ISNNI: 3007-3391 HEC Category: Y

Journal of Computers and Intelligent Systems
Vol. 4, No. 4, 2025, pages 420 - 432
Journal Homepage:
https://journals.iub.edu.pk/index.php/JCIS/

A Simulation-Driven Reliability-Aware Hybrid Deep Learning Framework for Energy-Efficient Routing in Dynamic IoT Wireless Sensor Networks

Ali Raza Abbasi[1] Muhammad Waqas Pasha[2] Hajra Naseer[3] Maheen Hussain[4]
[1]The Islamia University of Bahawalpur ,Pakistan, aliabbasibwp12@gmail.com
[2]UIT University ,Pakistan, wpasha@uitu.edu.pk
[3]The Islamia University of Bahawalpur ,Pakistan, hajra128821@gmail.com
[4]UIT University ,Pakistan, maheenw98@gmail.com

ARTICLE INFO

ArticleHistory:

Received:	July	7, 2026
Revised:	August	13, 2026
Accepted:	September	18, 2026
Available Online:	September	30, 2026

Keywords:

IoT; wireless sensor networks; energy-aware routing; CNN-BiLSTM; attention; uncertainty estimation; PPO; NS-3; network lifetime.

Classification Codes:

Funding:

This research received no specific grant from any funding agency in the public or not-for-profit sector.

ABSTRACT

The Internet of Things (IoT) wireless sensor networks (WSNs) are energy constrained, so the routing strategies must meet the requirements of minimizing battery usage, guaranteeing communication delay, and link reliability when the network environment changes. In this paper, we present a simulation-based approach, namely RALiWSN, that integrates CNN-BiLSTM-attention for network-state prediction with uncertainty-aware Proximal Policy Optimization (PPO). The CNN learns local relationships between network features, the BiLSTM captures temporal dependencies and attention weights captures informative observations. Before PPO picks a feasible next hop the fusion of predicted and currently observed states is controlled by an ensemble-based uncertainty estimate. The proposed evaluation plan will be based on the traces generated using NS-3 and the results will be compared with conventional routing, energy-aware routing, PPO based on current observations and the entire RALiWSN. Complementary available WSN data are identified for external validation and reference, if feature definitions are similar. For example, this manuscript contains a reproducible, seeded synthetic pilot dataset and synthetic comparative plots to show how the reporting procedure is applied. These pilot values are intended to be illustrative only, and not NS-3 measurements. Final claims must be simulated at the packet level, and then validated statistically by the user.

OPEN ACCESS

© 2026 The authors published by JCIS. This is an Open Access Article under the Creative Common Attribution Non-Commercial 4.0

CorrespondingAuthor'sEmail:

I. Introduction

Low-power sensor networks are becoming essential to various applications, such as environmental monitoring, industrial automation, smart agriculture, and infrastructure management, for IoT deployments. The sensor nodes are usually equipped with very small batteries, small memory, limited processing power, and wireless communications that are vulnerable to interference and attenuation. This limitation creates the following major design issues: energy-aware and reliable routing.This limitation poses the following major issues in the design: energy-aware and reliable routing (Kori et al., 2025; Ojha & Gupta, 2025).

The classical protocols are of fundamental importance. The communication costs are distributed in LEACH with adaptive clustering, cluster-head rotation and local aggregation (Heinzelman et al., 2002). To minimize number of long-distance transmissions and maximize energy efficient data gathering, PEGASIS organizes nodes into a chain (Lindsey et al., 2002). Routes are built on demand, and topology changes are detected through route-request, route-reply and route-error messages in AODV (Perkins et al., 2003). Although these protocols have the important routing requirement, they do not explicitly learn the temporal relationship between residual energy, RSSI, packet delivery, queue occupancy and delay.

Machine learning can be used to help make resource management decisions that adapt to the observed network behavior. Kori et al. (2025) present a recent survey on the resource-management techniques in WSNs (RSNs), which are mainly based on machine learning, and point out that energy, reliability, latency, and resource allocation are common problems in RSNs. Similarly, Ojha and Gupta (2025) provide an overview of the changing landscape of AI use in WSNs, such as routing, deployment, localization, and synchronization. While these reviews encourage data-centric network management, they also stress the importance of a consistent evaluation in different topologies and simulations.

With sequential network observations, CNNs can learn local feature interactions, and recurrent models can learn temporal dependencies. The Long Short-Term Memory (LSTM) was later proposed to overcome the problem of the conventional recurrent learning with long temporal dependencies (Hochreiter & Schmidhuber, 1997). Bidirectional recurrent processing can utilize context in a complete observation window (Schuster & Paliwal, 1997). Attention mechanisms offer a learnt weighting of informative representations (Bahdanau et al., 2015). In this work, the BiLSTM is trained with a historical window that is only available at decision time, and no future observations are fed to the BiLSTM.

Reinforcement learning provides an alternative way of sequential routing decisions. PPO is an optimised clipped policy objective to limit policy updates and is a popular policy gradient method (Schulman et al., 2017). The recent surveys on DRL for WSN routing highlights the increasing adoption of value-based algorithms and actor-critic algorithms for energy-aware routing and the challenges of scalability, training cost, and dynamic nature of the network (Ojha & Gupta, 2025; Intelligent Energy-Aware Routing in Wireless Sensor Networks, 2026).

The primary motivation behind RALiWSN is that the predictions shouldn't be trusted equally in all conditions. If there is no RSSI measurement or the channel changes rapidly, a predictor may become unsure. Thus, based on the predicted state and the current measurements, RALiWSN fuses the state with an uncertainty-dependent confidence weight and passes the fused state to PPO. It is not just the combination of CNN, BiLSTM, attention and PPO; it is the direct linkage between prediction uncertainty and the fusion of routing-state and a conservative fallback policy.

A. Objectives and contributions

1. Develop a reproducible NS-3-based IoT-WSN simulation and trace-generation pipeline.
2. Train a CNN-BiLSTM-attention model to predict short-horizon energy, link-quality, and delay indicators.
3. Estimate predictive uncertainty and fuse predictions with current observations.
4. Train a PPO agent to select feasible next-hop nodes using energy, reliability, delay, queue, and uncertainty information.
5. Evaluate the method across node densities, traffic regimes, channel variation, and missing observations using independent random seeds.

II. Literature Review

A. Conventional WSN routing

LEACH is one of the basic hierarchical routing protocols which creates clusters and cyclically changes cluster head's responsibilities to balance energy consumption (Heinzelman et al., 2002). PEGASIS adopts a chain-based data-gathering strategy and energy metrics to minimize the number of communications (Lindsey et al., 2002). AODV is a reactive route discovery and maintenance protocol for ad hoc networks (Perkins et al., 2003). Their assumptions and goals do not match, so the comparisons have to be based on protocol-appropriate implementations of time and radio settings, as well as well-documented traffic settings.

B. Machine learning for WSN resource management

Kori et al. (2025) give an overview on machine learning and computational-intelligence based resource discovery, scheduling, allocation and utilization in WSNs. Ojha and Gupta (2025) review and analyze the advancements in AI technology and highlight three key themes: energy efficiency, adaptive operation, and network optimization. The reviews also emphasize the importance of learning based control and highlight that numerical comparisons of different papers are not reliable because of the heterogenous datasets and evaluation setups.

C. Deep learning for temporal network-state prediction

LSTM networks have been shown to learn temporal dependencies (Hochreiter & Schmidhuber, 1997) with the use of gated memory cells. Bidirectional recurrent networks process sequences forwards and backwards (Schuster & Paliwal, 1997); this is suitable for fixed historical windows but not for using future observations that were not available at inference time. In the field of machine translation, attention mechanisms have been introduced to learn weights over encoded representations (Bahdanau et al., 2015). RALiWSN uses the concepts to predict network state, and should be compared to CNN, LSTM, BiLSTM, and CNN-BiLSTM ablations.

D. Reinforcement learning for adaptive routing

PPO alternates its sampling of the environment with a series of minibatch optimization steps employing a clipped surrogate objective (Schulman et al., 2017). This will be a good policy-learning baseline for next-hop selection. However, the design of rewards, the masking of the actions, scaling of rewards, stability of training, and the ability to transfer to novel topologies must be handled explicitly. In the survey (Intelligent Energy-Aware Routing in Wireless Sensor Networks, 2026), PPO is discussed in relation to DQN, DDQN, DDPG and actor-critic methods and the dimensions of energy, lifetime, convergence and scalability are identified as important factors to consider for comparisons.

E. Research gap

The gap proposed in this paper lies in the limited integration of (i) short-horizon temporal prediction, (ii) calibrated prediction uncertainty, (iii) measurement-prediction fusion, and (iv) reliability-aware PPO routing in one reproducible WSN evaluation pipeline. This is actually a research hypothesis that can be tested and not a claim that no previous study has ever brought these ideas together. At final stage, novelty should be confirmed by systematic literature search.

III. Dataset sources and provenance

A. Primary dataset: RALiWSN-Sim

NS-3 simulations should be used to produce the primary data set for the final study, which should be developed by the authors. NS-3 is a discrete-event network simulator that has a documented energy-source and device-energy-consumption model (The ns-3 Project, n.d.). The attributes in each record should include simulation seed, topology, node identifiers, neighbor identifiers, timestamp, residual energy, RSSI, SNR, PDR, delay, queue length, packet counts, link availability, selected next hop and next-step targets.

Dataset/source	Provenance and content	Role in this study
RALiWSN-Sim (proposed)	Author-generated NS-3 traces; configuration and scripts to be released with paper	Primary train/validation/test dataset
InSecTT WSN Power Consumption Dataset	Zenodo DOI: 10.5281/zenodo.7762712; measured power-consumption traces for BLE, Thread, EPhESOS and UWB sensor-node communication	External energy-behavior reference; protocol mismatch must be acknowledged
IWCS IIoT-WSN Dataset	Zenodo DOI: 10.5281/zenodo.20602871; OMNeT++/INET IEEE 802.15.4/RPL simulation dataset; PDR, delay, throughput, energy, RSSI and security scenarios	Optional external robustness/feature reference; not a direct NS-3 benchmark
NIST O-RAN with ML in ns-3	NIST public dataset and scripts associated with ns-3 machine-learning experiments; LTE/O-RAN domain	Methodological reference for reproducible simulation-data publication, not a WSN training substitute

The datasets have been made available via the following links:Access to the datasets are available through the following links: The conditions and licenses for the re-use of the datasets are to be ascertained at downloading. Permission and license terms are to be confirmed before redistribution, the IWCS record currently includes copyright information.

B. Synthetic pilot dataset included with this manuscript

As an example of how to demonstrate the data schema and the figure generation pipeline, a synthetic dataset seeded with some real data is included as RALiWSN_Synthetic_Pilot_Dataset.csv. It consists of 8000 channel/traffic runs across 10 run identifiers, 4 node densities (50, 100, 150, 200), 4 stylized channel/traffic scenarios and 50 timesteps per run-identifier/node-density/scenario combination. It includes residual energy, RSSI, SNR, PDR, delay, queue length, congestion index and missing-observation indicators. The next-step energy, PDR and delay targets are derived through explicit stylized equations that contain random perturbations.

This is NOT a public benchmark file, NOT an NS-3 trace file, and NOT a measured file. It is added just to ensure the examples of the analysis can be reproduced and to illustrate the desired schema. It needs to be replaced with traces generated by the NS-3 for an Empirical Journal submission.

C. Data split and leakage prevention

To split by complete simulation run and random seed, NOT by randomly shuffling similar temporal windows. Normalize the parameters of the fit only on the training runs. Do validation runs for hyperparameter selection and hold back independent seeds and, ideally, an unseen density for final testing. After trace generation, report number of runs, records, nodes, packets and scenarios.

IV. Proposed methodology and architecture

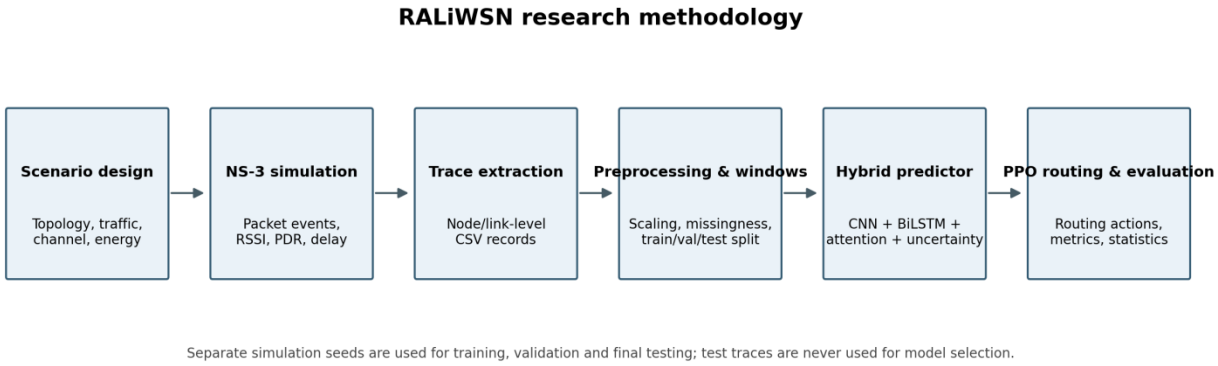


Figure 1. RALiWSN end-to-end research methodology. The primary empirical traces are intended to be generated by NS-3; the accompanying pilot CSV is synthetic and illustrative.

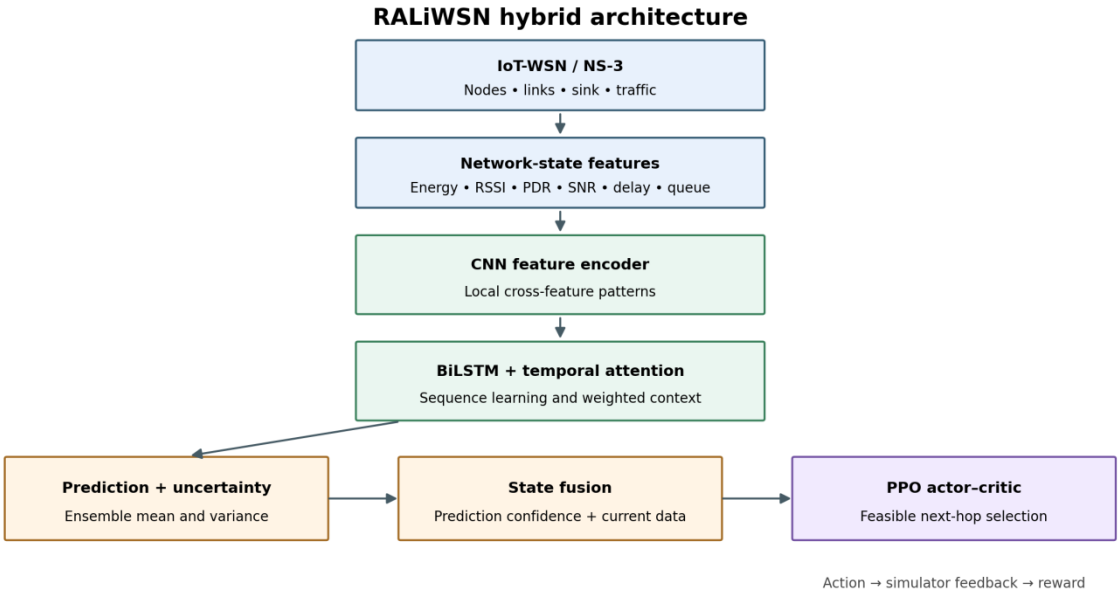


Figure 2. Proposed RALiWSN architecture: CNN-BiLSTM-attention prediction, uncertainty estimation, state fusion, and PPO next-hop selection.

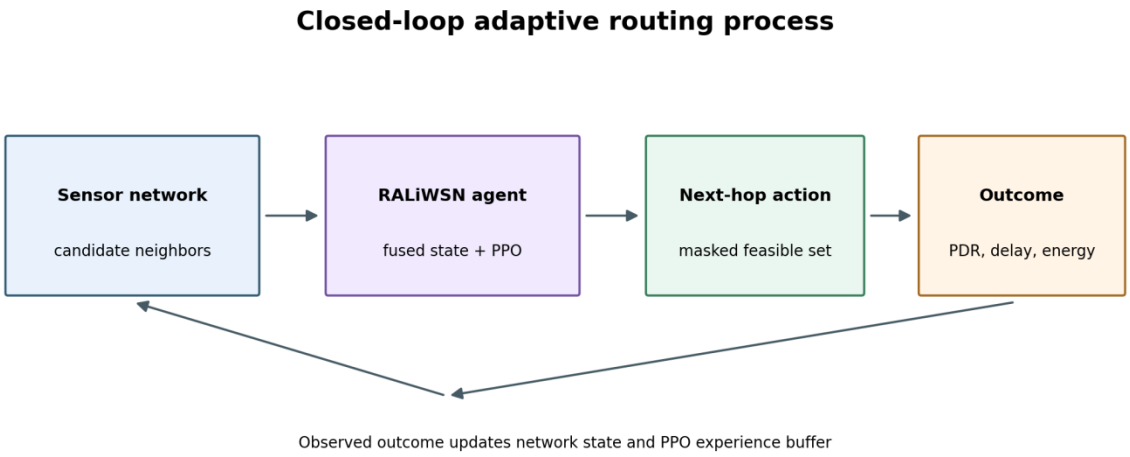


Figure 3. Closed-loop routing process showing how simulator outcomes are returned to the learning agent.

A. Network-state representation

When a node-link is decided at time t, x_t represents its observation as [E_t, RSSI_t, SNR_t, PDR_t, D_t, Q_t, C_t], where E is the node's remaining energy, RSSI is the received signal strength indicator, SNR is the signal-to-noise ratio, PDR is the percentage of data correctly received, D is the delay, Q is the node's queue size, and C is the link/connectivity indicator. Next observation is predicted based on a historical window W_t = (x_(t-T+1), ..., x_t). The candidate-neighbor features are stored per link so the policy can be used to compare neighbor forwarding actions.

B. CNN-BiLSTM-attention predictor

The local patterns in the feature sequence are extracted by a one-dimensional convolutional encoder: f(Conv1D(X; W_c) + b_c), where H_c is the local patterns. The encoded sequence is then given to a BiLSTM, with h_t^forward and h_t^backward representing the forward and backward state vectors, respectively. Attention scores are computed as e_t = v^T tanh(W h_t + b), and a_t = softmax(e_t), and the context c = Σ_t a_t h_t. A multi-output prediction head predicts all remaining energies, link-quality/PDR, and delay at the next step. The BiLSTM is bidirectional only across the historical window, and does not have access to future samples.

C. Uncertainty estimation and state fusion

Use an ensemble of K predictors or a calibrated probabilistic output. Calculate the means $\mu = (1/K)\Sigma_k y_hat_k$ and the variances $\sigma^2 = (1/K)\Sigma_k(y_hat_k-\mu)^2$ for ensemble predictions y_hat_k. Scale uncertainty u into [0,1] by means of validation derived scaling. Now, if the confidence $c = \exp(-\lambda u)$, then blend the predicted and measured state: s_fused = c μ + (1-c)x_current. Apply masks on features where observations are missing. When the state is below a threshold value selected by the validation, initiate a conservative measured-state routing rule.

D. PPO routing policy

The PPO state includes fused candidate-neighbor features, current node energy, destination-relative information, queue state, and uncertainty. The action is a viable next-hop identifier, unavailable and dead neighbors are masked. PPO is based on the objective extracted from Schulman et al. (2017) and then clipped. The policy is learned in training, the test runs are separated from training and validation.

E. Objective and reward

Define a normalized routing cost function $J = \alpha E + \beta D + \gamma(1-PDR) + \delta O$, where E is the normalized energy cost, D is the normalized delay, PDR is the estimated delivery probability and O is the normalized control overhead. The weights are all non-negative and add up to 1. The reward is $r_t = -J_t - P_t$ and P_t penalizes invalid forwarding, loops, disconnected paths, constraint violations. Report the selected weights and sensitivity analysis – do not tune the weights on the final test set.

V. Simulation configuration

Parameter	Proposed value / reporting requirement
Simulator	NS-3; record exact release, build options and modules
Deployment region	200 m × 200 m (initial configuration)
Node counts	50, 100, 150, 200
Sink/base station	One fixed sink; report coordinates
Initial energy	2 J/node (initial design assumption)
Simulation duration	1,000 s per run (initial design assumption)
MAC/PHY	IEEE 802.15.4-compatible model; specify exact NS-3 model and PHY settings
Packet payload	100 bytes (initial assumption)
Traffic	Periodic, event-driven and bursty
Channel cases	Stable, time-varying RSSI, congestion, intermittent links
Independent repetitions	At least 10 seeds per configuration
Radio parameters	Report Tx power, Rx sensitivity, data rate, current draw, path-loss and propagation model
Energy model	NS-3 energy source and device energy model; report all current/power parameters

The NS-3 energy framework provides energy-source and device-energy-consumption abstractions (The ns-3 Project, n.d.). A final paper must report exact device/radio parameters rather than assuming that generic values represent a specific hardware platform.

A. Learning hyperparameters

Component	Initial configuration (tune on validation data)
CNN	64 filters; kernel size 3
BiLSTM	2 layers; 128 hidden units per direction
Attention	64-dimensional attention layer
Dropout	0.30
Optimizer	Adam; initial learning rate 0.001
Batch size / epochs	64 / maximum 100; early stopping patience 10
Temporal windows	Compare T = 5, 10, 20, 30

Uncertainty ensemble	K = 5 initial models; calibrate on validation set
PPO policy	MLP 256-128; learning rate 0.0003
PPO settings	γ=0.99; GAE λ=0.95; clip=0.20; entropy coefficient=0.01

VI. Illustrative synthetic pilot results

The included seeded synthetic pilot scenario generator is used to produce the following tables and graphs. They are included to illustrate the reporting of results and the structure of figures. They are neither outputs of NS-3, nor real experimental measurements and cannot substantiate NS-3 superiority in RALiWSN. Assumed protocol-level means were specified for demonstration purposes, and disrupted over ten seeded replicates. Use Simulator output values and figures in place of values before submission.

Method	**Energy (J/node)**	**PDR (%)**	**Delay (ms)**	**FND (s)**	**Overhead ratio**
LEACH	0.821 ± 0.019	89.43 ± 0.74	41.16 ± 2.33	604.5 ± 27.2	1.012 ± 0.031
PEGASIS	0.709 ± 0.032	91.09 ± 0.51	58.06 ± 2.05	687.8 ± 24.6	0.833 ± 0.040
AODV	1.045 ± 0.023	92.61 ± 1.03	34.46 ± 1.39	561.3 ± 20.2	1.188 ± 0.040
Energy-aware	0.762 ± 0.028	92.18 ± 0.47	38.60 ± 2.08	696.8 ± 24.7	0.890 ± 0.029
PPO-current	0.681 ± 0.030	93.93 ± 1.06	31.35 ± 1.47	766.2 ± 19.5	0.771 ± 0.025
RALiWSN	0.616 ± 0.016	94.91 ± 0.69	28.48 ± 1.97	825.4 ± 15.8	0.705 ± 0.034

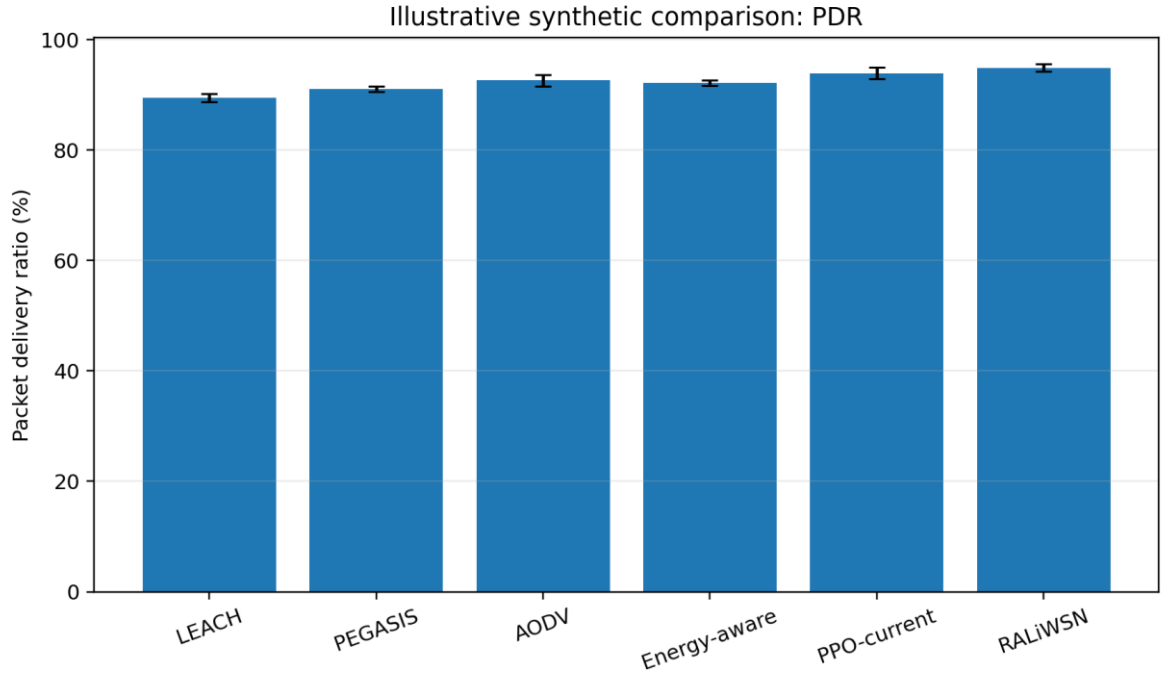


Figure 4. Illustrative synthetic packet-delivery ratio across six routing configurations (mean ± standard deviation, n=10 generated replicates).

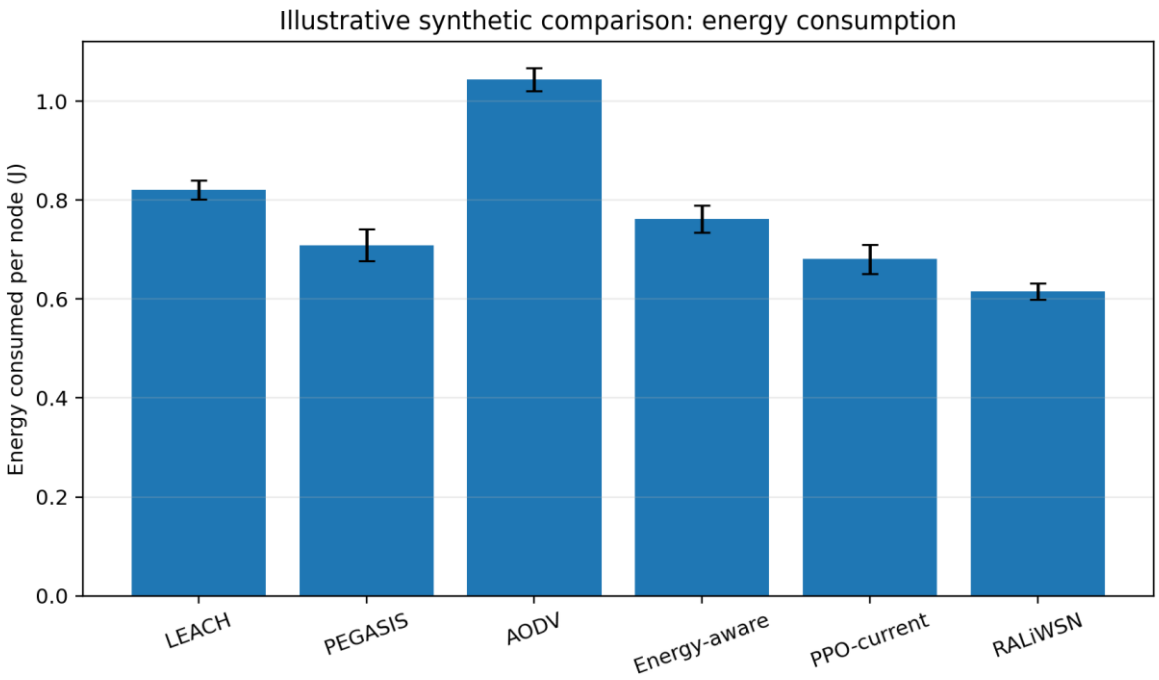


Figure 5. Illustrative synthetic energy consumption per node (mean ± standard deviation, n=10 generated replicates).

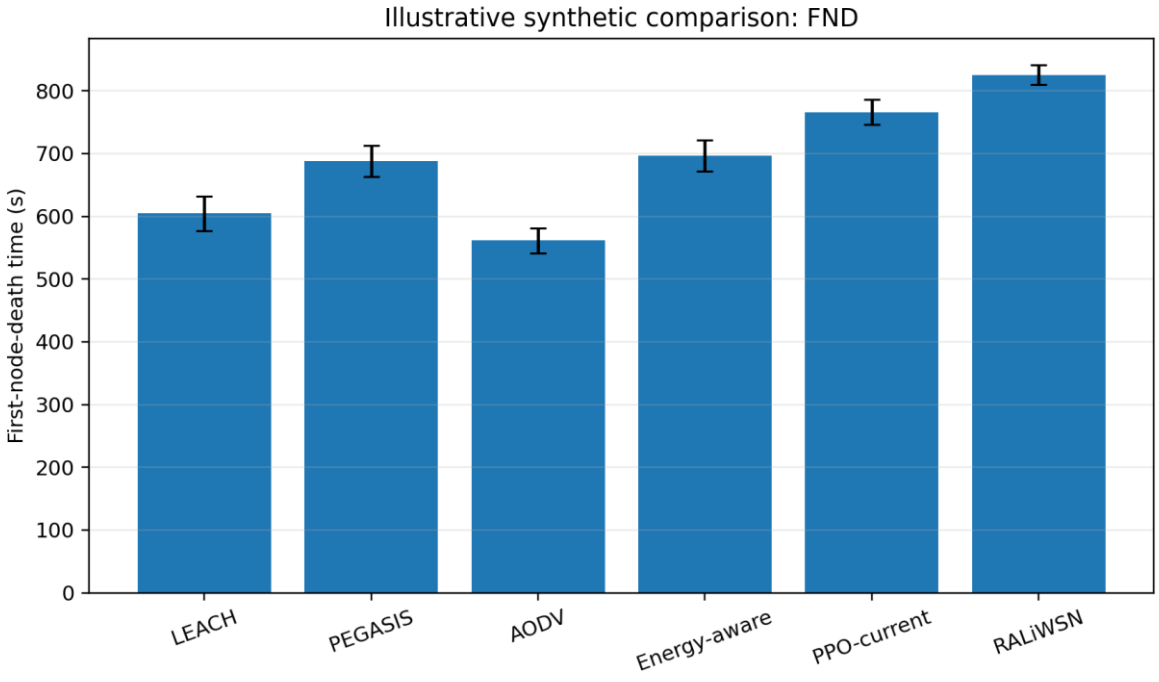


Figure 6. Illustrative synthetic first-node-death time (mean ± standard deviation, n=10 generated replicates).

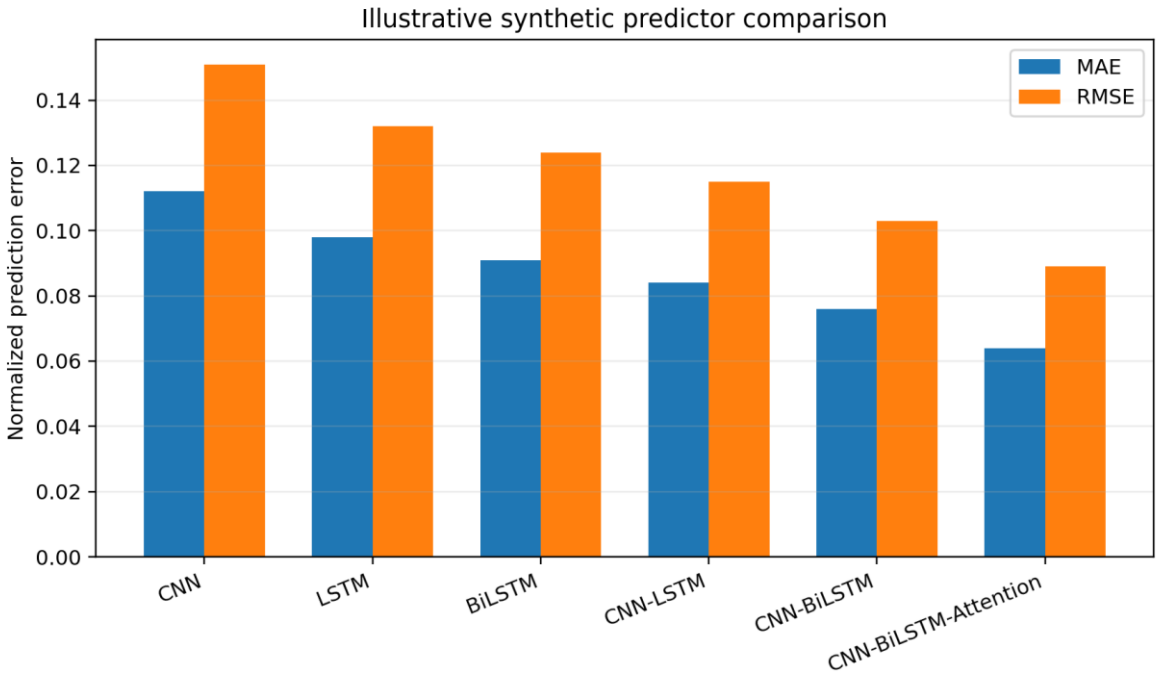


Figure 7. Illustrative synthetic prediction-error comparison. Values are generated for figure demonstration and are not trained-model measurements.

A. Interpretation limits

The means of the scenario are predefined and the equations are stylized in the synthetic pilot generator. It does not model the packet-event resolution simulation of packet queues, MAC contention, route discovery, retransmission, PHY propagation, or battery state transitions. These numbers should not be referred to as NS-3 results, and not be used as evidence of the superiority of RALiWSN over baselines. They are only meant to serve as a "validation" of the manuscript's reporting layout, and a set of sample CSV/plot artifacts.

B. Required final empirical result tables

Metric	Required final reporting
Energy	Per-node and network-total energy consumed; mean ± SD and 95% CI
PDR	Successfully received / transmitted data packets; specify whether retransmissions count
Delay	Mean and percentile end-to-end delay; define packet inclusion
Network lifetime	FND, HND, LND with exact death criterion
Throughput	Delivered application payload bits / observation interval
Overhead	Control bytes and control packets separately
Prediction	MAE, RMSE and calibration for each predicted target
Robustness	Performance by missingness level and channel regime

VII. Comparative and ablation evaluation

Compare the performance of LEACH, PEGASIS (as supported by the selected topology and stack), an energy-aware shortest-path baseline, PPO based on the current observations, and the

full RALiWSN system. The differences in implementation of the protocol should be documented. Never compare an IEEE 802.15.4 deployment with a different Wi-Fi deployment when you don't explicitly indicate the type of technology mismatch.

The following ablation configuration should be implemented: CNN only; LSTM/BiLSTM only; CNN-BiLSTM; CNN-BiLSTM-attention; predictor plus PPO without uncertainty; and complete uncertainty-aware RALiWSN. Compare all configuration with the same seeds, traffic, topologies and stopping conditions.

For the final configuration use at least 10 independent seeds. Report the mean, standard deviation, 95% confidence intervals, and effect sizes, as well as an appropriate paired (or unpaired) statistical test. Adjust for multiple comparisons as needed. Test the generalization of at least one unseen topology or node density.

VIII. Discussion

RALiWSN is designed to combine predictive representation learning with adaptive routing, rather than the prediction task being an isolated one. It is designed to minimize the dependence on predictions in the absence of or when observations are unstable, with its uncertainty-aware fusion. This benefit, however, has to be shown by comparing the full RALiWSN with a PPO policy with the same other parameters but without prediction uncertainty fusion.

Computational placement is the key trade-off in principal deployments. It is not always possible to train the ensemble predictor and PPO policy on every sensor node with limited battery capacities. A practical design could be implemented by having a central trainer and deploying a compact inference unit at the gateway or edge node, with sensor nodes sharing a compact link and energy summary. These summaries and the inference latency need to be measured with respect to communication cost.

A limitation is the simulation to reality gap. External energy measurements (e.g., InSecTT data) can be used to give context to an energy-model calibration, but BLE, Thread, EPhESOS and UWB measurements are not interchangeable with an IEEE 802.15.4 configuration in NS-3. Similarly, the IWCS dataset is generated with OMNeT++/INET and RPL, and should be considered as an external reference only, not as one that can be added without even thinking about it to the NS-3 records.

IX. Conclusion and future work

In this manuscript, RALiWSN, a hybrid learning architecture with consideration of reliability is proposed to facilitate energy-efficient routing in dynamic IoT-WSNs. The framework proposed in the paper integrates CNN-based local feature extraction, BiLSTM temporal modeling and attention, uncertainty estimation and PPO-based next-hop selection. It lays out a methodology for performing a simulation, a plan for provenance of datasets, baseline comparison, ablation experiments and evaluation metrics. Only a synthetic pilot dataset was provided to illustrate the data and reporting pipeline. The main empirical results still need to be validated with real NS-3 simulations and different random seeds.

Future work will make use of the full NS-3/Python interface, publish configuration scripts and trace generation code, calibrate the energy model with an appropriate hardware platform, and test the

routing policy on a small physical WSN testbed. Some other extensions are multi-agent PPO, federated learning, energy harvesting, adaptive duty cycling and graph neural network representations.

References

Bahdanau, D., Cho, K., & Bengio, Y. (2015). Neural machine translation by jointly learning to align and translate. In Proceedings of the 3rd International Conference on Learning Representations (ICLR). https://arxiv.org/abs/1409.0473

Heinzelman, W. B., Chandrakasan, A. P., & Balakrishnan, H. (2002). An application-specific protocol architecture for wireless microsensor networks. IEEE Transactions on Wireless Communications, 1(4), 660–670. https://doi.org/10.1109/TWC.2002.804190

Hochreiter, S., & Schmidhuber, J. (1997). Long short-term memory. Neural Computation, 9(8), 1735–1780. https://doi.org/10.1162/neco.1997.9.8.1735

Karoliny, J., Hörmann, L. B., & Peterseil, P. (2023). InSecTT WSN Power Consumption Dataset (Version 1.0) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.7762712

Kori, G. S., Kakkasageri, M. S., Chanal, P. M., Pujar, R. S., & Telsang, V. (2025). Wireless sensor networks and machine learning centric resource management schemes: A survey. Ad Hoc Networks, 167, 103698. https://www.sciencedirect.com/science/article/pii/S1570870524003093

Lindsey, S., Raghavendra, C. S., & Sivalingam, K. M. (2002). Data gathering algorithms in sensor networks using energy metrics. IEEE Transactions on Parallel and Distributed Systems, 13(9), 924–935. https://doi.org/10.1109/TPDS.2002.1036066

Ló Nunes Sebastião, O., Almeida, F., & Rady, J. (2026). IWCS-Dataset: Simulation dataset for cyberattack analysis in IIoT wireless sensor networks (Version 1.0) [Data set]. Zenodo. https://doi.org/10.5281/zenodo.20602871

Ojha, A., & Gupta, B. (2025). Evolving landscape of wireless sensor networks: A survey of trends, timelines, and future perspectives. Discover Applied Sciences, 7, Article 825. https://doi.org/10.1007/s42452-025-07070-6

Perkins, C., Belding-Royer, E., & Das, S. (2003). Ad hoc on-demand distance vector (AODV) routing (RFC 3561). RFC Editor. https://www.rfc-editor.org/rfc/rfc3561

Schulman, J., Wolski, F., Dhariwal, P., Radford, A., & Klimov, O. (2017). Proximal policy optimization algorithms. arXiv:1707.06347. https://doi.org/10.48550/arXiv.1707.06347

Schuster, M., & Paliwal, K. K. (1997). Bidirectional recurrent neural networks. IEEE Transactions on Signal Processing, 45(11), 2673–2681. https://doi.org/10.1109/78.650093

The ns-3 Project. (n.d.). Energy framework. https://www2.nsnam.org/docs/models/html/energy.html

National Institute of Standards and Technology. (2025). O-RAN with machine learning in ns-3 [Data set]. Data.gov. https://catalog.data.gov/dataset/o-ran-with-machine-learning-in-ns-3

Intelligent energy-aware routing in wireless sensor networks: A survey of deep reinforcement learning approaches. (2026). International Journal of Communication Systems. https://doi.org/10.1002/dac.70523

Appendix A. Synthetic pilot dataset generation and reproducibility

The included CSV was generated with NumPy's default_rng seed 20261114. The generator loops over 10 run identifiers, four node densities, four stylized scenarios, and 50 time steps, yielding 8,000 rows. The data generator uses explicit, simplified equations for residual-energy drift, RSSI

fluctuation, congestion, packet loss, and next-step targets. It is a demonstration artifact, not a network simulator.

Files supplied: RALiWSN_Synthetic_Pilot_Dataset.csv; RALiWSN_Illustrative_Simulation_Results.csv; Figure1_Methodology.png; Figure2_Architecture.png; Figure3_RoutingLoop.png; Figure4_PDR.png; Figure5_Energy.png; Figure6_FND.png; Figure7_Prediction.png.