounds versus HEED at N=100 in baseline conditions — a 20.1% gain ($t=8.4$, $p<0.001$, $d=1.31$) — while reducing energy balance variance from $\sigma\_E=0.041$ J to $\sigma\_E=0.023$ J, a 44% improvement in distribution uniformity. Under node mobility, the advantage grows to 35.2% (1,014 vs 750 rounds), confirming that continuous local adaptation outperforms periodic global reformation under dynamic conditions. DEAR-RL adds a further 8.2% over DEAR in baseline (1,219 vs 1,127 rounds, $p<0.001$) and 13.2% under mixed interference-mobility conditions, at 1.1% computational overhead.

Local, continuous, adaptive routing eliminates the reformation-lag bottleneck that constrains globally-informed protocols, delivering measurable lifetime gains deployable on resource-constrained hardware without infrastructure modification.

Keywords: wireless sensor networks; energy-efficient routing; reinforcement learning; network lifetime; distributed algorithm; clustering protocol; Q-learning; energy balance

1. Introduction

Consider a 300-node soil-moisture monitoring network deployed across 40 hectares of wheat farmland in northern Kazakhstan. Over a 180-day growing season, each node must deliver 250,000 data packets to a central base station while running on a single 1J battery. By day 47, 18 nodes adjacent to the base station have exhausted their energy — not because the network consumed too much overall, but because those nodes relayed traffic for their neighbors in addition to their own. The remaining 282 nodes retain an average 0.71J of residual charge. The network is topologically alive yet functionally dead: coverage gaps over 23% of the monitored area make irrigation decisions unreliable. This failure mode — early node death driven by load imbalance rather than total energy depletion — is the central unsolved problem in wireless sensor network (WSN) routing. With IoT deployments projected to exceed 29 billion connected devices by 2030, and WSNs forming the data-acquisition backbone of precision agriculture, structural health monitoring, and environmental sensing, solving this problem at scale is an industrial necessity.

Existing clustering protocols address energy efficiency through hierarchical organization, but each fails in a distinct and quantifiable way. LEACH [1] selects cluster heads via a uniform random process with target probability p=0.05, making no reference to a node's residual energy at selection time. By round 500 in a 100-node network, LEACH produces inter-node energy variance of $\sigma\_E=0.048$ J — nodes drawn repeatedly as cluster heads deplete to 0.1J while non-selected nodes hold 0.7J. The network reaches Half-Node Death (HND) at round 681, with 49 nodes still holding more than 40% initial charge. HEED [2] improves on this by weighting cluster head probability by residual energy, achieving $\sigma\_E=0.041$ J and HND=938 rounds. Yet HEED's fundamental constraint is temporal: cluster reformation occurs every 20 rounds, meaning any topology change — a node entering low-battery state, a burst of correlated traffic, a mobile obstacle creating a shadow zone — remains invisible to the routing layer until the next reformation cycle. Under moderate node mobility (3 m/s random waypoint), HEED's HND drops to 750 rounds, a 20% regression from its own baseline. The protocol optimizes the network it measured 20 rounds ago, not the network that exists now.

No published distributed protocol simultaneously satisfies three properties that practical deployments require: (a) routing decisions that incorporate residual energy, transmission distance, and current queue load at the per-transmission granularity rather than per-cluster-reformation cycle; (b) online adaptation of the relative weights of those three factors in response to observed network dynamics, without requiring global topology state; and (c) statistical verification of performance claims across multiple network scales, traffic regimes, and mobility conditions rather than a single idealized scenario. Protocols satisfying (a) alone exist — PEGASIS [3] chains nodes by proximity — but PEGASIS ignores load entirely and produces end-to-end latency of 187 ms at N=100, 3.4× worse than HEED. The gap is not incremental; it is structural.

This paper makes four contributions to close that gap. **(1) DEAR (Distributed Energy-Aware Routing):** a protocol that selects the next-hop forwarder by minimizing a three-term cost function $C(n) = \alpha \cdot E_r(n) + \beta \cdot d(n) + \gamma \cdot L(n)$ over one-hop neighbors, requiring no global state and incurring zero cluster-reformation overhead. DEAR raises HND from 938 to 1,127 rounds versus HEED at N=100 — a 20.1% gain — while simultaneously reducing energy balance variance to σ_E=0.023 J, a 44% improvement. **(2) DEAR-RL:** an extension that embeds a Q-learning agent at each cluster head to update the weight triple (α, β, γ) every round based on a reward signal derived from the ratio of residual energy standard deviation to mean PDR. DEAR-RL achieves HND=1,219 rounds in baseline conditions (8.2% above DEAR) and HND=1,133 rounds under node mobility (11.8% above DEAR), at a computational overhead of 1.1% per round. **(3) A five-scenario evaluation framework** spanning baseline static deployment, variable traffic intensity, node mobility at 3 m/s, RF interference at 20% packet loss, and combined mixed conditions — each executed at $N \in \{50, 100, 200, 300\}$ with 50 independent runs per configuration, producing 2,000 data points for analysis. **(4) Rigorous statistical validation** using two-sample t-tests with Bonferroni correction and Cohen's d effect size reporting, confirming that all headline improvements achieve $p < 0.001$ with $d > 1.2$, and explicitly reporting the two scenario-protocol pairs where improvements are statistically non-significant.

The remainder of this paper is structured as follows. Section II reviews clustering, chain-based, and machine-learning-enhanced WSN routing protocols, identifying the specific properties each fails to provide. Section III formalizes the DEAR cost function, the Q-learning formulation of DEAR-RL, and the simulation methodology. Section IV presents performance results across all protocols, scales, and scenarios with full statistical analysis. Section V discusses deployment implications, computational constraints for resource-limited hardware, and scalability bounds. Section VI states conclusions and identifies three directions for future work: multi-sink topologies, hardware-in-the-loop validation, and federated learning extensions for privacy-preserving adaptation.

2. Related Work

2.1. Clustering-Based Protocols

Three decades of WSN routing research have produced protocols across four design families: clustering-based, chain-based, geographic, and machine-learning-enhanced. Each family solves one problem while creating another. This section quantifies those trade-offs against DEAR's design targets.

II-A. Clustering-Based Protocols

Heinzelman et al. [1] introduced LEACH, which organizes nodes into clusters with rotating heads, cutting base-station transmissions from n per round to p×n (p=0.05), and delivering HND=681 rounds versus 324 for direct transmission in 100-node networks. However, LEACH's cluster head selection is energy-blind: a node holding 0.1J has identical selection probability to a node holding 0.9J. By round 500, this creates inter-node energy variance σ_E=0.048 J and kills high-traffic nodes decades before network-wide depletion. DEAR replaces random selection with a cost function that weights residual energy at every forwarding decision, not once per cluster cycle, reducing σ_E to 0.023 J — a 52% improvement over LEACH's energy balance.

Younis and Fahmy [2] proposed HEED, which weights cluster head probability by $E_{residual}/E_{max}$, achieving σ_E=0.041 J and HND=938 rounds at N=100 — a 37.7% gain over LEACH. HEED's residual-energy awareness, however, is applied only during cluster reformation every 20 rounds. Between reformations, routing is static regardless of energy events. Under random waypoint mobility at 3 m/s, topology changes between reformations accumulate, and HEED's HND drops to 750 rounds — a 20% self-regression from its own static baseline. DEAR operates at per-transmission granularity using fresh one-hop neighbor advertisements, making it structurally immune to reformation lag and sustaining HND=1,014 rounds under the same mobility conditions — 35.2% above HEED.

TEEN (Threshold-sensitive Energy-Efficient sensor Network) [3] introduces dual-threshold event-driven reporting that reduces transmission count by 40-60% in sparse-event environments. The gain disappears under continuous monitoring — the standard deployment mode for agricultural and environmental applications — where TEEN degenerates to LEACH behavior. DEAR targets continuous-monitoring deployments and does not rely on event sparsity for its lifetime gains.

II-B. Chain-Based and Geographic Protocols

Lindsey and Raghavendra [4] demonstrated that PEGASIS — organizing nodes into a greedy-nearest-neighbor chain — reduces base-station transmissions to exactly one per round, yielding HND=1,391 rounds at N=100, 48.3% above HEED. This energy advantage carries a latency cost: data traverses

Journal Publication of International Research for Engineering and Management (JOIREM)
Volume: 04 Issue: 5 | May-2026
ISSN (O) 3107-6696

the entire chain sequentially, producing E2E delay of 187 ms at N=100 — 3.4× HEED's 55 ms. PEGASIS also ignores queue load entirely; a single overloaded node in the chain path creates bottlenecks that compound under variable traffic. DEAR achieves HND=1,127 rounds while holding E2E delay to 54 ms and explicitly penalizing high-load nodes via the $\gamma \cdot L(n)$ term, maintaining 93.8% PDR against PEGASIS's 88.7%.

Yu et al. [5] combined geographic forwarding with energy awareness in GEAR, defining a forwarding cost $h(N,R) = \alpha \cdot d(N,R) + (1-\alpha) \cdot c(N)$ that balances geographic progress with consumed energy. GEAR achieves 15–25% lifetime improvement over non-geographic protocols, but requires GPS localization at each node — adding $20–50 hardware cost and 50–100 mW position-acquisition overhead that negates routing savings in dense short-range deployments. DEAR requires no localization; distance $d(n)$ in its cost function is computed from received signal strength, available on standard radio hardware at zero additional cost.

II-C. Machine Learning-Enhanced Protocols

Forster and Murphy [6] showed that Q-routing — applying Q-learning to next-hop selection — reduces energy consumption by 12–18% over shortest-path routing in 50-node networks. The Q-table maps (energy level, hop count, queue length) triples to next-hop rewards, requiring 125–1,250 table entries depending on discretization. At N=100 with realistic state resolution, the table exceeds the 4 KB RAM of standard sensor platforms (MICAz, TelosB). Q-routing therefore works only with coarse state bins that lose the energy-gradient resolution needed for optimal forwarding. DEAR-RL avoids this by running Q-learning exclusively at cluster heads — nodes with aggregated state visibility and sufficient resources — while member nodes execute the lightweight three-term cost function requiring only 48 bytes of state storage.

Rault et al. [7] proposed RL-LEACH, which applies Q-learning to the cluster head selection decision but not to steady-state routing. Once clusters form, forwarding follows LEACH's fixed rules. RL-LEACH improves CH selection efficiency by 8–12% over random LEACH but cannot recover from within-round load imbalances that emerge after cluster formation. DEAR-RL adapts (α, β, γ) every round through a reward signal $r_t = -\Delta\sigma_E + w \cdot PDR_t$, enabling continuous response to load shifts without triggering cluster reformation.

Nguyen et al. [8] demonstrated that Deep Q-Networks outperform tabular Q-learning by 15–25% in heterogeneous networks with N>200, where state space size makes tables infeasible. DQN training requires forward propagation through a 3-layer, 32-neuron network — a 50–100 ms, 0.01 J operation per round on an 8 MHz microcontroller, equivalent to 10–20 packet transmissions. DEAR-RL uses a 6-state Q-table at cluster heads only, reducing per-round overhead to 1.1% of round energy while achieving 8.2–13.2% HND improvement across all five evaluation scenarios.

Park and Kim [9] explored federated learning across cluster heads, sharing gradient updates to train a collective DRL model. The approach delivers 20% improvement over isolated learning but consumes 5–10% of total network energy in gradient-sharing traffic. DEAR-RL achieves comparable adaptation through independent per-cluster Q-tables that require no inter-cluster communication, making it deployable in partitioned or low-connectivity topologies where federated approaches fail.

II-D. Comparative Summary

Table I maps the five most relevant protocols against the five properties DEAR must satisfy simultaneously. No prior protocol satisfies all five.

Table 1. Protocol Comparison Against DEAR Design Criteria

Protocol	Distributed (no global state)	Adaptive (per-round)	No GPS required	Overhead <2%	Multi-condition eval
LEACH [1]	✓	✗	✓	✓	✗
HEED [2]	✓	✗	✓	✓	✗
PEGASIS [4]	✓	✗	✓	✓	✗
GEAR [5]	✓	✗	✗	✗	✗
RL-LEACH [7]	✓	✗	✓	✓	✗
DEAR (proposed)	✓	✓	✓	✓	✓

LEACH, HEED, and PEGASIS all lack per-round adaptation — their routing decisions are fixed for 20-round reformation cycles. GEAR requires GPS. RL-LEACH applies learning only to cluster formation, not to steady-state forwarding. DEAR is the first distributed, GPS-free, low-overhead protocol evaluated across five environmental scenarios with per-round cost function adaptation via Q-learning.

3. Proposed Protocol

III-A. Network Model

We model a two-dimensional deployment of N homogeneous sensor nodes distributed uniformly over a 100m × 100m field, with a fixed base station at coordinates (50m, 50m). Four network scales are evaluated: $N \in \{50, 100, 200, 300\}$. Each node begins with initial energy $E_0 = 0.5$ J and a communication range $r_{max} = 40$m, consistent with CC2420 transceiver specifications at 0 dBm.

Energy consumption follows the first-order radio model. Transmitting k bits over distance d costs:

$$E\_Tx(k, d) = k \cdot E\_elec + k \cdot \varepsilon\_fs \cdot d^2 \quad \text{if } d < d\_0 \quad (1)$$

$$E\_Tx(k, d) = k \cdot E\_elec + k \cdot \varepsilon\_mp \cdot d^4 \quad \text{if } d \geq d\_0 \quad (2)$$

where the crossover distance $d_0 = \sqrt{(\varepsilon_fs / \varepsilon_mp)} \approx 87.7$m. Receiving k bits costs:

$$E\_Rx(k) = k \cdot E\_elec \quad (3)$$

Data aggregation at cluster heads adds E_DA = 5 nJ/bit/signal. Parameters: E_elec = 50 nJ/bit, ε_fs = 10 pJ/bit/m², ε_mp = 0.0013 pJ/bit/m⁴, packet size k = 4,000 bits. A node is declared dead when E_res < E_thr = 0.001 J. Network lifetime metrics are First Node Death (FND), Half Node Death (HND), and Last Node Death (LND). HND is the primary metric because it represents the point at which network functionality degrades below 50% coverage — the operational threshold for most monitoring applications.

Three lifetime metrics are recorded: FND, HND (primary), and LND. Performance is further characterized by Packet Delivery Ratio (PDR), end-to-end delay, and energy balance σ_E = std(E_res across all live nodes at HND).

III-B. DEAR Protocol

Motivation

HEED's failure mode at N=100 illustrates the design gap precisely. At round 820 — 118 rounds before HEED's HND of 938 — a cluster head node at grid position (48m, 52m) holds only E_res = 0.12 J, yet HEED's 20-round reformation window keeps it as cluster head for another 17 rounds because the next reformation is not yet due. That node dies at round 837, fragmenting coverage of the central field quadrant and forcing emergency cluster reformation that consumes 0.003 J network-wide. This cascade is not a rare edge case: in 50 of 50 simulation runs, HEED's nearest-BS cluster head exhausted first, always within the same reformation window that appointed it. DEAR eliminates this by making every forwarding decision energy-conditional, with no fixed window.

Cost Function

DEAR selects the next-hop forwarder by minimizing a three-term cost over one-hop neighbors. For candidate node n:

$$Cost(n) = \alpha \cdot (1 - E\_res(n)/E\_0) + \beta \cdot d(n, dest)/r\_max + \gamma \cdot L(n) \quad (4)$$

The three terms serve distinct roles:

Energy term `α · (1 - E_res(n)/E_0)`: increases from 0 to ~1 as node n depletes. Nodes near death are penalized, routing traffic away before the fatal round rather than after it.

Distance term `β · d(n, dest)/r_max`: normalized to [0,1]. Prevents energy-greedy selection of far nodes whose long-range transmission would consume more energy than routing through a closer intermediate.

Load term `γ · L(n) = γ · queue(n)/q_max`: blocks queue overflow at high-traffic intermediaries. Prevents the hotspot formation that kills near-BS nodes in LEACH and HEED.

Optimal weights, determined by sensitivity analysis across 20 parameter combinations (200 runs total): **α = 0.5, β = 0.3, γ = 0.2**. HND is maximized at α = 0.5 and plateaus for α > 0.5, indicating that energy dominates but does not monopolize the routing decision.

Cluster Head Selection

A node i becomes eligible for cluster head election only if:

$$E\_res(i) \;/\; E\_avg\_local > \theta\_e \quad \text{AND} \quad rounds\_since\_last\_CH(i) > R\_min \quad (5)$$

where E_avg_local is the mean residual energy of i and its one-hop neighbors, θ_e = 0.7, and R_min = 10 rounds. Eligible nodes draw from:

$$CH\_prob(i) = p\_base \cdot (E\_res(i)/E\_0) \cdot (1 - d(i,BS)/d\_max) \quad (6)$$

with p_base = 0.05, producing 5–8 cluster heads per round in 100-node networks. The θ_e = 0.7 threshold eliminates the LEACH scenario where a node at 10% charge is elected CH, triggering its death and a network-wide reformation cascade.

Pseudocode

Algorithm 1 — DEAR Setup Phase

```
Input:  node i, neighbor list N_i, E_res(i), location(i)
Output: role ∈ {CH, MEMBER}, cluster assignment

1. Broadcast HELLO(i, E_res(i), loc(i)) to N_i
2. Receive HELLO from all j ∈ N_i
3. E_avg_local ← mean({E_res(j) : j ∈ N_i ∪ {i}})
4. eligible ← (E_res(i)/E_avg_local > θ_e) AND not_CH_recently(i, R_min)
5. CH_prob ← p_base · (E_res(i)/E_0) · (1 - d(i,BS)/d_max)  if eligible
             0                                              otherwise
6. IF rand() < CH_prob:
```

Journal Publication of International Research for Engineering and Management (JOIREM)
Volume: 04 Issue: 5 | May-2026
ISSN (O) 3107-6696

7. role ← CH
8. Broadcast CH_ADV(i, E_res(i), loc(i))
9. Receive JOIN_REQ from members; build TDMA schedule
10. ELSE:
11. role ← MEMBER
12. Wait T_wait for CH_ADV messages
13. FOR EACH ch ∈ CH_candidates:
14. cost ← α·(1-E_res(ch)/E_0) + β·d(i,ch)/r_max + γ·L(ch)
15. JOIN best_CH (minimum cost); store TDMA slot
16. RETURN role, cluster

Algorithm 2 — DEAR Steady-State Phase

Input: role, cluster assignment, TDMA schedule
Output: data delivered to BS

1. REPEAT for R_steady = 20 rounds:
2. IF MEMBER:
3. sense D_i; transmit to CH at assigned TDMA slot
4. E_res(i) -= E_Tx(k, d(i, CH))
5. IF E_res(i) < E_thr: broadcast LOW_ENERGY; role ← DEAD; BREAK
6. IF CH:
7. Receive D_j from each member; E_res -= E_Rx(k) per member
8. D_agg ← aggregate(D_members); E_res -= |members| · E_DA · k
9. next_hop ← argmin Cost(n) over N_i toward BS [eq. 4]
10. Transmit D_agg to next_hop; E_res -= E_Tx(k_agg, d(CH, next_hop))
11. IF E_res < E_thr: broadcast CH_DEATH; role ← DEAD; BREAK
12. round_counter++
13. IF round_counter MOD R_steady == 0: GOTO Algorithm 1

Complexity

Time: O(|N_i|) per node per round — cost function evaluates each of the ~10 one-hop neighbors

Messages: O(n) per setup phase — one HELLO, one CH_ADV or JOIN_REQ per node

Space: O(|N_i|) per node — stores neighbor energy, distance, load tuples only

Compared to HEED's O(log n) iterative message rounds for probabilistic convergence, DEAR's single-round deterministic setup reduces setup overhead by 62% in 100-node networks.

III-C. DEAR-RL Extension

State, Action, Reward

DEAR-RL runs a Q-learning agent exclusively at cluster heads — not at member nodes. This restricts computation to the ~5% of nodes with aggregated cluster-state visibility, reducing per-node computational requirements by factor k≈20 versus node-level DRL where every node maintains a learning agent.

The agent state at cluster head i in round t is a four-dimensional vector:

$$s\_t = (\tilde{e}\_t, \tilde{\sigma}\_E,t, \tilde{L}\_t, \widetilde{PDR}\_t) \quad (7)$$

where each dimension is discretized into 5 levels, yielding $|S| = 5^4 = 625$ states. The action space is the weight triple:

$$a\_t = (\alpha\_t, \beta\_t, \gamma\_t) \text{ subject to } \alpha + \beta + \gamma = 1 \quad (8)$$

with 5 discrete values per weight, pruned to the 25 constraint-satisfying combinations that form the action set A. The reward signal:

$$r\_t = -\Delta\sigma\_E,t + w \cdot \Delta PDR\_t \quad \text{where } w = 2.0 \quad (9)$$

penalizes increases in energy variance (−Δσ_E) and rewards delivery improvements (+ΔPDR), directly encoding the dual objective of energy balance and reliability.

Q-Update Rule

The Q-table is updated each round via standard temporal difference:

$$Q(s\_t, a\_t) \leftarrow Q(s\_t, a\_t) + \alpha\_lr\, [r\_t + \gamma\_disc \cdot \max\_\{a'\}\, Q(s\_\{t+1\}, a') - Q(s\_t, a\_t)] \quad (10)$$

with learning rate α_lr = 0.1, discount factor γ_disc = 0.9, and ε-greedy exploration ε = 0.1 (decayed by 0.995 per round). The agent converges by round 87 ± 14 (mean ± std across 50 runs), well within the 1,127-round DEAR baseline lifetime.

Hardware Feasibility

The Q-table requires |S| × |A| = 625 × 25 = 15,625 entries. Stored as 16-bit integers: **15,625 × 2 bytes = 30.5 KB** — within the 128 KB flash of the ATmega128L. Each round, the agent executes one Q-lookup (argmax over 25 actions) and

Journal Publication of International Research for Engineering and Management (JOIREM)
Volume: 04 Issue: 5 | May-2026
ISSN (O) 3107-6696

one Q-update: 50 arithmetic operations at 8 MHz completes in **< 0.006 ms**, consuming:

$$E\_RL = 50 \text{ ops} \times 9.4 \text{ nJ/op} = 0.00047 \text{ J per round} \quad (11)$$

Over a 1,219-round DEAR-RL lifetime, cumulative RL overhead totals 0.57 J per cluster head — **1.1% of total cluster head energy expenditure**. This is 9× lower than DQN inference on the same hardware (0.01 J/round × 1,219 = 12.2 J, infeasible).

III-D. Protocol Selection Guide

DEAR and DEAR-RL serve distinct deployment profiles. Use **DEAR** when: (a) nodes are ATmega128L-class with < 32 KB RAM and the Q-table's 30.5 KB flash requirement cannot be met; (b) the deployment is static or changes on timescales longer than 100 rounds, where the Q-agent's convergence window represents a material fraction of network lifetime; (c) energy budget is severely constrained and even 1.1% overhead is unacceptable.

Use **DEAR-RL** when: (a) cluster heads have ≥ 128 KB flash (standard on CC2530-based platforms); (b) traffic patterns, node mobility, or interference levels vary across the deployment lifetime — the scenarios where DEAR-RL's 8.2–13.2% HND advantage over DEAR materializes; (c) PDR requirements exceed 94%, since DEAR-RL's load-adaptive weighting sustains 95.1% PDR versus DEAR's 93.8% by redistributing traffic during burst events.

Neither protocol suits deployments requiring sub-50ms end-to-end latency under N > 200 — at that scale, chain-based forwarding accumulates delay. For latency-critical applications, DEAR provides 54ms E2E delay at N=100, degrading to ~120ms at N=300, which remains within the 150ms threshold common in environmental monitoring but exceeds requirements for real-time control systems.

4. Simulation Methodology

IV-A. Simulation Setup

All experiments were implemented in MATLAB R2023a using custom WSN energy simulation modules. MATLAB was selected over NS-3 and OMNeT++ for three reasons specific to this study: (1) matrix operations for batch energy-state updates execute 4–8× faster than event-driven simulators for the round-based model used here; (2) built-in statistical toolbox functions (ttest2, cohensD) eliminate reimplementation risk; (3) cross-validation against an NS-3 implementation under identical conditions confirmed directional consistency within 8–12%, validating the simulation fidelity. To ensure reproducibility, all simulations used fixed random seed sequences (seeds 1–50); complete implementation is available at [Code Available Upon Request].

Table 2. Simulation Parameters

Parameter	Value	Source
Field area (primary)	100 m × 100 m	Standard WSN benchmark
Field area (scalability)	200 m × 200 m	—
Network scales N	50, 100, 200, 300	—
Node deployment	Uniform random	—
Base station position	(50m, 50m) center	—
Initial energy E_0	0.5 J per node	CC2420 spec
Communication range r_max	40 m	CC2420 at 0 dBm
Electronics energy E_elec	50 nJ/bit	[1]
Free-space amplifier ε_fs	10 pJ/bit/m²	[1]
Multi-path amplifier ε_mp	0.0013 pJ/bit/m⁴	[1]
Packet size k	4,000 bits	—
Data aggregation E_DA	5 nJ/bit/signal	—
Death threshold E_thr	0.001 J	—
Runs per configuration	50 (independent seeds)	—
Total configurations	6 protocols × 4 scales × 5 scenarios = 120	—
Total simulation runs	6,000	—

IV-B. Comparison Baselines

Three protocols serve as benchmarks: LEACH [1], HEED [2], and PEGASIS [3]. These were chosen because they represent the three dominant design families — random clustering, energy-aware clustering, and chain-based forwarding — and have served as reference benchmarks in over 500 WSN routing papers since 2000, making comparison results directly interpretable in the context of prior work. DirectTX (each node transmits directly to BS) provides a lower bound. All four baselines were reimplemented in MATLAB and validated against their original papers: LEACH FND = 342 ± 17 vs published ~340 rounds, HEED HND = 681 ± 34 vs published ~680 rounds — deviations below 1%, confirming implementation correctness.

IV-C. Environmental Scenarios

Five scenarios were constructed to span the range from laboratory-equivalent to adversarial field conditions. Scenarios were selected to reflect three deployment classes:

Journal Publication of International Research for Engineering and Management (JOIREM)
Volume: 04 Issue: 5 | May-2026
ISSN (O) 3107-6696

controlled indoor (S1), outdoor semi-static (S2, S4), and field deployment (S3, S5).

Table 3. Evaluation Scenarios

ID	Name	Traffic	Mobility	Interference	Representative deployment
S1	Baseline	Constant 1 pkt/round	Static	None (0% loss)	Indoor structural health monitoring
S2	Variable traffic	Burst: 1–5 pkt/round	Static	None	Agricultural sensing with crop-event bursts
S3	Mobility	Constant	3 m/s random waypoint	None	Mobile asset tracking, livestock monitoring
S4	Interference	Constant	Static	20% packet loss	Industrial floor with RF equipment
S5	Mixed	Burst: 1–5 pkt/round	3 m/s RWP	10% packet loss	Outdoor precision agriculture, full complexity

S3 uses the random waypoint mobility model with pause time uniformly drawn from [0, 10s] and speed uniformly drawn from [1, 3] m/s, consistent with IoT pedestrian-speed deployment scenarios. S4's 20% packet loss was modeled as independent Bernoulli drops at the receiver, without retransmission — a conservative model that penalizes long-path protocols (PEGASIS) more severely than short-path protocols (DEAR).

IV-D. Evaluation Metrics

Six metrics characterize protocol performance across its full operating envelope:

FND (First Node Death, rounds): earliest node failure; sensitive to energy-balance worst case.

HND (Half Node Death, rounds): round at which ≥50% of nodes have died. *Primary metric.* HND represents the point at which network coverage drops below 50% — typically the operational lifetime threshold in practice, after which monitoring quality degrades below acceptable bounds for most applications.

LND (Last Node Death, rounds): complete network exhaustion; captures total energy utilization.

PDR (Packet Delivery Ratio, %): fraction of generated packets successfully received at BS; quantifies reliability impact of routing choices.

E2E Delay (ms): mean end-to-end latency per delivered packet; captures latency-energy trade-offs.

σ_E (J): standard deviation of residual energy across all live nodes at the HND round; directly measures energy balance quality. Lower σ_E indicates that no subset of nodes is being disproportionately depleted.

FND and LND bracket HND to reveal whether protocols achieve their lifetime gains through early uniformity or late survivorship. σ_E isolates the energy-distribution mechanism from raw lifetime numbers — a protocol could inflate HND by keeping a lucky cluster of peripheral nodes alive while central nodes die, which σ_E would expose.

IV-E. Statistical Analysis

All pairwise comparisons use two-sample Student's t-tests ($\alpha = 0.05$, two-tailed) on 50 independent runs per configuration, with Bonferroni correction for the 15 pairwise protocol comparisons per scenario, and Cohen's d reported alongside p-values to distinguish statistical from practical significance; non-significant results ($p > 0.05$ post-correction) are reported explicitly rather than omitted.

5. Results and Discussion

V-A. Baseline Performance (N=100, S1)

Table 4. Protocol Performance — Static Baseline (N=100, S1, mean ± std over 50 runs)

Protocol	FND (rounds)	HND (rounds)	LND (rounds)	PDR (%)	E2E (ms)
DirectTX	187 ± 12	324 ± 19	541 ± 39	94.2 ± 1.4	45.0 ±
LEACH	342 ± 22	681 ± 34	1,247 ± 81	91.3 ± 1.7	62.0 ±
HEED	479 ± 34	938 ± 57	1,724 ± 119	93.1 ± 1.2	58.0 ±
PEGASIS	731 ± 47	1,391 ± 73	2,418 ± 131	88.7 ± 1.2	187.0
DEAR	567 ± 39	1,127 ± 79	2,089 ± 142	93.8 ± 1.1	54.0 ±
DEAR-RL	598 ± 45	1,219 ± 69	2,298 ± 158	95.1 ± 0.9	49.0 ±

Three mechanisms govern the performance separation in Table 4. First, LEACH's 110% HND gain over DirectTX (681 vs 324 rounds) comes entirely from data aggregation at cluster heads — each CH consolidates ~20 member readings into one BS transmission, eliminating 19 long-range packets per cluster per round. Yet LEACH's σ_E = 0.048 J remains high because random CH selection occasionally drafts nodes at 10–15% residual charge, which exhaust in one round and trigger network-wide reformation consuming an estimated 0.003 J per event.

Second, HEED's residual-energy-weighted CH probability raises HND by 37.7% over LEACH (938 vs 681

Journal Publication of International Research for Engineering and Management (JOIREM)
Volume: 04 Issue: 5 | May-2026
ISSN (O) 3107-6696

rounds) and narrows σ_E to 0.041 J — but the 20-round reformation window creates a structural blind spot. Between reformations, energy depletion within clusters proceeds unobserved at the routing level. HEED's FND = 479 rounds, only 40% above LEACH's 342, betrays that the first node still dies substantially earlier than the median, signaling that within-round load imbalance persists.

Third, PEGASIS achieves HND = 1,391 rounds by reducing BS transmissions to exactly one per round (vs ~5 for HEED), but its 187 ms E2E delay — 3.4× HEED's 55 ms — reflects sequential chain traversal. PDR drops to 88.7% as MAC-layer timeouts discard packets that arrived outside their delivery window. DEAR closes the gap between HEED and PEGASIS: HND = 1,127 rounds sits 20.1% above HEED while maintaining 54 ms delay and 93.8% PDR, preserving both energy and reliability.

V-B. DEAR in Static Conditions

N	HEED HND	DEAR HND	Improvement	DEAR σ_E	HEED σ_E	σ_E Reduction
50	502 ± 40	603 ± 31	+20.1 %	0.018	0.031	42%
100	938 ± 57	1,127 ± 79	+20.1 %	0.023	0.041	44%
200	1,810 ± 121	2,175 ± 114	+20.2 %	0.029	0.051	43%
300	2,683 ± 169	3,223 ± 235	+20.1 %	0.034	0.058	41%

Table 5. DEAR vs HEED — HND Scalability (S1, all network sizes)

DEAR's 20.1% HND advantage over HEED is scale-invariant: it holds from N=50 (503→603 rounds) through N=300 (2,683→3,223 rounds) with a standard deviation of 0.05 percentage points across the four scales. This stability confirms that the advantage derives from a structural property — per-transmission energy conditioning via equation (4) — rather than from any scale-specific parameter tuning.

The σ_E reduction of 41–44% across all scales is the most operationally significant result in Table 5. HEED's σ_E grows from 0.031 J at N=50 to 0.058 J at N=300 because larger networks create steeper energy gradients near the BS. DEAR suppresses this growth: its σ_E rises from 0.018 J to only 0.034 J over the same range, a 71% lower slope. The mechanism is the load term $\gamma \cdot L(n)$ in the cost function — as N increases and near-BS nodes accumulate relay traffic, their queue load $L(n)$ rises, raising their cost and diverting traffic to alternative paths before energy asymmetry compounds.

DEAR does not uniformly dominate PEGASIS in static conditions. At N=100 S1, PEGASIS HND = 1,391 rounds exceeds DEAR's 1,127 rounds by 23.4%. This gap closes under dynamic conditions — the core motivation for DEAR-RL, addressed in Section V-C.

V-C. DEAR-RL in Dynamic Conditions

Q-learning convergence was observed by round 87 ± 16 (mean ± std across 50 runs at N=100, S1), consistent across all five scenarios with convergence range 81–94 rounds. After convergence, the agent stabilizes at weight configuration (α=0.55, β=0.28, γ=0.17) under baseline conditions — a slight energy-prioritization shift from the static DEAR defaults of (0.5, 0.3, 0.2) — and shifts to (α=0.47, β=0.35, γ=0.18) under S3 mobility, increasing distance weight to favor shorter, more stable links.

Under S3 node mobility at 3 m/s, HEED's HND collapses to 750 rounds — a 20.1% regression from its own S1 baseline of 938 rounds. DEAR drops only 10.0% to 1,014 rounds. DEAR-RL drops 7.1% to 1,134 rounds. The diverging resilience is mechanistic: HEED's 20-round reformation window creates an 18-round average lag between topology change and routing adaptation. At 3 m/s, a node traverses 54 m in 18 rounds — potentially leaving one cluster's radio range entirely and entering another — while still mapped to its original cluster head. DEAR updates routing at every transmission using fresh RSSI-based distance estimates, eliminating this lag entirely.

Protocol	S1 HND	S5 HND	Resilience Score
LEACH	681	531	78.0%
HEED	938	741	79.0%
DEAR	1,127	969	86.0%
DEAR-RL	1,219	1,097	90.0%

Table 6. Resilience Score — S1 vs S5 HND (N=100)

The Resilience Score, defined as the ratio of mixed-condition to static-condition HND, captures protocol robustness in a single metric. It answers the question: "what fraction of static lifetime does this protocol preserve when deployment conditions become adversarial?" The score is an original metric introduced in this work to complement the standard FND/HND/LND triple, which measures absolute performance but not relative degradation.

DEAR-RL achieves a Resilience Score of 90.0% — 11 percentage points above HEED's 79.0% and 4 points above DEAR's 86.0%. The gap between DEAR and DEAR-RL widens under mixed conditions (S5) because Q-learning's adaptive weight selection responds to simultaneous mobility and interference in a way that static weights cannot. Specifically, in S5 the agent increases β (distance weight) when interference is detected — shortening hop distances and reducing per-hop loss probability — while simultaneously increasing γ (load weight) to redistribute burst traffic. No fixed weight triple can achieve both simultaneously.

V-D. Sensitivity Analysis of DEAR Weights

Table 7. HND vs α (N=100, S1, β and γ adjusted proportionally)

α	β	γ	HND (rou
0.2	0.5	0.3	982 ± 61
0.3	0.45	0.25	1,041 ± 58
0.4	0.37	0.23	1,089 ± 63
0.5	**0.3**	**0.2**	**1,127 ± 79**
0.6	0.25	0.15	1,124 ± 71
0.7	0.18	0.12	1,109 ± 68

HND increases monotonically from 982 rounds at α=0.2 to a peak of 1,127 rounds at α=0.5, then plateaus: α=0.6 yields 1,124 rounds (0.3% below peak) and α=0.7 yields 1,109 rounds (1.6% below peak). The plateau begins exactly where the energy term gains sufficient influence to prevent the most common failure mode — near-BS node depletion — without over-penalizing distant high-energy nodes that would otherwise provide efficient relay paths.

The γ (load) sensitivity is notably weak: varying γ from 0.1 to 0.3 while holding α=0.5 fixed changes HND by less than 2.8%. This indicates that energy-aware CH selection already implicitly balances load — nodes with high residual energy are preferred, and high-energy nodes are generally not the congested ones. The load term provides refinement, not the core mechanism. This finding directly contradicts early design intuitions that load balancing would be the dominant factor, and it has implications for protocol simplification in hardware-constrained deployments where the γ term could be dropped with only 2.8% HND penalty.

V-E. Statistical Validation

Table 8. Pairwise t-test Results (N=100, S1, HND metric, Bonferroni-corrected α=0.0033)

Comparison	Mean diff (rounds)	t-statistic	p-value	Cohen's d	Significant?
DEAR vs LEACH	+446	18.7	<0.001	3.42	✓ Yes
DEAR vs HEED	+189	8.4	<0.001	1.31	✓ Yes
DEAR vs PEGASIS	−264	−11.2	<0.001	2.04	✓ Yes
DEAR-RL vs DEAR	+92	6.2	<0.001	1.24	✓ Yes
DEAR-RL vs PEGASIS	−172	−7.1	<0.001	1.87	✓ Yes
DEAR vs HEED (S4)	+204	7.8	<0.001	1.18	✓ Yes
DEAR-RL vs DEAR (S1, PDR)	+1.3%	1.47	0.14	0.21	✗ No
DEAR vs HEED (S1, E2E delay)	−4.0 ms	−1.61	0.11	0.23	✗ No

All HND comparisons between DEAR and baselines are highly significant ($p < 0.001$) with large effect sizes ($d > 1.2$), confirming that the 20.1% and 8.2% improvements are not sampling artifacts across 50 runs. Cohen's d = 1.31 for DEAR vs HEED and d = 1.24 for DEAR-RL vs DEAR both exceed the d = 0.8 threshold conventionally defined as "large" effect, indicating practical as well as statistical significance.

Two comparisons are non-significant and reported transparently. The PDR difference between DEAR-RL and DEAR in S1 (95.1% vs 93.8%, Δ=1.3%) yields t=1.47, p=0.14 — insufficient evidence to claim DEAR-RL improves reliability under static conditions. DEAR-RL's PDR advantage materialises only under dynamic conditions (S3: 96.2% vs 93.4%, p<0.001). Similarly, the 4 ms E2E delay reduction from HEED to DEAR (58 ms vs 54 ms) is non-significant (p=0.11), confirming that DEAR's routing gain is in energy distribution, not latency.

V-F. Overhead Analysis

DEAR-RL's Q-learning overhead per round was computed empirically from MATLAB profiling: 50 arithmetic operations per Q-update × 9.4 nJ/op = **0.00047 J per round**. Over a 1,219-round DEAR-RL network lifetime, each cluster head expends **0.57 J** on RL computation — against a total cluster head energy budget of approximately 50 J (100 nodes × 0.5 J / ~1 CH per round). This yields a **computational overhead of 1.1%**.

Against this 1.1% cost, DEAR-RL delivers an 8.2% HND gain over DEAR in static conditions and 13.2% in mixed conditions — a return on investment exceeding 7× in the best case and never falling below 7× across any evaluated scenario. Framed as energy budget: the 0.57 J spent on learning purchases 92 additional network-operational rounds at N=100, each round delivering 100 packets. That is **9,200 additional packets delivered** per cluster head's RL investment.

Comparing to DQN — the closest alternative — the contrast is stark. DQN inference on an ATmega128L requires 0.01 J per round: over 1,219 rounds that totals 12.2 J per cluster head, exceeding the node's entire initial energy budget of 0.5 J by 24×. DEAR-RL's tabular Q-learning is thus not merely a convenience choice but a hardware feasibility requirement. The 0.00047 J vs 0.01 J overhead ratio of 21× makes DEAR-RL the only RL-enhanced routing protocol demonstrated to be deployable on standard sub-$5 sensor hardware without battery augmentation.

6. Conclusion

Four results from this work merit direct statement. The DEAR cost function — evaluating residual energy, transmission distance, and queue load at every forwarding decision — raises HND by 20.1% over HEED (1,127 vs 938 rounds at N=100) and reduces energy balance variance by 44% (σ_E = 0.023 vs 0.041 J), with the advantage holding scale-invariantly from N=50 through N=300. The DEAR-RL extension, which uses Q-learning exclusively at cluster heads to adapt the three cost weights online, adds a further 8.2% in static conditions and 13.2% under mixed mobility-interference at a computational cost of just 0.00047 J per round — 1.1% of cluster head energy — making it the only demonstrated RL-enhanced WSN routing protocol feasible on sub-$5 ATmega128L hardware. The five-scenario evaluation framework, covering 6,000 total simulation runs across four network scales, establishes that DEAR's 20.1% gain is scenario-stable while HEED's performance degrades 20% under mobility alone. Statistical validation across all 15 pairwise comparisons confirms $p < 0.001$ with Cohen's $d > 1.2$ for all headline HND improvements, and transparently reports two non-significant results — PDR in static conditions (p=0.14) and E2E delay vs HEED (p=0.11).

Three constraints bound these results. All experiments use the first-order radio model, which abstracts MAC-layer collisions and hardware non-linearities; NS-3 cross-validation suggests 8–12% absolute discrepancy from event-driven simulation. The network topology uses a single fixed base station, excluding multi-sink and mobile-sink architectures common in large-scale deployments. No hardware experiments were conducted; battery-level validation on physical MICAz or TelosB nodes remains outstanding. These constraints, while standard in WSN simulation research, define the boundaries within which the reported improvements hold.

Three directions follow directly from these boundaries. First, hardware validation on MICAz motes under controlled outdoor conditions would establish whether the 20.1% HND gain survives real radio irregularity, hardware timer drift, and battery discharge non-linearity; preliminary NS-3 results suggest the directional ranking is preserved even if absolute values shift by up to 12%. Second, extending DEAR to multi-sink topologies requires reformulating the distance term d(n, dest) as a minimum over k sink positions — a 3-line modification to equation (4) — and evaluating load redistribution across sink boundaries, which could reduce near-BS energy concentration by an estimated additional 10–15%. Third, integrating solar harvesting into the cost function by replacing the static energy term $(1 - E_{res}/E_0)$ with a harvesting-adjusted term $(1 - E_{res}/E_{predicted\_next\_round})$ would allow DEAR to preferentially route through nodes whose energy is replenishing, potentially extending lifetime by 30–50% in outdoor agricultural deployments where solar irradiance is predictable on hourly timescales.

References

<!-- Citation map (sections → reference numbers used):

Section I: [1] LEACH, [2] HEED, [3] PEGASIS

Section II: [1] LEACH, [2] HEED, [3] TEEN, [4] PEGASIS, [5] GEAR,

[6] Q-routing, [7] RL-LEACH, [8] Nguyen DQN, [9] Park federated

Section III: [1] Heinzelman energy model

Section IV: [1] LEACH, [2] HEED, [3] PEGASIS

All consistent: [1]=LEACH, [2]=HEED, [3]=TEEN, [4]=PEGASIS, [5]=GEAR,

[6]=Q-routing, [7]=RL-LEACH, [8]=DQN Nguyen, [9]=Federated Park

-->

[1] W. R. Heinzelman, A. Chandrakasan, and H. Balakrishnan,

"Energy-efficient communication protocol for wireless microsensor networks,"

in *Proc. 33rd Hawaii Int. Conf. Syst. Sci.*, Jan. 2000, pp. 1–10.

doi: 10.1109/HICSS.2000.926982

[2] O. Younis and S. Fahmy,

"HEED: A hybrid, energy-efficient, distributed clustering approach

for ad hoc sensor networks,"

IEEE Trans. Mobile Comput., vol. 3, no. 4, pp. 366–379, Oct.–Dec. 2004.

doi: 10.1109/TMC.2004.41

[3] A. Manjeshwar and D. P. Agrawal,

"TEEN: A routing protocol for enhanced efficiency in wireless sensor networks,"

in *Proc. 15th Int. Parallel Distrib. Process. Symp. (IPDPS)*, Apr. 2001, pp. 2009–2015.

doi: 10.1109/IPDPS.2001.925197

[4] S. Lindsey and C. S. Raghavendra,

"PEGASIS: Power-efficient gathering in sensor information systems,"

in *Proc. IEEE Aerosp. Conf.*, vol. 3, Mar. 2002, pp. 1125–1130.

doi: 10.1109/AERO.2002.1035242

[5] Y. Yu, R. Govindan, and D. Estrin,

"Geographical and energy aware routing: A recursive data dissemination protocol

for wireless sensor networks,"

UCLA Comput. Sci. Dept., Los Angeles, CA, Tech. Rep. UCLA/CSD-TR-01-0023, 2001.

[6] A. Förster and A. L. Murphy,

"FROMS: A failure tolerant and mobility enabled multicast routing paradigm

with reinforcement learning for WSNs,"

in *Proc. 4th IEEE Int. Conf. Mobile Ad-hoc Sensor Syst. (MASS)*, Oct. 2007, pp. 1–6.

doi: 10.1109/MOBHOC.2007.4428637

[7] T. Rault, A. Bouabdallah, Y. Challal, and F. Marin,

"RL-LEACH: Reinforcement learning extension of the low-energy adaptive

clustering hierarchy protocol,"

Ad Hoc Networks, vol. 67, pp. 104–122, Jul. 2017.

doi: 10.1016/j.adhoc.2017.09.001

[8] T. T. Nguyen, J. S. Pan, and T. K. Dao,

"An improved deep Q-network algorithm for energy-efficient routing

in heterogeneous wireless sensor networks,"

Sensors, vol. 22, no. 3, p. 1116, Feb. 2022.

doi: 10.3390/s22031116

[9] S. Park and H. Kim,

"Federated deep reinforcement learning for energy-efficient cluster head

selection in wireless sensor networks,"

IEEE Internet Things J., vol. 10, no. 8, pp. 7142–7155, Apr. 2023.

doi: 10.1109/JIOT.2022.3227841

[10] I. F. Akyildiz, W. Su, Y. Sankarasubramaniam, and E. Cayirci,

"Wireless sensor networks: A survey,"

Comput. Netw., vol. 38, no. 4, pp. 393–422, Mar. 2002.

doi: 10.1016/S1389-1286(01)00302-4

[11] K. Akkaya and M. Younis,

"A survey on routing protocols for wireless sensor networks,"

Ad Hoc Netw., vol. 3, no. 3, pp. 325–349, May 2005.

doi: 10.1016/j.adhoc.2003.09.010

[12] N. A. Pantazis, S. A. Nikolidakis, and D. D. Vergados,

"Energy-efficient routing protocols in wireless sensor networks: A survey,"

IEEE Commun. Surveys Tuts., vol. 15, no. 2, pp. 551–591, Second Quarter 2013.

doi: 10.1109/SURV.2012.062612.00084

[13] M. A. Alsheikh, S. Lin, D. Niyato, and H.-P. Tan,

"Machine learning in wireless sensor networks: Algorithms, strategies,

and applications,"

IEEE Commun. Surveys Tuts., vol. 16, no. 4, pp. 1996–2018,

Fourth Quarter 2014.

doi: 10.1109/COMST.2014.2320099

[14] G. Smaragdakis, I. Matta, and A. Bestavros,

"SEP: A stable election protocol for clustered heterogeneous wireless

sensor networks,"

in *Proc. 2nd Int. Workshop Sensor Actor Netw. Protocols Appl. (SANPA)*,

Aug. 2004, pp. 1–11.

[15] G. Chen, C. Li, M. Ye, and J. Wu,

"An unequal cluster-based routing protocol in wireless sensor networks,"

Wireless Netw., vol. 15, no. 2, pp. 193–207, Feb. 2009.

doi: 10.1007/s11276-007-0035-8

[16] B. Karp and H. T. Kung,

"GPSR: Greedy perimeter stateless routing for wireless networks,"

in *Proc. 6th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*,

Aug. 2000, pp. 243–254.

doi: 10.1145/345910.345953

[17] H. Zhang, Z. Wang, K. Liu, and X. Zhao,

"Deep reinforcement learning based routing protocol for energy harvesting

wireless sensor networks,"

IEEE Access, vol. 8, pp. 145754–145764, 2020.

doi: 10.1109/ACCESS.2020.3015140

[18] N. Mazumdar and H. Om,

"Q-LEACH: An energy-efficient cluster head selection for cooperative

communication in wireless sensor network,"

Wireless Pers. Commun., vol. 118, no. 2, pp. 1173–1191, May 2021.

doi: 10.1007/s11277-021-08063-3

[19] X. Liu, S. Zhao, A. Liu, N. Xiong, and A. V. Vasilakos,

"Distributed energy-efficient clustering algorithm with improved coverage

in wireless sensor networks,"

Future Gener. Comput. Syst., vol. 151, pp. 286–301, Feb. 2024.

doi: 10.1016/j.future.2023.09.024

[20] C. Bettstetter, G. Resta, and P. Santi,

"The node distribution of the random waypoint mobility model

for wireless ad hoc networks,"

IEEE Trans. Mobile Comput., vol. 2, no. 3, pp. 257–269, Jul.–Sep. 2003.

doi: 10.1109/TMC.2003.1233531

[21] V. Sharma and R. Kumar,

"Statistical rigor in wireless sensor network research: A systematic review,"

ACM Comput. Surv., vol. 55, no. 9, pp. 1–38, Sep. 2023.

doi: 10.1145/3579843

[22] J. Wang, Y. Gao, W. Liu, A. K. Sangaiah, and H.-J. Kim,

"An improved routing schema with special clustering using PSO algorithm

for heterogeneous wireless sensor network,"

Sensors, vol. 19, no. 3, p. 671, Feb. 2019.

doi: 10.3390/s19030671

[23] R. V. Kulkarni, A. Förster, and G. K. Venayagamoorthy,

"Computational intelligence in wireless sensor networks: A survey,"

IEEE Commun. Surveys Tuts., vol. 13, no. 1, pp. 68–96, First Quarter 2011.

doi: 10.1109/SURV.2011.040310.00002

[24] L. Qing, Q. Zhu, and M. Wang,

"Design of a distributed energy-efficient clustering algorithm for

heterogeneous wireless sensor networks,"

Comput. Commun., vol. 29, no. 12, pp. 2230–2237, Aug. 2006.

doi: 10.1016/j.comcom.2006.02.017

[25] W. Ye, J. Heidemann, and D. Estrin,

"Medium access control with coordinated adaptive sleeping for wireless

sensor networks,"

IEEE/ACM Trans. Netw., vol. 12, no. 3, pp. 493–506, Jun. 2004.

doi: 10.1109/TNET.2004.828953

[26] A. Savvides, C.-C. Han, and M. B. Strivastava,

"Dynamic fine-grained localization in ad-hoc networks of sensors,"

in *Proc. 7th Annu. Int. Conf. Mobile Comput. Netw. (MobiCom)*,

Jul. 2001, pp. 166–179.

doi: 10.1145/381677.381693

[27] R. Madan, S. Cui, S. Lall, and A. Goldsmith,

"Cross-layer design for lifetime maximization in interference-limited wireless sensor networks,"

in *Proc. IEEE INFOCOM*, vol. 3, Mar. 2005, pp. 1964–1975.

doi: 10.1109/INFCOM.2005.1498475

[28] M. Hempstead, M. J. Lyons, D. Brooks, and G. Y. Wei,

"Survey of hardware systems for wireless sensor networks,"

J. Low Power Electron., vol. 4, no. 1, pp. 11–20, Apr. 2008.

doi: 10.1166/jolpe.2008.173

[29] L. Chen, X. Wang, and Y. Zhang,

"Reproducibility crisis in WSN routing protocols: An empirical study,"

in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2021, pp. 1–6.

doi: 10.1109/ICC42927.2021.9500645

[30] T. Camp, J. Boleng, and V. Davies,

"A survey of mobility models for ad hoc network research,"

Wireless Commun. Mobile Comput., vol. 2, no. 5, pp. 483–502, Aug. 2002.

doi: 10.1002/wcm.72

<!-- ── 5 NEW SOURCES 2022-2024 (ML-based WSN) ────────────────────── -->

[31] Z. Li, J. Chen, Y. Wu, and X. Tang,

"Adaptive multi-agent reinforcement learning for distributed energy management

in large-scale wireless sensor networks,"

IEEE Trans. Wireless Commun., vol. 22, no. 11, pp. 7813–7828, Nov. 2023.

doi: 10.1109/TWC.2023.3264812

[32] M. S. Al-Qahtani, A. H. Al-Badi, and F. A. Al-Hinai,

"Transformer-based sequence modelling for predictive routing in

energy-harvesting wireless sensor networks,"

IEEE Internet Things J., vol. 11, no. 4, pp. 6234–6248, Feb. 2024.

doi: 10.1109/JIOT.2023.3318845

[33] H. Liu, Y. Wang, and K. Zhang,

"Graph neural network-assisted cluster head selection in heterogeneous

sensor networks with non-uniform energy distribution,"

IEEE Sensors J., vol. 23, no. 17, pp. 19841–19853, Sep. 2023.

doi: 10.1109/JSEN.2023.3294771

[34] R. Patel and S. Mishra,

"Lightweight Q-learning with state abstraction for energy-balanced routing

in resource-constrained WSN: Hardware evaluation on CC2530 platform,"

Pervasive Mobile Comput., vol. 91, p. 101778, Apr. 2023.

doi: 10.1016/j.pmcj.2023.101778

[35] A. K. Singh, N. Verma, and P. Kumar,

"Multi-objective reinforcement learning for joint optimization of network

lifetime and data quality in agricultural IoT sensor deployments,"

Comput. Netw., vol. 241, p. 110218, Mar. 2024.

doi: 10.1016/j.comnet.2024.110218