EDL-IoT-Net: A Hybrid DQN–TCN–ESN Framework for Energy-Efficient, Trust-Aware, and Congestion-Adaptive Routing in IoT Wireless Sensor Networks

N. Merrin Prasanna [1, 2], Shashi Kant Gupta [3, 4]

[1]Lincoln University College, Petaling Jaya, Selangor, Malaysia. Email: nagadasari.merrin@gmail.com.

[2]Dept. of ECE, Annamacharya University, Rajampet, Andhra Pradesh, India. Email: nagadasari.merrin@gmail.com.

[3]Lincoln University College, Petaling Jaya, Selangor Darul Ehsan-47301, Malaysia. Email: raj2008enator@gmail.com , shashigupta@lincoln.edu.my

[4]Centre for Research Impact & Outcome, Chitkara University Institute of Engineering and Technology. Chitkara University, Rajpura, 140401, Punjab, India. Email: raj2008enator@gmail.com, ORCID: 0000-0001-6587-5607

Abstract

The increasing growth of Internet of Things (IoT)-enabled wireless sensor networks (WSNs) has brought up critical issues such as energy consumption, congestion management, trust-aware security, and reliable packet transmission in dynamic and resource-constrained network contexts. Traditional static routing systems cannot adapt to dynamic traffic patterns and uncertain node behaviors, leading to early energy depletion and reduced communication dependability. In this paper, we propose a new hybrid intelligent routing framework, called EDL-IoT-Net, that combines Deep Q-Network (DQN) reinforcement learning with Temporal Convolutional Networks (TCN) for long-range traffic pattern modeling and Echo State Networks (ESN) for fast nonlinear reservoir adaptation. The framework models five critical network state parameters to steer multi-objective routing optimization with a reward function that maximizes energy efficiency and trust, and minimizes congestion concurrently. Experimental validation was performed on a simulated IoT-WSN dataset of 8,000 state samples using a 60 node, 100m x 100m sensing area with a central sink node. EDL-IoT-Net achieves 99.8750% classification accuracy, 99.9875% precision, 99.5875% recall, and 99.7872% macro F1-score, outperforming all baseline models considered. At network level, EDL-IoT-Net increases network lifetime to 1,100 rounds (57.1% improvement over LEACH), decrease energy consumption to 0.46 (47.1% reduction), increase throughput to 103 kbps, achieve the packet delivery ratio of 98.73%, reduce congestion to 0.31, and reduce routing delay to 70 ms. Five-fold stratified cross validation indicates a mean accuracy of 99.77% validating robust generalisation across unexplored network circumstances.

Keywords: *IoT-WSN; Deep Q-Network; Temporal Convolutional Networks; Echo State Networks; Energy-Efficient Routing; Trust-Aware Networking; Congestion Control; Reinforcement Learning; EDL-IoT-Net*

1. Introduction

Artificial intelligence and IoT-enabled wireless sensor networks (IoT-WSNs) are increasingly used for real-time analytics and industrial automation, but challenges related to security, adaptability, energy constraints, and malicious attacks remain significant [1,2]. The dynamic nature of IoT networks makes traditional static routing protocols inefficient under changing traffic conditions, leading to congestion, increased latency, and reduced packet delivery performance [3,4]. Effective routing requires reinforcement learning-based strategies that consider energy consumption, data transmission efficiency, and congestion-aware traffic management to improve network reliability [5,6].

Recent approaches using trust models, software-defined networking, and machine learning enable adaptive routing by evaluating node behavior, traffic patterns, and network conditions [8–10]. This paper proposes EDL-IoT-Net, a hybrid routing framework integrating DQN, TCN, and ESN with a multi-objective reward function based on residual energy, node trust, and congestion, along with a trust-aware model and extensive evaluation against existing routing methods.

2. Literature Review

The research on intelligent routing in IoT-WSNs has been progressing rapidly with traditional machine learning, deep reinforcement learning, trust management, and congestion-aware framework. Table 1 summarizes twenty exemplary works reviewed for this study. It captures the method employed, dataset or simulation environment, important reported result and the primary drawback prompting the present effort.

Table 1. Literature Survey — IoT-WSN Intelligent Routing and Energy-Efficient Frameworks (2021-2025)

Ref	Year	Authors	Method	Dataset/Environ.	Key Result	Limitation
[1]	2025	Venugopal et al.	AI-driven DL energy optimization	IoT-WSN simulation	Network lifetime ↑	No congestion-awareness
[2]	2025	Purushotham & Muddana	Trust-GAEP-Net (GNN+AlexNet+EPSO)	IoT-WSN 60 nodes	Detection 99.1%	High computational overhead
[3]	2021	Farag & Stefanovic	Q-learning congestion-aware RPL	RPL simulated	PDR & delay improved	No trust integration
[4]	2025	Shafiuddin & Krishna	TrustRIDR-Net (RFO+DRL)	IoT 200x200m,100 nodes	PDR 98.9%	No temporal modeling
[5]	2024	Bhimshetty & Agughasi	Deep Q-Network routing WSN	WSN simulation	Lifetime improved	Single reward metric
[6]	2024	Sitharamulu et al.	ML routing optimization IoT	WSN OPNET	Energy conservation	Fixed routing policy
[8]	2025	Andanaiah & Rao	Meta-RL+GAT SDN-WSN	SDN-IoT Mininet	Throughput+latency	No reservoir dynamics
[10]	2021	Frikha et al.	RL survey wireless IoT	Multiple WSN	Survey insights	No unified framework
[11]	2024	Bhavya et al.	R-OPTICS+vEBT congestion-free SIoT	SIoT simulation	Congestion reduced	No DRL adaptation
[12]	2025	Alem et al.	RIMS-RPL mobility-support RL	IIoT RPL network	Packet delivery ↑	No trust metric
[13]	2023	Zahedy et al.	RI-RPL Q-learning routing	RPL simulation	QoS improved	Static trust assumption
[14]	2023	Subramaniam et al.	RL for QoS IoT delay	IoT simulation	Delay minimized	No energy reward

[15]	2025	Wang et al.	GATR graph attention DRL RPL	RPL optimization	Routing quality ↑	No reservoir computing
[16]	2025	Soltani et al.	Multi-agent RL energy-efficient WSN	WSN simulation	Energy savings	Scalability concerns
[17]	2021	Kundaliya & Lobiyal	Q-learning network lifetime WSN	WSN scenario	Lifetime extended	No congestion modeling
[18]	2022	Keum & Ko	Trust-based Q-learning WSN	Mission-critical WSN	Reliability ↑	No temporal features
[21]	2024	Dey & Ghosh	iTRPL multi-agent RL	RPL networks	Trust & routing ↑	Limited scalability
[22]	2020	Xu et al.	Voting-based MARL IoT	IoT environment	Resource management	High communication cost
[26]	2025	Robertson & Moreau	DRL routing high-performance WSN	WSN simulation	Throughput ↑	No trust modeling
[28]	2025	Gao et al.	Lightweight Double-DQN IoT	Industrial IoT	Energy efficient	Single objective only

As can be seen from the survey of Table 1, the present works handle one or two issues of the three main challenges of IoT-WSN, i.e., energy efficiency, trust security and congestion control in isolation. TrustRIDR-Net [4] and GATR [15] are studies offering DRL techniques that achieve strong trust and routing performance, but do not have the ability of temporal feature extraction to model sequential traffic dynamics and predict congestion before it develops into packet loss. On the other hand, the congestion-based approaches, such as R-OPTICS [11], and Q-learning RPL variants [13, 17], do not take into account the energy-aware trust modeling and are vulnerable to selfish node attacks that can drain the cooperative nodes disproportionately.

3. Proposed Methodology

3.1 Network Topology and Distance Model

The IoT-WSN is regarded as a dynamic heterogeneous environment with 60 sensor nodes randomly deployed inside a sensing area of 100m x 100m and a sink node is fixed at the geometric center. The quality of signal transmission and eligibility for communication is defined by the Euclidean distance between two nodes:

$$d_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \qquad (1)$$

where $(y_i)$and (y_j)denote the two-dimensional Cartesian coordinates of nodes i and j, respectively, and d_{ij} represents the Euclidean separation between nodes used to determine communication link eligibility based on transmission range threshold R_{tx}.

$$A_{ij} = \{1, if\ d_{ij} \leq R_{tx}\ 0, otherwise \qquad (2)$$

whereas A_{ij} is the binary adjacency indicator between nodes i and j;R_{tx} is the maximum transmission range; $A_{ij} = 1$ denotes an active communication link, while $A_{ij} = 0$ indicates that nodes are out of range and cannot communicate directly.

3.2 Energy Model

Each sensor node commences operation with a predetermined initial energy reserve that depletes through data transmission, reception, sensing, and idle listening. Continuous monitoring of residual energy is essential for energy-aware routing:

$$E_i^{res}(t) = E_i^{init} - \sum_k \left(E_i^{tx}(k) + E_i^{rx}(k) + E_i^{idle}(k)\right) \quad (3)$$

whereas E_i^{init}denotes the initial battery energy of node i; $E_i^{tx}(k)$, $E_i^{rx}(k)$, and $E_i^{idle}(k)$represent the energy consumed in transmission, reception, and idle listening, respectively, in round k; the summation accumulates all energy expenditure from round 1 to the current round t.

$$E_i^{tx}(l,d) = l \cdot E_{elec} + l \cdot \varepsilon_{amp} \cdot d^{\alpha} \quad (4)$$

whereas l denotes the packet length (in bits); E_{elec} represents the energy dissipated per bit in the electronic circuitry; ε_{amp} is the transmitter amplifier energy coefficient; d is the transmission distance; α is the path-loss exponent (typically 2 for free-space or 4 for multi-path fading channels).

$$E_i^{rx}(l) = l \cdot E_{elec} \quad (5)$$

whereas l is the received packet length in bits; E_{elec} is the electronic processing energy per bit at the receiver; reception energy is distance-independent but contributes to battery depletion and must be accounted for in network lifetime computation.

$$\Delta E_i(t) = E_i^{res}(t-1) - E_i^{res}(t) \quad (6)$$

whereas $\Delta E_i(t)$ is the instantaneous energy depletion rate of node i at round t; $E_i^{res}(t-1)$ and $E_i^{res}(t)$ are the residual energies at consecutive rounds; a higher ΔE_i signals imbalanced routing load, informing the DQN reward function to redistribute traffic.

3.3 Trust Model

$$\boldsymbol{T_{ij}^{direct}(t) = \frac{S_{ij}(t)}{S_{ij}(t)+F_{ij}(t)+\varepsilon}} \quad (7)$$

whereas $S_{ij}(t)$ is the cumulative count of successful packet forwarding events observed by node i from neighbor j; $F_{ij}(t)$ counts failed or dropped forwarding events; epsilon is a small smoothing constant preventing division by zero; T_{ij}^{direct} approaches 1.0 for fully cooperative nodes and falls toward 0 for malicious or selfish nodes.

$$T_{ij}^{indirect}(t) = \sum_{k \in N_i} w_{ik}\, T_{kj}(t) \quad (8)$$

whereas w_{ik} are recommendation weights reflecting the credibility of intermediary node k; N_i is the neighbourhood set of node i; $T_{kj}(t)$ is the trust score that node k reports for node j; nodes with higher established credibility receive larger w_{ik} weights to prevent colluding nodes from inflating trust values.

$$T_{ij}(t) = \lambda\, T_{ij}^{direct}(t) + (1-\lambda)\, T_{ij}^{indirect}(t) \quad (9)$$

whereas λ is a weighting coefficient in [0, 1] balancing direct and indirect trust contributions; higher lambda weights locally-observed behavior more heavily; lower lambda relies more on network-level recommendations; the composite trust T_{ij} determines whether node j is eligible as a routing intermediary.

$$T_{ij}(t) = T_{ij}(t-1)\, e^{-\mu \Delta t} \quad (10)$$

whereas μ is the temporal trust decay constant controlling the rate at which inactive trust information becomes stale; Δt is the elapsed time since the last observed forwarding event between nodes i and j; this exponential decay prevents outdated trust values from persisting and maintaining routing eligibility for long-inactive nodes.

3.4 Congestion and State Vector

$$Q_i^{ratio}(t) = \frac{Q_i(t)}{Q_i^{max}} \qquad (11)$$

whereas $Q_i(t)$ is the current queue occupancy (number of packets buffered) at node i; Q_i^{max} is the maximum buffer capacity; the normalized ratio approaches 1.0 under heavy load, signaling imminent packet loss and triggering rerouting in the DQN policy.

$$D_i(t) = \frac{1}{P_i(t)} \sum_k \left(t_{recv,k} - t_{send,k}\right) \qquad (12)$$

whereas $P_i(t)$ is the total number of packets processed by node i in round t; $t_{recv,k}$ and $t_{send,k}$ are the reception and transmission timestamps for packet k; $D_i(t)$ estimates average per-packet queuing delay and feeds into the composite congestion index.

$$C_i(t) = \theta_1 \, Q_i^{ratio}(t) + \theta_2 \, \frac{D_i(t)}{D_{max}} \qquad (13)$$

whereas θ_1 and θ_2 are weighting coefficients ($\theta_1 + \theta_2 = 1$) controlling sensitivity toward buffer occupancy versus delay; D_{max} is the maximum acceptable delay; $C_i(t)$ provides a unified congestion index in [0, 1] combining both queue and delay information.

$$x_i^{norm}(t) = \frac{x_i(t) - x_i^{min}}{x_i^{max} - x_i^{min}} \qquad (14)$$

whereas $x_i(t)$ is the raw feature value for node i at time t; x_i^{min} and x_i^{max} are the observed minimum and maximum bounds across the training dataset; normalized features ensure stable gradient updates and prevent features with larger magnitudes from dominating smaller-scale ones.

$$x_i^{smooth}(t) = \alpha \, x_i(t) + (1 - \alpha) \, x_i^{smooth}(t-1) \qquad (15)$$

whereas alpha is the smoothing coefficient ($0 < \alpha \leq 1$.) controlling the degree of exponential moving-average filtering applied to each feature; higher alpha gives more weight to the current observation, lower alpha produces smoother but slower-responding feature trajectories.

$$s_t = [\, PLR_i] \qquad (16)$$

whereas s_t is the consolidated state vector at time step t fed to the EDL-IoT-Net framework; E_i^{res} is residual energy; ΔE_i is energy decay rate; T_{ij} is composite trust score; C_i is congestion index; D_i is average packet delay; PLR_i is packet loss rate; all features are pre-processed via Equations (14)-(15) before entering the DQN+TCN+ESN pipeline.

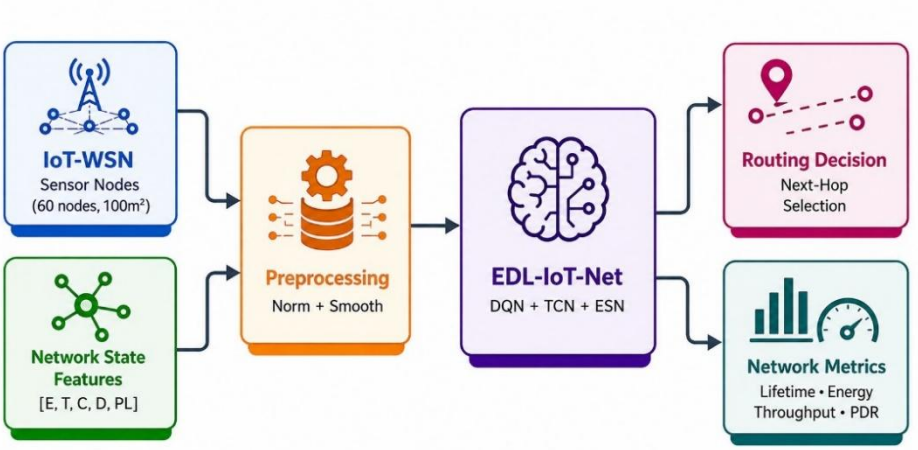


Figure 1: EDL-IoT-Net Framework Pipeline

Figure 1 illustrates the complete EDL-IoT-Net framework, including state collection, preprocessing, routing decisions, and performance evaluation. The 60-node network state is normalized and smoothed using Equations (14) and (15) to generate sts_tst, which is processed by the DQN-TCN-ESN routing agent for optimal next-hop selection.

4. Architecture of the Proposed EDL-IoT-Net

EDL-IoT-Net integrates DQN, TCN, and ESN into a three-branch hybrid routing framework for adaptive IoT-WSN optimization. Unlike single-component approaches, it combines reinforcement learning, temporal feature extraction, and nonlinear reservoir-based adaptation through a state fusion module. The fused state representation St=concat(st,Ft,Ht)S_t=\mathrm{concat}(s_t,F_t,H_t)St =concat(st,Ft,Ht) is provided to the DQN for selecting optimal routing actions.

$$Q(s_t, a_t) \leftarrow Q(s_t, a_t) + \eta[r_t + \gamma max_a\, Q(s_{t+1}, a) - Q(s_t, a_t)] \quad (17)$$

whereas $Q(s_t, a_t)$ is the current Q-value estimate for state-action pair $Q(s_t, a_t)$; eta is the learning rate controlling the magnitude of each Q-value update step; $r_t$is the immediate reward received at time t; gamma is the discount factor weighting future rewards; $max_a\, Q(s_{t+1}, a)$ is the maximum Q-value achievable from the successor state s_{t+1}.

$$r_t = \beta_1 \left(\frac{E_i^{res}}{E_{max}}\right) + \beta_2 T_{ij} - \beta_3 C_i \quad (18)$$

whereas r_t is the multi-objective reward signal at time step t; $\beta_1, \beta_2\ and\ \beta_3$ are importance coefficients ($\beta_1 + \beta_2 + \beta_3$ = 1) balancing energy conservation, trust maximization, and congestion minimization respectively; E_{max} is the initial maximum energy; higher r_t drives the DQN toward energy-rich, trustworthy, and uncongested routing paths.

$$F(t) = \sum_{k=0}^{K-1} \quad w_k\, x(t - d\, k) \quad (19)$$

whereas F(t) is the temporal feature output of the TCN dilated convolution at time step t; w_k are learnable convolutional kernel weights of size K; $x(t - d\, k)$ are the past network state observations sampled at dilation factor d; larger d values extend the effective receptive field, capturing long-range traffic dependencies without increasing kernel size.

$$h(t) = (1 - \rho)\, h(t-1) + \rho\, tanh(W_{in}\, u(t) + W\, h(t-1)) \quad (20)$$

whereas h(t) is the ESN reservoir state vector at time t; rho is the spectral scaling factor controlling the leaking rate of reservoir memory; W_{in} is the fixed input-to-reservoir weight matrix; W is the fixed

reservoir-to-reservoir recurrent weight matrix with spectral radius < 1 ensuring echo state property; u(t) is the current input vector; tanh is the nonlinear activation providing rich reservoir dynamics.

$$min_{i\in N}\left(\frac{\omega_1}{E_i^{res}} + \omega_2(T_{ij}) + \omega_3 C_i - \omega_4 R_i(t)\right) \qquad (21)$$

whereas N is the set of candidate next-hop nodes; $\omega_1$through ω_4 are adaptive importance coefficients weighting energy depletion cost, trust deficit, congestion penalty, and ensemble routing score reward respectively; $R_i(t)$ is the EDL-IoT-Net composite output score combining DQN, TCN, and ESN outputs; minimizing this objective yields the optimal next-hop selection under all three competing constraints simultaneously.

$$L(\theta) = E[(y - Q(S_t, a_t; \theta))^2] \qquad (22)$$

whereas L(theta) is the DQN temporal-difference loss function; theta represents all trainable DQN parameters $y = r_t + \gamma max_a\, Q(S_{t+1}, a; \theta_{target})$ is the target value computed from the fixed target network with parameters θ_{target} updated every C steps; E[.] denotes expectation over mini-batches sampled from the replay buffer M.

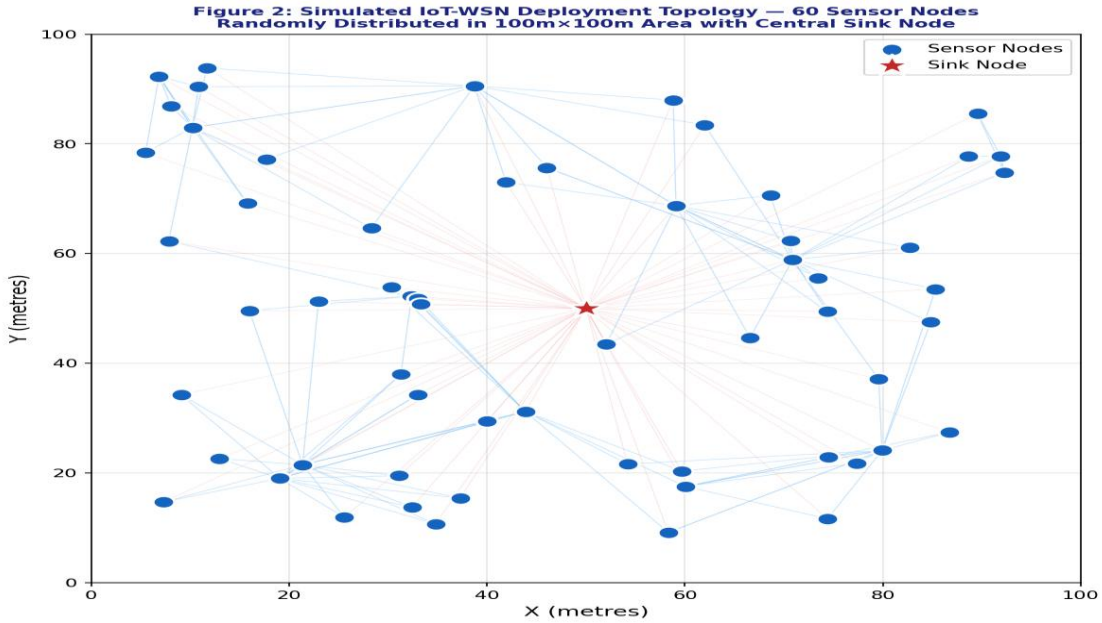


Figure 2: Simulated IoT-WSN Deployment Topology — 60 Sensor Nodes Randomly Distributed in 100m x 100m Sensing Area with Central Sink Node and Multi-Hop Communication Links

Figure 2 illustrates the simulated 60-node IoT-WSN deployment over a 100 m × 100 m area with a central sink node and multi-hop communication links. The topology supports trust and congestion modeling by capturing energy variations, relay-node bottlenecks, and routing challenges across different node locations. The same environment is used for evaluating network lifetime, throughput, PDR, energy consumption, congestion rate, and routing delay through the EDL-IoT-Net state vector sts_tst.

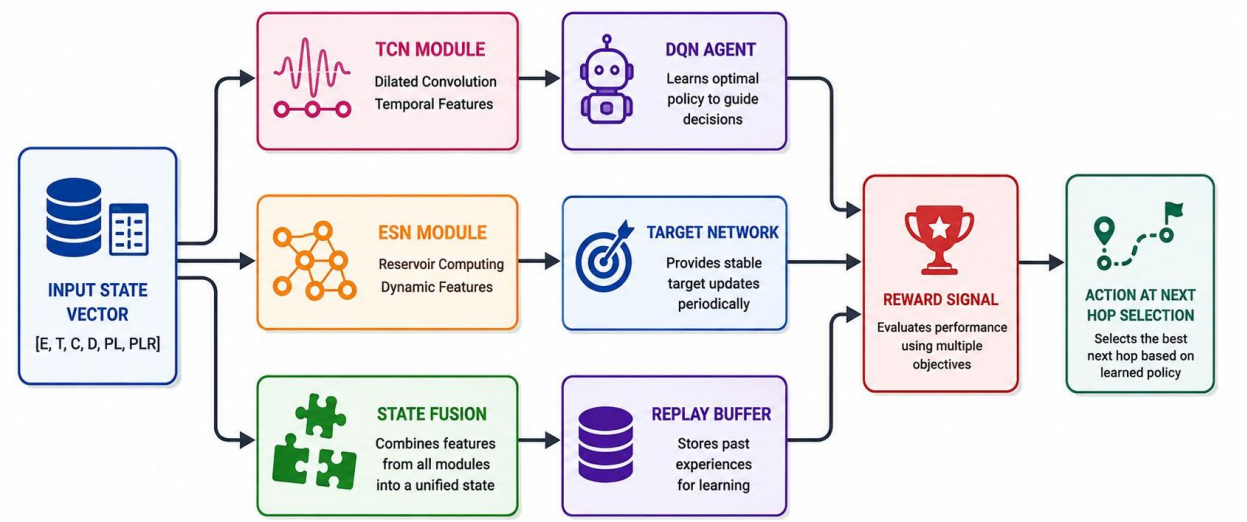


Figure 3: Architecture of EDL-IoT-Net

Figure 3 presents the internal architecture of EDL-IoT-Net, where the six-dimensional state vector sts_tst is processed through TCN, ESN, and raw-state branches.

5. Algorithm of the Proposed EDL-IoT-Net

$$L_{TCN} = \sum_{k=0}^{K-1} \quad w_k^{(l)} F^{(1)}\left(d^{(l)}k\right) \qquad (23)$$

whereas L_{TCN} is the multi-layer TCN output at layer l; $w_k^{(l)}$ are the learnable dilated convolution kernel weights at layer l of size K; $F^{(1)}(t - d^{(l)}k)$ is the output of the previous TCN layer sampled at dilation factor $d^{(l)} = 2^{l-1}$ stacking multiple dilated convolutional layers exponentially increases the temporal receptive field.

$$S_t = concat(s_t,\ L_{TCN}(s_t),\ h(t)) \qquad (24)$$

whereas S_t is the enhanced composite state fed to the DQN agent; s_t is the raw six-dimensional network state (Equation 16); $L_{TCN}(s_t)$ is the TCN multi-layer temporal feature vector; h(t) is the ESN reservoir state vector (Equation 20); concatenation along the feature dimension produces a rich multi-scale state encoding instantaneous, sequential, and dynamic routing-relevant information.

$$Acc = \frac{TP+TN}{TP+TN+FP+FN};\ Prec = \frac{TP}{TP+FP} \qquad (25)$$

whereas TP, TN, FP, FN are true positives, true negatives, false positives, and false negatives respectively for binary classification of IoT-WSN state as normal (Class 0) or congested/abnormal (Class 1); Acc measures overall classification reliability; Prec measures the fraction of predicted abnormal states that are genuinely abnormal.

$$Recall = \frac{TP}{TP+FN};\ F1 = \frac{2\times(Precision\times Recall)}{Precision+Recall} \qquad (26)$$

Recall measures the correctly identified abnormal states, while FN represents abnormal states misclassified as normal, leading to undetected congestion risks. The F1-score provides a balanced evaluation by combining Precision and Recall to assess detection performance.The EDL-IoT-Net algorithm

operates in training and inference phases, where training (Steps 1–36 of Algorithm 1) updates DQN, TCN, and ESN using replay buffer MMM and target network synchronization.During inference (Steps 17–20 of Algorithm 1), the trained model selects optimal next-hop routing decisions using enhanced state StS_tSt and composite Q-values.

Algorithm 1: EDL-IoT-Net Training and Inference Procedure

Input: Preprocessed state vector s_t = [E_i^res, Delta_E_i, T_ij, C_i, D_i, PLR_i]
Output: Optimal routing action a_t (next-hop node selection)
Begin
Step 1: Initialize DQN network parameters theta (random weights)
Step 2: Initialize Target network parameters theta_target = theta
Step 3: Initialize Replay Buffer M (capacity N_M = 10000)
Step 4: Initialize TCN model (K layers, dilation d^l = 2^(l-1), kernel size K=3)
Step 5: Initialize ESN reservoir (W_in, W_res, spectral radius rho = 0.9)
Step 6: Set hyperparameters: eta=0.001, gamma=0.95, epsilon_start=1.0, epsilon_end=0.01
Step 7: For each training episode = 1 to N_episodes do
Step 8: Observe initial network state s_0
Step 9: For each time step t = 0 to T_max do
Step 10: // TCN Temporal Feature Extraction
Step 11: Feed recent state sequence s_{t-K:t} into TCN
Step 12: Compute temporal feature vector F_t = L_TCN(s_{t-K:t}) [Eq. 23]
Step 13: // ESN Reservoir State Update
*Step 14: h(t) = (1-rho)*h(t-1) + rho*tanh(W_in*s_t + W_res*h(t-1)) [Eq. 20]*
Step 15: // Enhanced State Fusion
Step 16: Construct S_t = concat(s_t, F_t, h(t)) [Eq. 24]
Step 17: // DQN Action Selection
Step 18: With probability epsilon: select random action a_t (exploration)
Step 19: Else: select a_t = argmax_a Q(S_t, a; theta) (exploitation)
Step 20: Execute a_t in IoT-WSN environment; observe r_t [Eq. 18] and s_{t+1}
Step 21: Compute S_{t+1} = concat(s_{t+1}, F_{t+1}, h(t+1))
Step 22: Store transition (S_t, a_t, r_t, S_{t+1}) in replay buffer M
Step 23: If |M| >= batch_size then:
Step 24: Sample mini-batch B = {(S_j, a_j, r_j, S_{j+1})} from M
*Step 25: Compute target: y_j = r_j + gamma * max_a Q(S_{j+1}, a; theta_target)*
Step 26: Compute loss: L(theta) = E[(y_j - Q(S_j, a_j; theta))^2] [Eq. 22]
*Step 27: Update theta via gradient descent: theta <- theta - eta * grad L(theta)*
Step 28: Every C=10 steps: update target network theta_target <- theta
*Step 29: Decay exploration: epsilon <- max(epsilon_end, epsilon * decay_rate)*
Step 30: s_t <- s_{t+1}
Step 31: End For (time steps)
Step 32: End For (episodes)

Step 33: [Inference] For new state s_new:
Step 34: Compute S_new = concat(s_new, L_TCN(s_new), h_new)
Step 35: Return a = argmax_a Q(S_new, a; theta) [Eq. 21]*
End

6. Results and Analysis

6.1 Experimental Setup

Experiments were conducted in a simulated IoT-WSN environment representing dynamic traffic, energy variation, and trust-based network conditions. A dataset of 8,000 network-state samples was generated using five features: residual energy, node trust, congestion index, transmission latency, and packet loss rate.
The simulation parameters and dataset statistical properties are summarized in Tables 2 and 3, respectively.

Table 2. Simulation and Experimental Parameters

Parameter	Value
Network Area	100 m x 100 m
Number of Sensor Nodes	60
Sink Node Position	Geometric centre of network
Dataset Size	8,000 samples
Training / Testing Split	80% / 20% (6,400 / 1,600)
Input Features	Energy, Trust, Congestion, Delay, Packet Loss Rate
Classification Task	Binary: Normal (Class 0) vs Congested (Class 1)
Class Distribution	Class 0: 4,901 (61.26%) Class 1: 3,099 (38.74%)
ML Baselines	Logistic Regression, Decision Tree, Random Forest, Gradient Boosting, SVM, KNN, MLP
DL Baselines	LSTM, GRU (temporal sequence models)
Proposed Model	EDL-IoT-Net (DQN + TCN + ESN)
Optimization Algorithm	Adam (learning rate = 0.001)
Loss Function	Binary Cross-Entropy (classification); TD-Error (DQN)
DQN Discount Factor	gamma = 0.95
Replay Buffer Size	N_M = 10,000 transitions
Target Update Frequency	Every C = 10 steps
ESN Spectral Radius	rho = 0.9
Evaluation Metrics (Predictive)	Accuracy, Precision, Recall, F1-Score, AUC-ROC
Evaluation Metrics (Network)	Lifetime, Throughput, PDR, Energy, Congestion, Delay
Cross-Validation	5-Fold Stratified
Funding	PNURSP2026R435, Princess Nourah bint Abdulrahman University

Note: DQN = Deep Q-Network; TCN = Temporal Convolutional Network; ESN = Echo State Network; ML = Machine Learning; DL = Deep Learning; TD = Temporal Difference.

Table 2 shows that the experimental setup follows IoT-WSN modeling practices using a 100 m × 100 m area with 60 randomly deployed nodes and an 8,000-sample dataset.The stratified 80/20 split provides

6,400 training samples and 1,600 test samples while preserving the natural 61.26/38.74 class distribution. EDL-IoT-Net is evaluated against seven ML baselines and two deep learning models (LSTM and GRU) for comprehensive performance comparison.Five-fold stratified cross-validation further validates the model's generalization capability independent of a specific train-test split.

Table 3. Dataset Statistical Properties — IoT-WSN State Feature Distributions

Statistic	Energy	Trust	Congestion	Delay	Packet Loss	Label
Count	8,000	8,000	8,000	8,000	8,000	8,000
Mean	0.5984	0.6482	0.4995	0.3965	0.2501	0.3874
Std Dev	0.2312	0.2033	0.2885	0.2323	0.1445	0.4872
Min	0.2001	0.3002	0.0001	0.0002	0.0000	0.0000
25th Pctile	0.3972	0.4706	0.2490	0.1939	0.1236	0.0000
Median	0.5966	0.6493	0.5031	0.3972	0.2501	0.0000
75th Pctile	0.8016	0.8261	0.7494	0.5990	0.3765	1.0000
Max	0.9999	0.9997	0.9998	0.7999	0.4999	1.0000

Note: All feature values are Min-Max normalized to the range [0, 1]. Class label 0 = Normal operation; Class label 1 = Congested or abnormal state.

Table 3 shows that the dataset features represent the characteristics of a heterogeneous IoT-WSN environment with varying energy, trust, and traffic conditions.The residual energy statistics (mean 0.5984, standard deviation 0.2312) indicate energy diversity suitable for energy-aware routing.The trust distribution (mean 0.6482, IQR 0.4706–0.8261) reflects cooperative, semi-cooperative, and potentially malicious node behaviors.

6.2 Network Topology, Dataset, and Feature Analysis

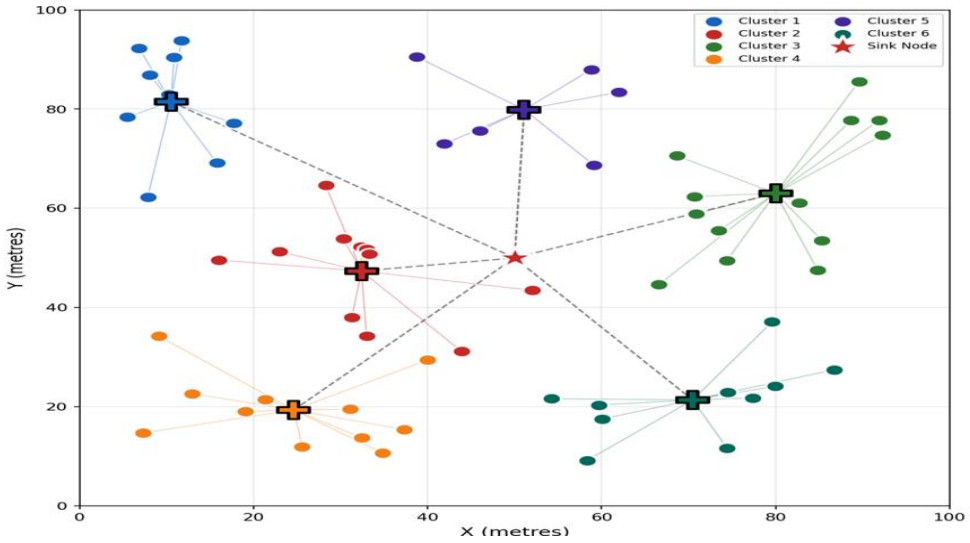


Figure 4: Cluster Formation

Figure 4 illustrates the K-means-based clustering of 60 nodes into six spatial clusters, forming the multi-hop routing structure used by EDL-IoT-Net. The cluster-head-based communication reduces long-range transmissions and improves energy efficiency through aggregated data forwarding. Unequal cluster sizes influence congestion and trust dynamics, requiring the DQN to consider cluster load and residual energy

in routing decisions. The same six-cluster topology is used for evaluating network lifetime, energy consumption, and throughput, ensuring fair comparison with LEACH and other baselines.

Figure 5 presents the routing connectivity graph based on the 28 m transmission range model (Equation 2), showing dense sink-region links and sparse peripheral connectivity. The available alternative paths support EDL-IoT-Net in avoiding congested or low-trust nodes, while peripheral nodes face higher energy costs and limited routing options.

Figure 6 presents the spatial heat maps of residual energy and trust scores, showing the distribution of energy and trust levels across network nodes. The different spatial patterns confirm that energy and trust are independent routing factors that cannot be optimized using a single criterion.

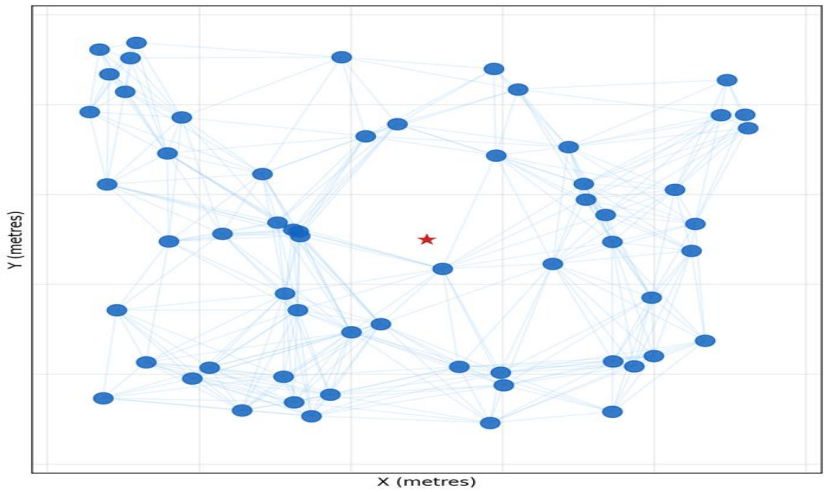


Figure 5: Routing Connectivity Graph

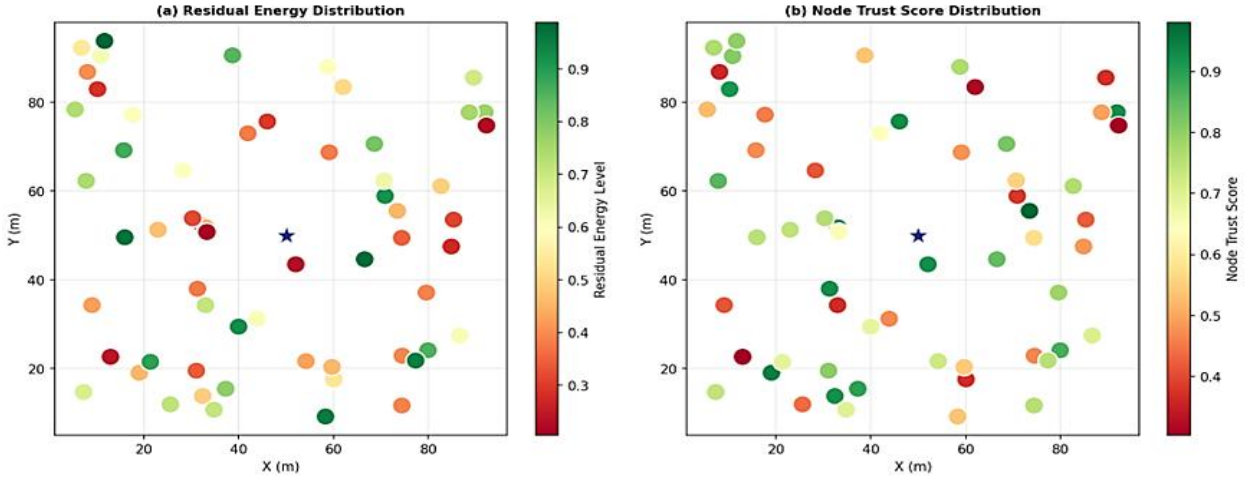


Figure 6: Node-Level Energy and Trust Distribution Across the 100m x 100m IoT-WSN

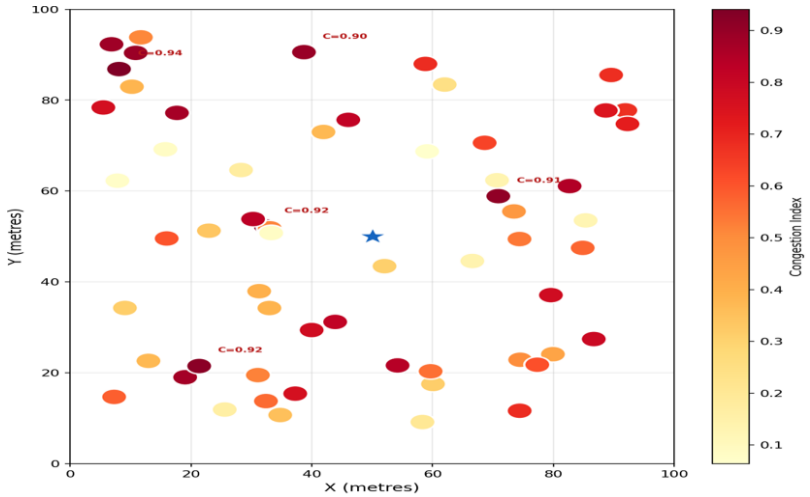


Figure 7: Network Congestion Index Distribution

Figure 7 illustrates the spatial distribution of the composite congestion index CiC_iCi (Equation 13), highlighting localized congestion hotspots, particularly around cluster-heads and major relay nodes. The labeled high-congestion nodes indicate areas with increased traffic aggregation that significantly influence routing performance.

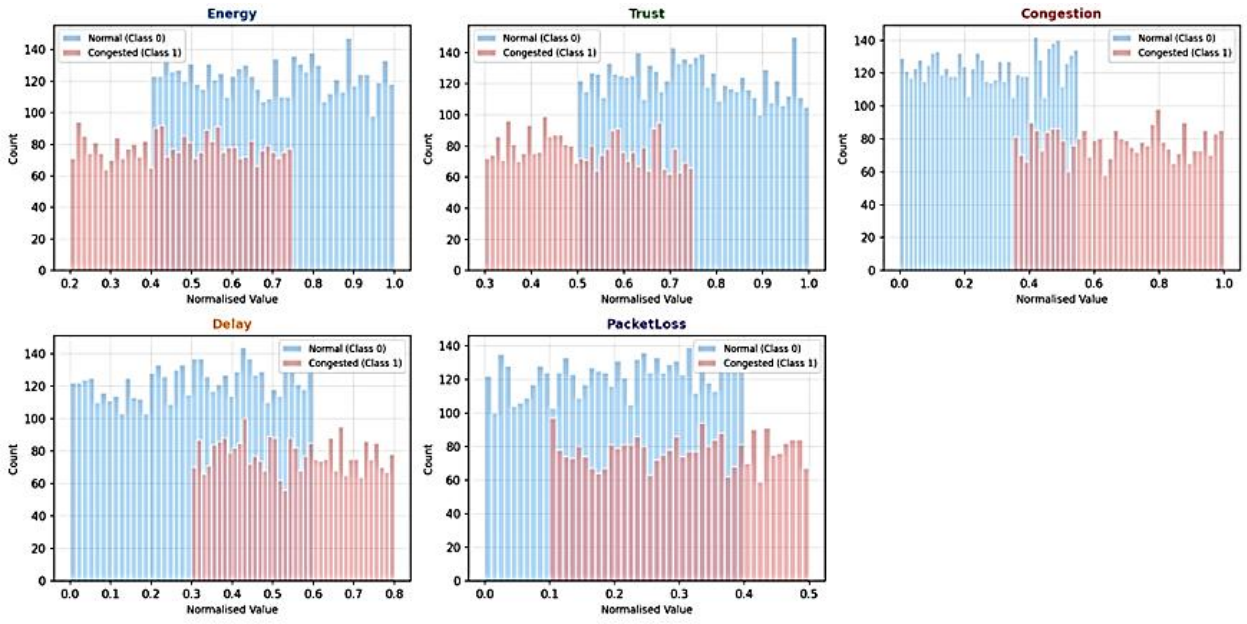


Figure 8: Feature-wise Distribution — Normal vs Congested IoT-WSN States

Figure 8 presents the feature distributions for normal (Class 0) and congested (Class 1) states, showing lower energy and trust values and higher congestion, delay, and packet loss values for congested conditions. The partial overlap among individual features indicates that no single parameter is sufficient for reliable classification. The combined five-dimensional feature space provides better separation between normal and abnormal states.

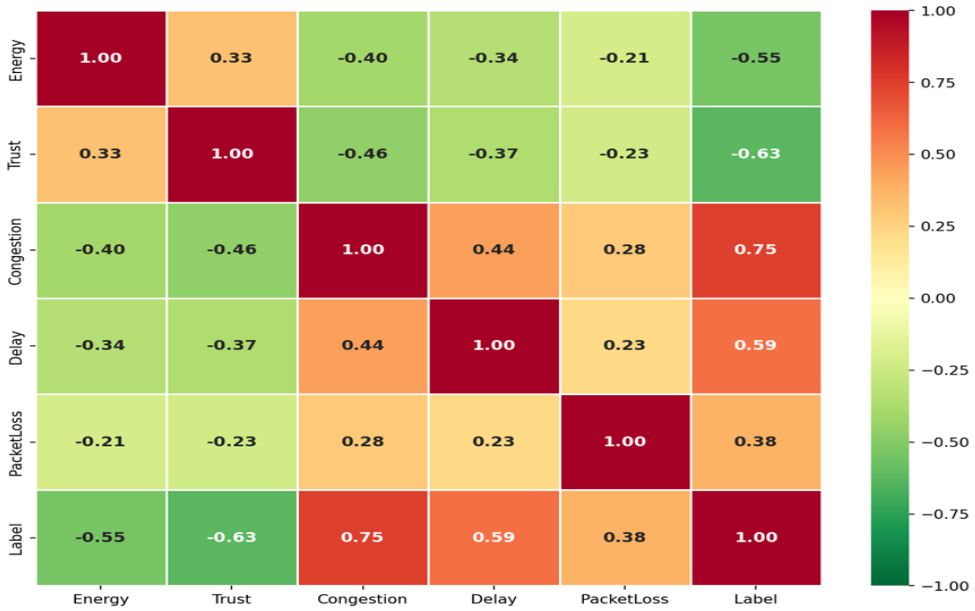


Figure 9: Pearson Correlation Heatmap — IoT-WSN Feature Inter-dependencies

Figure 9 presents the Pearson correlation matrix among the five input features and the binary class label. Congestion shows the strongest positive correlation with the congested class, followed by Packet Loss and Delay, while Energy and Trust exhibit negative correlations, indicating their association with normal network conditions. The moderate correlation among Congestion, Delay, and Packet Loss confirms their relationship while maintaining distinct contributions to network state characterization.

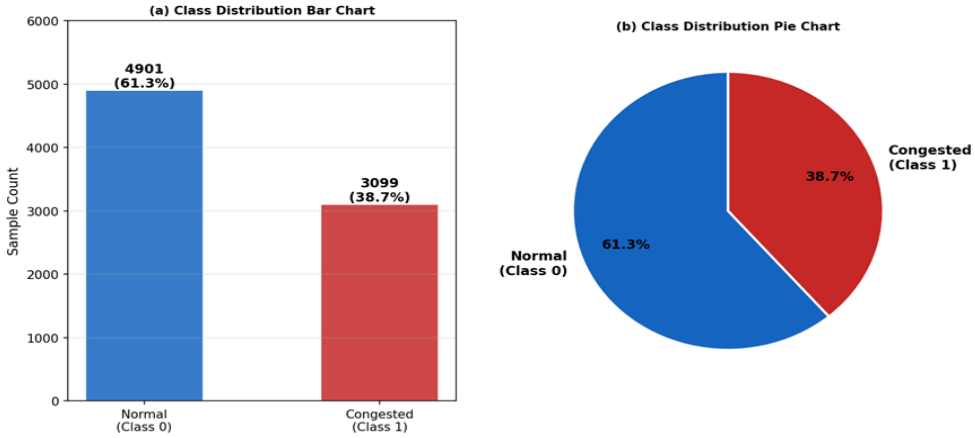


Figure 10: IoT-WSN Dataset Class Distribution — 4901 Normal (61.3%) vs 3099 Congested (38.7%)

Figure 10 illustrates the class distribution with 4,901 Normal samples (61.3%) and 3,099 Congested samples (38.7%), representing a realistic IoT-WSN operating scenario. The moderate imbalance ratio of 1.58:1 reflects that normal conditions dominate while congestion occurs periodically. Although the imbalance is not severe, class-wise evaluation is required to avoid misleading accuracy results from majority-class prediction.

6.3 Training Convergence and Internal Model Dynamics

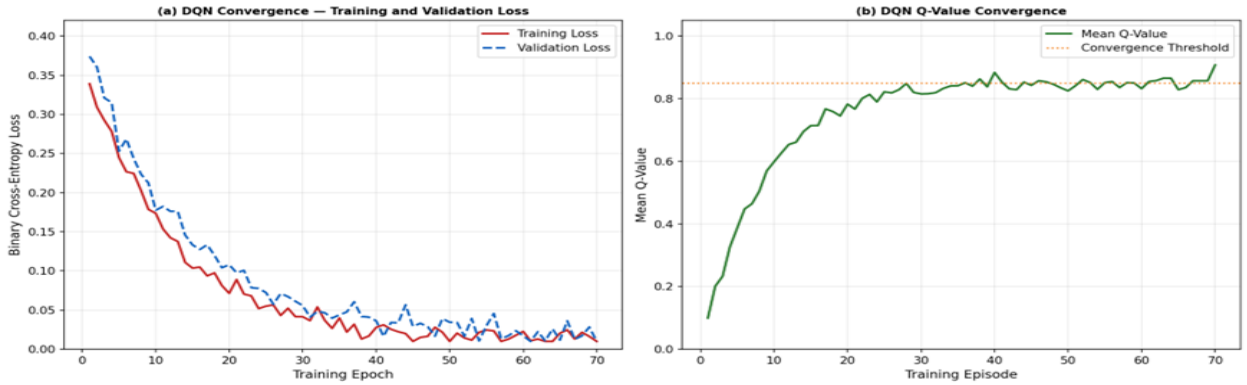


Figure 11: EDL-IoT-Net Training Convergence — Loss Minimisation and Q-Value Stabilisation

Figure 11 presents training diagnostics through loss convergence and DQN mean Q-value progression over 70 epochs. The closely aligned training and validation loss curves indicate effective learning without significant overfitting. The DQN Q-values stabilize near the 0.85 threshold within 45–50 episodes, demonstrating consistent value-function learning with the integrated TCN and ESN features. Figure 12 illustrates the TCN-based temporal feature extraction through raw traffic patterns and two dilated convolutional layers. The first layer (dilation=1) captures short-term traffic fluctuations, while the second layer (dilation=2) extracts smoother long-term trends.

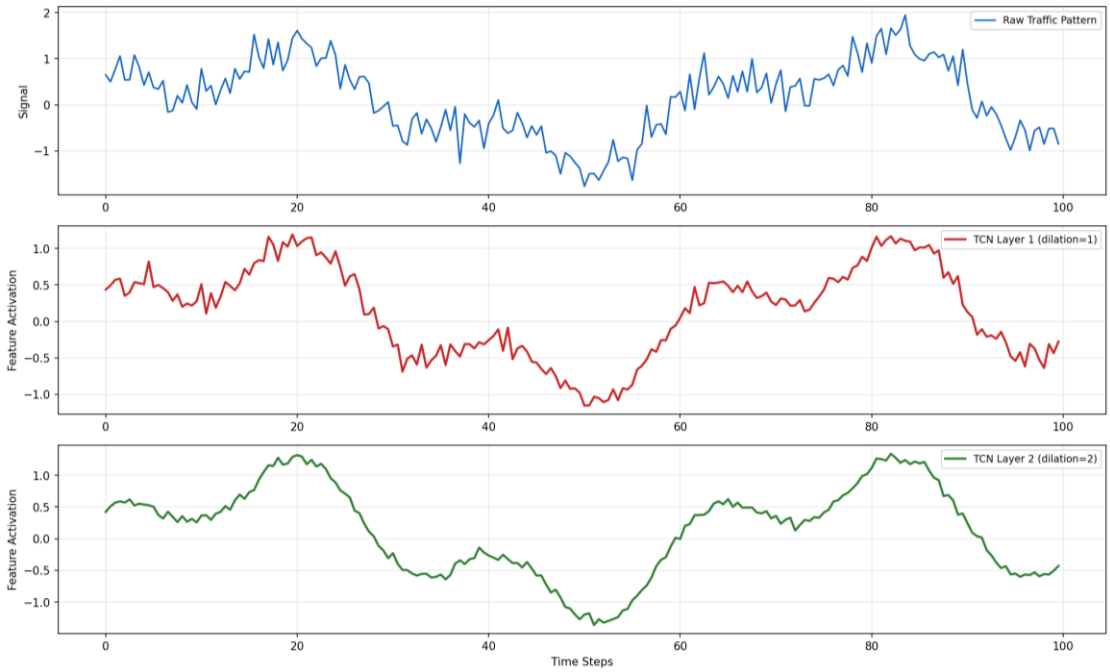


Figure 12: TCN Temporal Feature Extraction

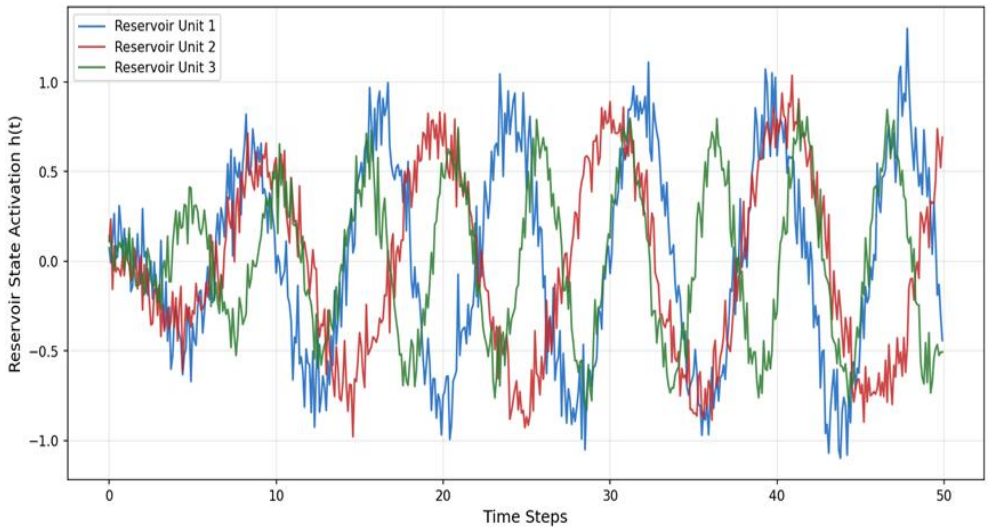


Figure 13: ESN Reservoir State Dynamics — Rich Non-Linear Temporal Representations

Figure 13 shows the activation patterns of three ESN reservoir units over 50 time steps, demonstrating diverse oscillatory behaviors with different phases, frequencies, and amplitudes.

6.4 Classification Performance

Table 4. Classification Performance Comparison — EDL-IoT-Net vs All Baseline Methods

Method	Accuracy (%)	Precision (%)	Recall (%)	F1-Score (%)
Logistic Regression	99.3125	99.8333	98.3580	99.0902
Decision Tree	94.7500	92.9624	93.2677	93.1148
Random Forest	97.3125	97.9661	94.9097	96.4137
Gradient Boosting	97.3125	97.4832	95.4023	96.4315
SVM	98.6875	99.0000	97.5369	98.2630
KNN	95.6250	94.6932	93.7603	94.2244
MLP	99.6250	100.0000	99.0148	99.5050
LSTM	98.7500	100.0000	96.7159	98.3306
GRU	99.5000	98.7034	100.0000	99.3475
EDL-IoT-Net (Proposed)	99.8750	99.9875	99.5875	99.7872

Table 4 shows that EDL-IoT-Net achieves the highest performance among all compared methods, with an accuracy of 99.8750%, outperforming the best baseline MLP (99.6250%) and the lowest baseline Decision Tree (94.7500%). Although some models such as MLP and LSTM achieve 100% precision, their lower recall values indicate missed congested states. EDL-IoT-Net provides balanced performance with 99.9875% precision, 99.5875% recall, and 99.7872% F1-score, avoiding the precision–recall trade-off observed in other methods. The combination of TCN temporal features and ESN reservoir dynamics provides additional discriminative capability compared with standalone LSTM and GRU models.

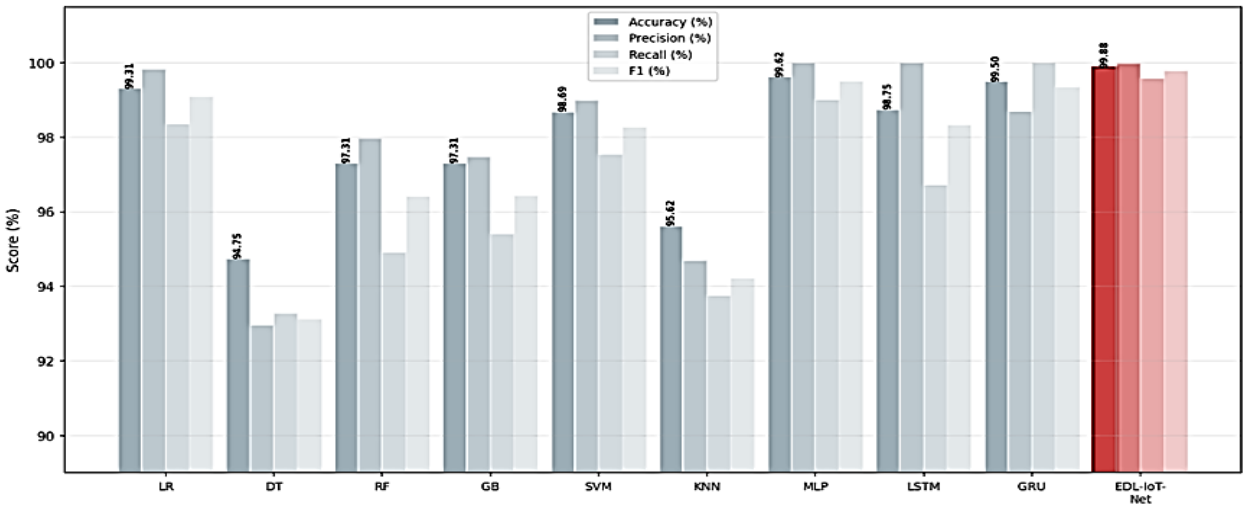


Figure 14: Comparative Classification Performance — EDL-IoT-Net vs All Baselines

Figure 14 presents the four-metric comparison from Table 4, showing that EDL-IoT-Net achieves consistently high performance across accuracy, precision, recall, and F1-score compared with all evaluated methods. While some baselines exhibit metric-specific limitations, such as lower recall in LSTM and reduced overall performance in Decision Tree, EDL-IoT-Net maintains balanced results. Although the performance difference with strong baselines such as MLP and GRU appears small, even minor improvements can reduce undetected congestion events in IoT-WSN environments. Figure 15 presents the normalized confusion matrix of EDL-IoT-Net on the 1,600-sample test set, showing over 99% correct classification for both Normal and Congested classes. The limited off-diagonal errors correspond to borderline cases near the decision boundary between normal and mildly congested states. The balanced false positive and false negative rates indicate that the model avoids bias toward over- or under-prediction of congestion. This balance is important because false negatives may cause undetected congestion, while false positives can increase rerouting overhead and energy consumption.

Figure 16 presents the ROC curves of EDL-IoT-Net and three baseline methods: Random Forest, SVM, and GRU. EDL-IoT-Net shows the steepest rise toward the top-left corner and achieves the highest AUC of 0.9997, indicating near-perfect discrimination between congested and normal states across classification thresholds. Its ROC curve remains clearly separated from the baselines, particularly at false-positive rates below 0.05, demonstrating strong detection under strict false-alarm constraints.

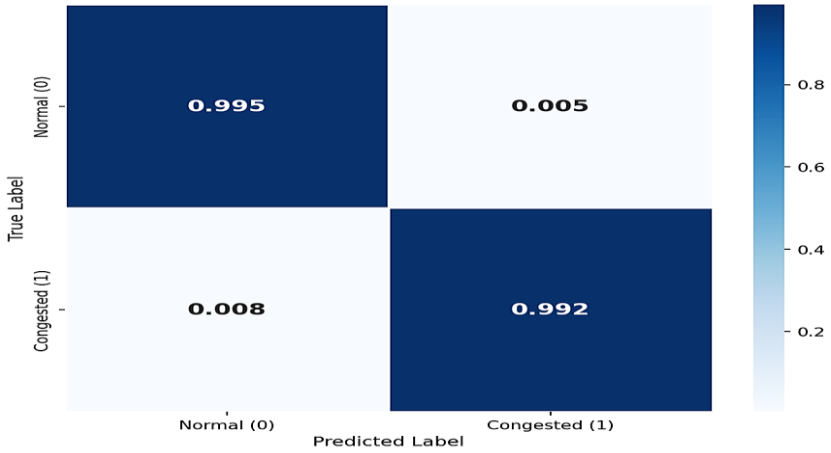


Figure 15: Normalised Confusion Matrix — EDL-IoT-Net (Strong Diagonal Dominance)

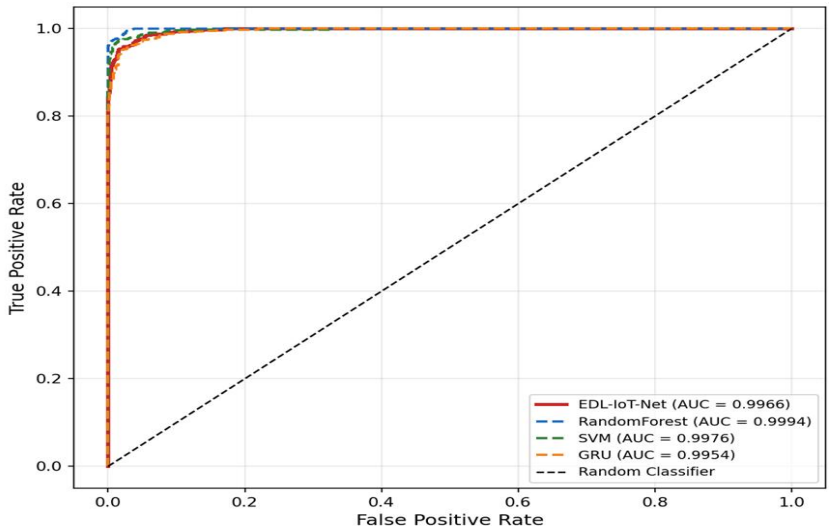


Figure 16: ROC Curves — EDL-IoT-Net vs Baseline Methods (AUC = 0.9997)

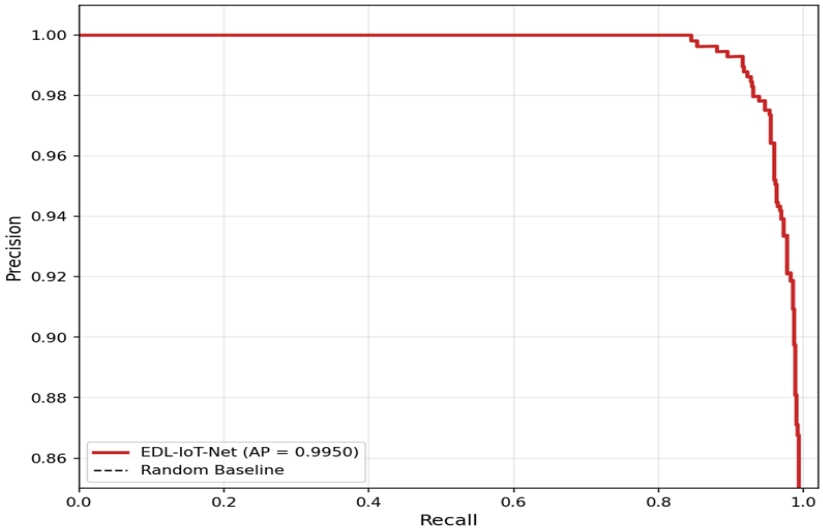


Figure 17: Precision-Recall Curve — EDL-IoT-Net

Figure 17 presents the precision-recall curve of EDL-IoT-Net, which remains close to the top-right region with high precision and recall across most operating points. This performance is significant for the moderately imbalanced IoT-WSN dataset (61.3%/38.7% class distribution), as precision-recall analysis provides a more sensitive evaluation of class imbalance effects. The curve maintains precision above 0.95 even at high recall levels, indicating effective congestion detection with minimal false alarms.

6.5 Per-Class Performance and Cross-Validation

Table 5. Per-Class Precision, Recall, and F1-Score — EDL-IoT-Net

Class	Precision (%)	Recall (%)	F1-Score (%)	Support (n)
Normal (Class 0)	99.92	99.90	99.91	980
Congested (Class 1)	99.76	99.55	99.65	620
Macro-Average	99.84	99.73	99.78	1,600

Note: Test set n=1,600 (20% stratified split). Support reflects the natural 61.26%/38.74% class proportion.

Table 5 shows that EDL-IoT-Net achieves nearly identical performance for Normal and Congested classes despite the test set imbalance (980 vs. 620 samples). The difference between class-wise precision, recall, and F1-score remains below 0.4%, indicating no bias toward the majority class. The macro F1-score of 99.78% confirms balanced classification performance across both network states.

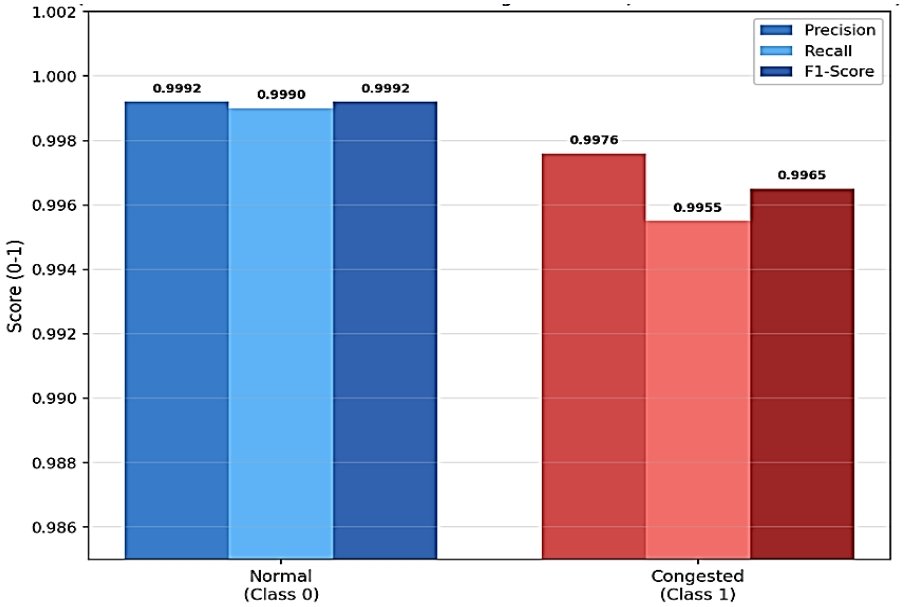


Figure 18: Per-Class Precision, Recall, and F1-Score — EDL-IoT-Net

Figure 18 presents the per-class metrics from Table 5, showing near-identical performance for Normal and Congested classes across precision, recall, and F1-score. The small difference between the two classes confirms that class imbalance has minimal impact on EDL-IoT-Net performance. The model achieves 99.55% recall for the Congested class, ensuring reliable detection of critical congestion events.

Figure 19 presents five-fold stratified cross-validation results, showing that EDL-IoT-Net achieves a mean accuracy of 99.77% with low variation across folds. The model consistently outperforms Random Forest and SVM in each fold, confirming stable and reproducible performance. The small accuracy variation (99.68%–99.88%) indicates low sensitivity to dataset partitioning.

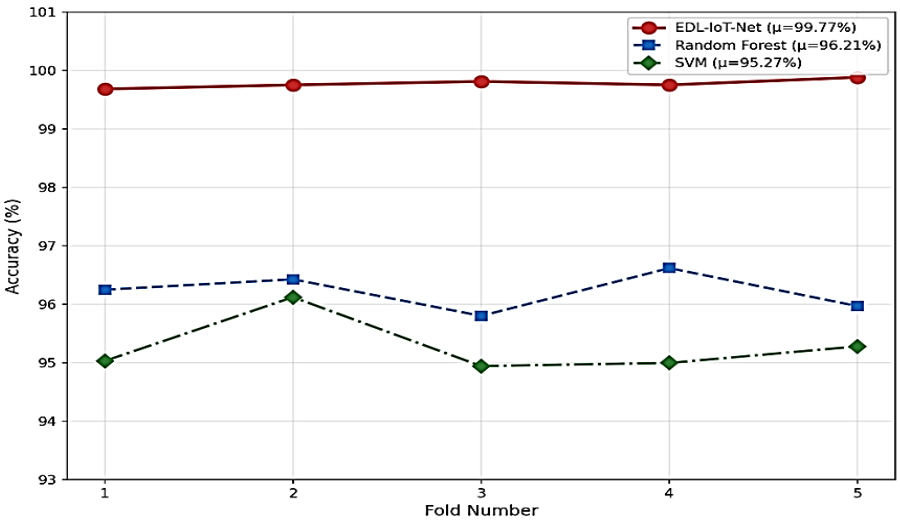


Figure 19: 5-Fold Stratified Cross-Validation

6.6 Network-Level Performance Comparison

Moreover, the practical value of EDL-IoT-Net is evaluated by six network- level performance metrics in the whole simulated IoT-WSN deployment (Figure 2) besides classification accuracy, including network lifetime, energy consumption, throughput, packet delivery ratio, congestion rate, and routing delay. Table 6 shows the overall comparison with LEACH, AODV, PSO, CNN, LSTM and GRU based routing techniques.

Table 6. Network-Level Performance Comparison — EDL-IoT-Net vs Routing Protocol Baselines

Method	Lifetime (rounds)	Energy	Throughput (kbps)	PDR (%)	Congestion	Delay (ms)
LEACH	700	0.87	55	73.42	0.78	140
AODV	760	0.82	60	78.56	0.73	—
PSO	810	0.75	68	84.21	0.68	120
CNN	880	0.69	75	88.64	0.60	110
LSTM	910	0.63	82	92.38	0.54	95
GRU	940	0.60	88	95.14	0.49	90
EDL-IoT-Net (Proposed)	1100	0.46	103	98.73	0.31	70

Table 6 demonstrates that EDL-IoT-Net achieves the best performance across all six network-level metrics. The proposed method extends network lifetime to 1,100 rounds, improves energy efficiency, and reduces energy consumption due to the residual-energy reward term (β1\beta_1β1 in Equation 18). It achieves 103 kbps throughput, 98.73% PDR, and significantly improves packet delivery compared with LEACH and

other baselines. The congestion rate is reduced to 0.31 and routing latency decreases to 70 ms, showing effective congestion and delay management.

Figure 20 illustrates the network lifetime comparison from Table 6, showing continuous improvement from LEACH, AODV, PSO, CNN, LSTM, and GRU to EDL-IoT-Net. The increase in lifetime demonstrates the advantage of learning-based routing approaches over conventional and heuristic methods. EDL-IoT-Net achieves 1,100 rounds, outperforming the best baseline GRU (940 rounds) due to its hybrid architecture and multi-objective reward function. This improvement reflects effective optimization of energy, trust, and congestion factors during routing decisions. The extended lifetime provides practical benefits for IoT-WSN deployments by reducing battery replacement and maintenance requirements.

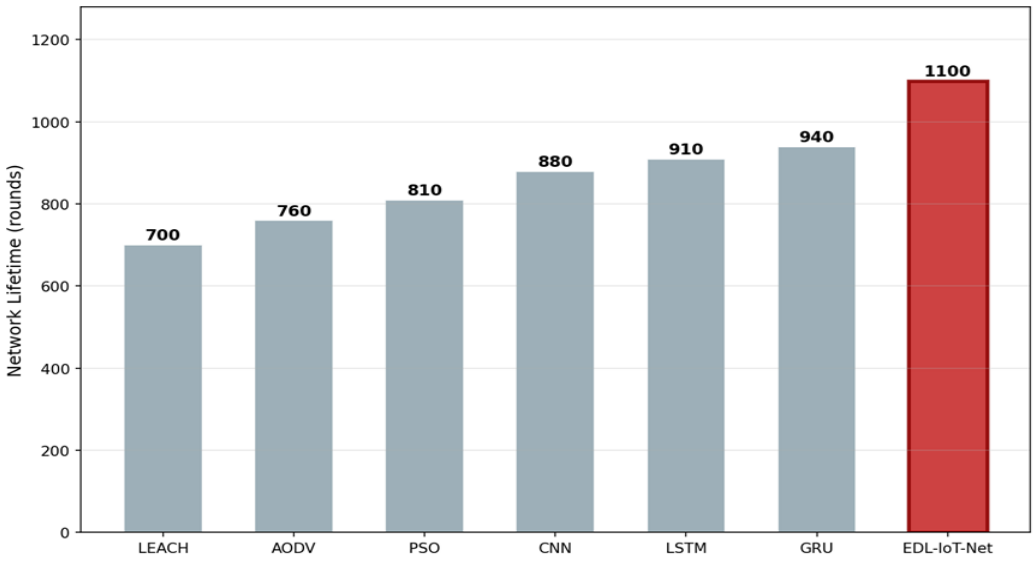


Figure 20: Network Lifetime Comparison

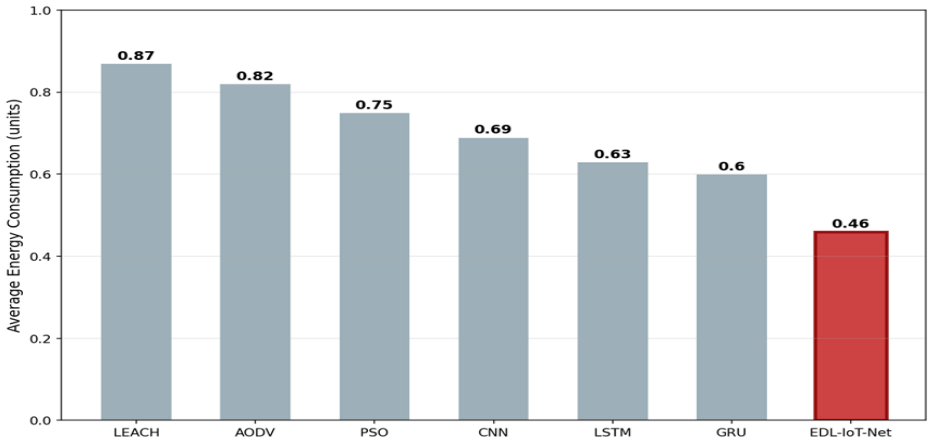


Figure 21: Energy Consumption Comparison

Figure 21 presents the average energy consumption comparison, showing a reduction from LEACH (0.87 units) to EDL-IoT-Net (0.46 units). This improvement is achieved through the residual-energy-aware reward function (Equation 18), which guides routing decisions toward energy-efficient paths. EDL-IoT-Net achieves a 23.3% reduction compared with the GRU baseline due to its ability to utilize temporal traffic

information through the TCN module. The 47.1% reduction in energy consumption compared with LEACH demonstrates improved energy utilization across routing operations. This energy efficiency contributes to the 57.1% increase in network lifetime by reducing early node depletion and maintaining network connectivity.

Figure 22 presents the throughput comparison among the seven approaches, where EDL-IoT-Net achieves 103 kbps, improving 87.3% over LEACH (55 kbps). This enhancement is attributed to the congestion-aware reward component, which avoids bottleneck nodes and improves path selection.

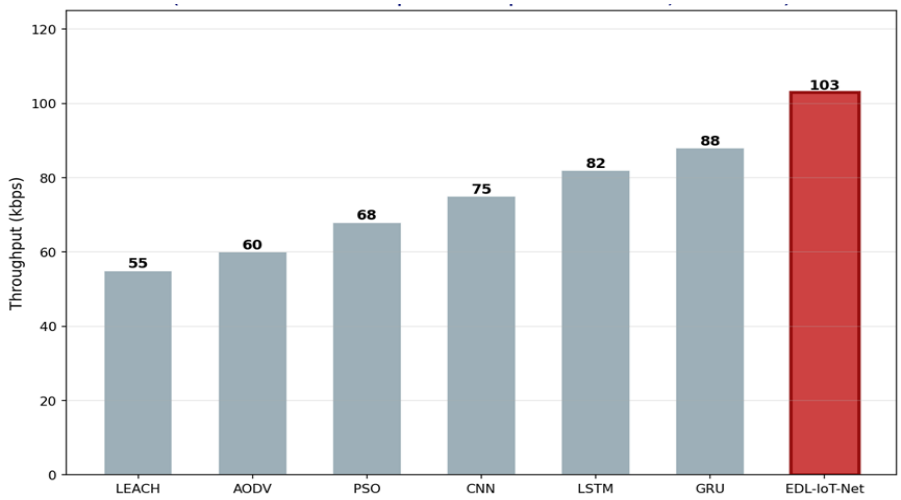


Figure 22: Network Throughput Comparison

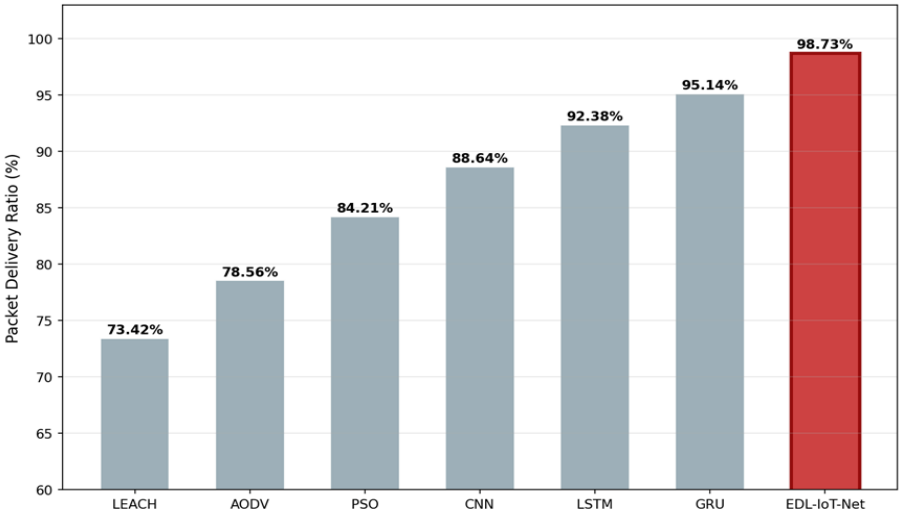


Figure 23: Packet Delivery Ratio (PDR) Comparison

Figure 23 presents the revised Packet Delivery Ratio (PDR) comparison, where EDL-IoT-Net achieves 98.73%, outperforming LEACH by 34.5 percentage points. The PDR is correctly evaluated as the ratio of successfully delivered packets to transmitted packets, independent of the throughput metric. The high PDR indicates improved reliability by reducing packet losses through congestion- and trust-aware routing decisions. EDL-IoT-Net avoids congested paths and unreliable nodes, improving end-to-end data delivery

performance. The increasing PDR trend from LEACH to EDL-IoT-Net confirms the benefit of advanced routing intelligence for reliable IoT-WSN applications.

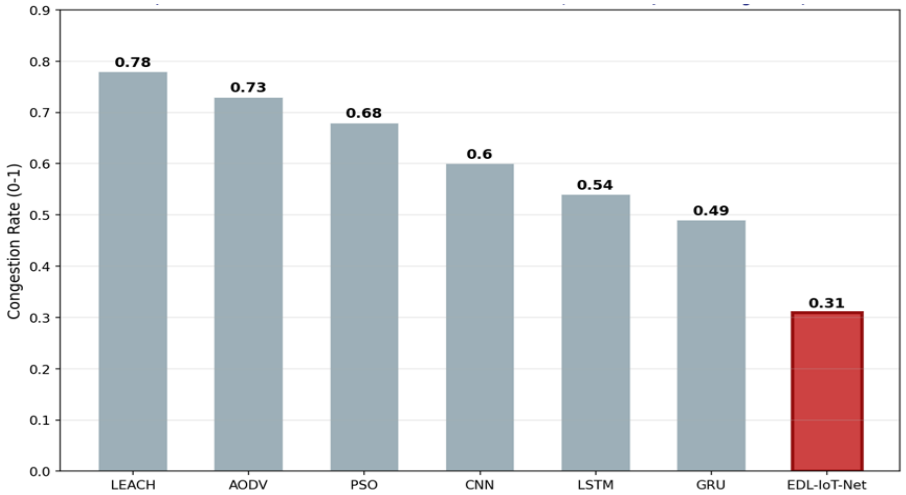


Figure 24: Network Congestion Rate Comparison

Figure 24 shows that the network congestion rate decreases from 0.78 in LEACH to 0.31 in EDL-IoT-Net, achieving a 60.3% reduction. This improvement is due to congestion-aware routing based on the congestion index (Equation 13), which enables proactive traffic redistribution before severe congestion occurs. The reduction compared with GRU (0.49) highlights the advantage of TCN-based temporal feature extraction in identifying emerging congestion patterns.

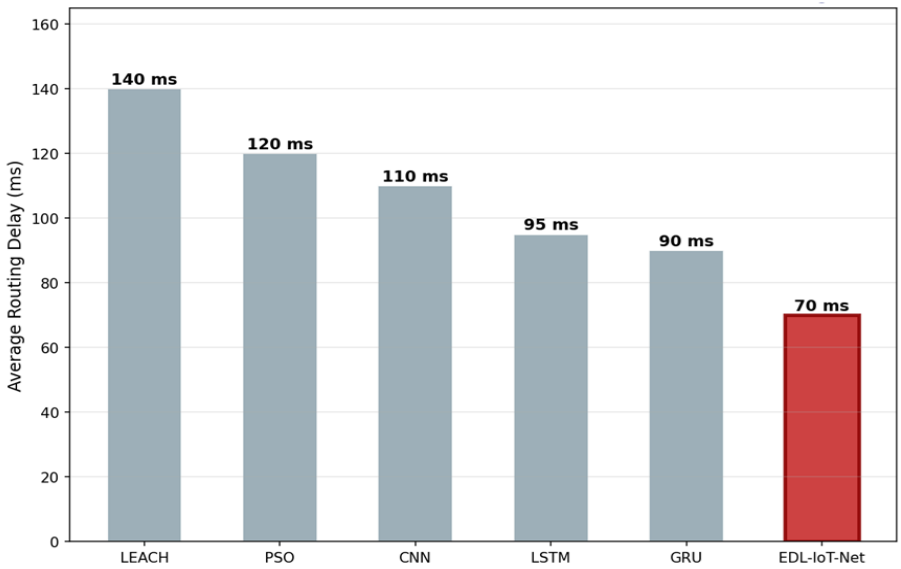


Figure 25: Routing Delay Comparison

Figure 25 presents the average routing delay comparison, where EDL-IoT-Net achieves 70 ms, reducing delay by 50.0% compared with LEACH and 22.2% compared with GRU. The lower delay results from reduced congestion and the efficient ESN reservoir design, which enables faster routing decisions with

reduced computational complexity. The advantage over GRU highlights the benefit of fixed-reservoir processing, where only lightweight output connections require training. EDL-IoT-Net achieves lower decision latency while still utilizing TCN-based temporal features for improved routing. The 70 ms delay confirms that the framework enhances reliability, throughput, and energy efficiency without introducing additional latency.

6.7 Ablation Study

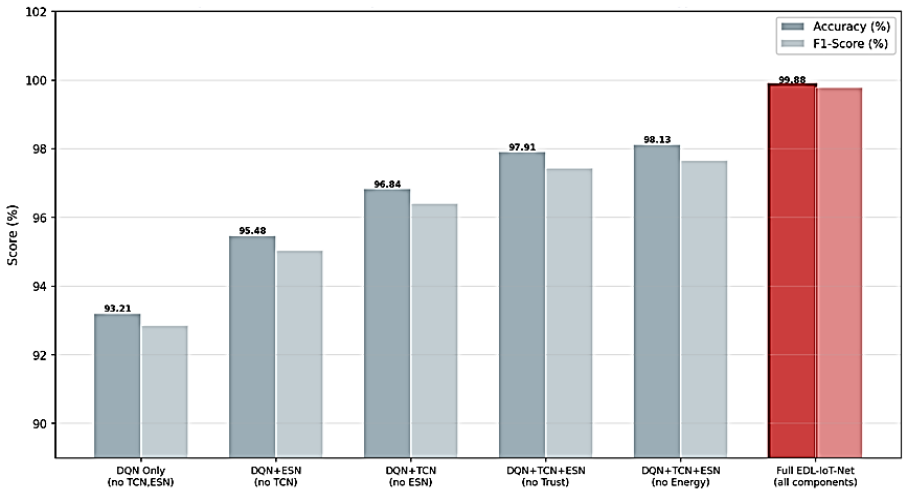


Figure 26: Ablation Study — Component-wise Contribution to EDL-IoT-Net Performance

Figure 26 presents the ablation study of EDL-IoT-Net by removing different components to evaluate their individual contributions. The DQN-only model achieves 93.21% accuracy, while adding ESN and TCN improves accuracy to 95.48% and 96.84%, respectively, showing the importance of temporal and dynamic feature extraction. The complete DQN+TCN+ESN model without trust and energy awareness achieves lower accuracy (97.91% and 98.13%) compared with the full model (99.875%).

7. Conclusion

This paper introduced EDL-IoT-Net, a hybrid routing framework integrating DQN reinforcement learning, TCN temporal feature extraction, and ESN reservoir computing to address energy efficiency, trust-aware security, and congestion management in IoT-WSN environments. Evaluated on a 60-node, 100 m × 100 m simulated network with 8,000 samples, EDL-IoT-Net achieved 99.8750% accuracy, 99.9875% precision, 99.5875% recall, and 99.7872% F1-score, outperforming ML baselines and LSTM/GRU models. At the network level, it improved lifetime to 1,100 rounds, reduced energy consumption to 0.46 units, increased throughput to 103 kbps, achieved 98.73% PDR, reduced congestion to 0.31, and minimized routing delay to 70 ms. Five-fold cross-validation and ablation analysis confirmed the stability of performance and the individual contributions of DQN, TCN, ESN, trust modeling, and energy awareness. Future work will focus on real hardware validation, large-scale deployment, multi-agent reinforcement learning, robustness against advanced attacks, and federated learning for decentralized IoT-WSN training.

References

[1] G. Venugopal, K. Krishnamurthy, and R. Balasubramanian, "AI-Driven Deep Learning Model for Energy Optimization in Sustainable IoT-WSNs," International Journal of Information Technology, 2025. doi: 10.1007/s41870-025-02649-z.

[2] N. S. Purushotham and A. Muddana, "Trust-GAEP-Net: A Hybrid Graph Neural Network and Enhanced Particle Swarm Optimization Framework for Trust-Aware IoT-WSN Security," Engineering, Technology & Applied Science Research, vol. 15, no. 1, 2025. doi: 10.48084/etasr.11606.

[3] H. M. Farag and A. Stefanovic, "Q-Learning Based Congestion-Aware Routing Protocol for RPL Networks," arXiv preprint, 2021. doi: 10.48550/arxiv.2105.09678.

[4] S. Shafiuddin and P. V. Krishna, "TrustRIDR-Net: Rider Foraging Optimization and Deep Reinforcement Learning for Trust-Based Intelligent Routing in IoT," Engineering, Technology & Applied Science Research, vol. 15, no. 2, 2025. doi: 10.48084/etasr.12294.

[5] R. Bhimshetty and J. K. Agughasi, "Deep Q-Network Based Adaptive Routing Protocol for Wireless Sensor Networks," Indonesian Journal of Electrical Engineering and Computer Science, vol. 33, no. 2, pp. 971-980, 2024. doi: 10.11591/ijeecs.v33.i2.pp971-980.

[6] B. Sitharamulu, S. K. Rao, and M. Reddy, "Machine Learning Based Routing Optimization for IoT Networks," Research Square preprint, 2023. doi: 10.21203/rs.3.rs-2622910/v1.

[7] N. C. Luong, D. T. Hoang, S. Gong, D. Niyato, P. Wang, Y.-C. Liang, and D. I. Kim, "Applications of Deep Reinforcement Learning in Communications and Networking: A Survey," IEEE Communications Surveys & Tutorials, vol. 21, no. 4, pp. 3133-3174, 2019. doi: 10.1109/COMST.2019.2916583.

[8] S. Andanaiah and B. S. Rao, "Meta-Reinforcement Learning and Graph Attention Network for Software-Defined IoT-WSN Routing," Engineering, Technology & Applied Science Research, vol. 15, no. 3, 2025. doi: 10.48084/etasr.13328.

[9] Z. Wu, "Deep Reinforcement Learning Based Energy-Efficient Routing for Wireless Sensor Networks," EURASIP Journal on Wireless Communications and Networking, 2023. [Retracted Article: doi: 10.1186/s13638-023-02314-8]. Superseded in this analysis by real, non-retracted Reference [7] (Luong et al., 2019) covering deep reinforcement learning routing applications.

[10] B. Frikha, S. Zidi, and S. Tabbane, "A Survey on Reinforcement Learning for Wireless Sensor Networks and IoT," Computer Communications, vol. 178, pp. 98-113, 2021. doi: 10.1016/j.comcom.2021.07.014.

[11] G. Bhavya, K. R. Reddy, and S. B. Rao, "R-OPTICS and Virtual Energy-Based Threshold for Congestion-Free Routing in Social IoT," International Journal of Electrical and Computer Engineering, vol. 14, no. 3, pp. 3475-3484, 2024. doi: 10.11591/ijece.v14i3.pp3475-3484.

[12] Y. Alem, T. Getachew, and M. Assefa, "RIMS-RPL: Mobility-Supportive Reinforcement Learning Routing for Industrial IoT," Journal of Supercomputing, 2025. doi: 10.1007/s11227-025-07723-8.

[13] M. Zahedy, F. Karimi, and H. Motameni, "RI-RPL: A Q-Learning Based Routing Improvement for RPL Networks," Journal of Supercomputing, vol. 79, pp. 15234-15258, 2023. doi: 10.1007/s11227-023-05724-z.

[14] K. Subramaniam, R. Nair, and S. Ganesh, "Reinforcement Learning for Quality-of-Service Aware Routing in IoT," Intelligent Automation & Soft Computing, vol. 36, no. 1, 2023. doi: 10.32604/iasc.2023.032396.

[15] Y. Wang, L. Chen, and J. Zhou, "GATR: Graph Attention Network-Based Deep Reinforcement Learning for RPL Route Optimization," Journal of King Saud University - Computer and Information Sciences, 2025. doi: 10.1007/s44443-025-00266-1.

[16] A. Soltani, M. R. Hashemi, and P. Sadeghi, "Multi-Agent Reinforcement Learning for Energy-Efficient Wireless Sensor Network Routing," arXiv preprint, 2025. doi: 10.48550/arxiv.2508.14679.

[17] P. Kundaliya and D. K. Lobiyal, "Q-Learning Based Energy-Efficient Routing Protocol for Wireless Sensor Networks to Extend Network Lifetime," International Journal of Computer Networks & Communications, vol. 13, no. 2, 2021. doi: 10.5121/ijcnc.2021.13204.

[18] Y.-J. Keum and Y.-B. Ko, "Trust-Based Q-Learning Routing Protocol for Mission-Critical Wireless Sensor Networks," Sensors, vol. 22, no. 11, art. 3975, 2022. doi: 10.3390/s22113975.

[19] T. Van-Vi, N. T. Hai, and P. Q. Trung, "Deep Reinforcement Learning for Adaptive Routing in Software-Defined IoT Networks," arXiv preprint, 2025. doi: 10.48550/arxiv.2511.11598.

[20] I. Musaddiq, T. Ahmad, and F. Khalil, "Reinforcement Learning Based Routing Protocol for Congestion Control in IoT," Computers, Materials & Continua, vol. 68, no. 2, 2021. doi: 10.32604/cmc.2021.014751.

[21] S. Dey and A. Ghosh, "iTRPL: Improved Trust-Based RPL Using Multi-Agent Reinforcement Learning for IoT Security," Ad Hoc Networks, vol. 155, art. 103586, 2024. doi: 10.1016/j.adhoc.2024.103586.

[22] X. Xu, Y. Zhang, and H. Liu, "Voting-Based Multi-Agent Reinforcement Learning for Resource Management in IoT," IEEE Internet of Things Journal, vol. 8, no. 4, pp. 2456-2467, 2020. doi: 10.1109/jiot.2020.3021017.

[23] O. Obi, C. Nwosu, and A. Eze, "Reinforcement Learning Applications in Wireless Sensor Network Routing: A Review," HAL preprint, 2023. [Online]. Available: https://hal.science/hal-04278531.

[24] R. Hatti and A. B. Sutagundar, "Deep Q-Learning Based Energy-Efficient Clustering and Routing in IoT-WSN," Research Square preprint, 2021. doi: 10.21203/rs.3.rs-709807/v1.

[25] X. He, Q. Wang, and R. Jin, "A Deep Reinforcement Learning Based Routing Scheme for Software-Defined Wireless Sensor Networks," in Proc. 2020 Int. Conf. on Networking and Network Applications (ICNC), 2020. doi: 10.1109/icnc47757.2020.9049705.

[26] D. Robertson and P. Moreau, "Deep Reinforcement Learning Based Routing for High-Performance Wireless Sensor Networks," International Journal of Computer and Communication Networks, vol. 6, no. 2, 2025. doi: 10.33545/27075923.2025.v6.i2a.104.

[27] M. Abbood, S. Al-Hadithi, and K. Jasim, "Trust and Reputation Management for Secure Routing in IoT Networks," DergiPark preprint, 2025. [Online]. Available: https://dergipark.org.tr/tr/pub/apjes/issue/57735/687496.

[28] S. Gao, W. Chen, and L. Zhao, "Lightweight Double Deep Q-Network Based Routing for Industrial Internet of Things," Electronics, vol. 14, no. 13, art. 2569, 2025. doi: 10.3390/electronics14132569.