

Article

Energy-Efficient Dynamic Retransmission Timeouts with Enhanced Stability for Constrained Application Protocol-Based Internet of Things Networks via Edge Intelligence-Assisted Cross-Layer Congestion Control

Suyoung Choi 

Department of Software Engineering, Dong-A University, Busan 49315, Republic of Korea;
csy100479@hanmail.net

Abstract

The co-existence of event-driven critical traffic and time-driven periodic traffic inevitably exacerbates cross-layer network congestion in resource-constrained edge environments. Although hybrid protocol architectures integrating the Constrained Application Protocol (CoAP) at the edge and Quick UDP Internet Connections (QUIC) in the core network have emerged, existing gateways manage these protocols independently, failing to provide an organic congestion control mechanism. To overcome these limitations, this paper proposes an ultra-lightweight Edge Intelligence (EI)-assisted end-to-end (E2E) CoAP-QUIC cross-layer congestion control framework powered by Proximal Policy Optimization (PPO). The proposed scheme introduces an ultra-lightweight traffic classification mechanism that instantly distinguishes traffic classes by parsing the existing two-bit type field in the CoAP header, effectively bypassing the payload inspection overhead. On the basis of this, the PPO agent shapes its reward function in real time, actively shifting optimization weights between delay reduction and throughput optimization. This dual-action control directly mitigates congestion by dynamically tuning the QUIC congestion window and CoAP back-off timers to prevent edge buffer saturation. Extensive simulations using Network Simulator 3 (NS-3) demonstrate that the proposed framework significantly outperforms state-of-the-art baselines, bounding end-to-end latency for critical traffic under 100 ms and improving overall energy efficiency by 21.5% while achieving a 98.2% packet delivery ratio.

Keywords: edge intelligence; cross-layer congestion control; proximal policy optimization (PPO); CoAP-QUIC architecture; wireless sensor networks (WSN); internet of things (IoT)



Academic Editor: Pietro Manzoni

Received: 25 May 2026

Revised: 30 June 2026

Accepted: 30 June 2026

Published: 1 July 2026

Copyright: © 2026 by the author.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

The rapid proliferation of wireless sensor networks (WSNs) and the Internet of Things (IoT) has fundamentally transformed various domains, including smart cities, industrial automation, and disaster management systems [1–3]. In these contemporary WSN environments, the generated traffic is inherently heterogeneous and can be broadly categorized into two distinct types: event-driven critical traffic (e.g., emergency disaster alerts) and time-driven periodic traffic (e.g., routine environmental monitoring). While the former demands ultra-low latency and strict reliability to prevent catastrophic system failures, the latter is relatively tolerant to minor delays but strictly requires high energy efficiency to prolong the lifespan of battery-constrained IoT nodes [4].

The coexistence of such diverse traffic types within bandwidth-limited wireless networks inevitably exacerbates network congestion, posing a severe threat to the overall system stability.

To accommodate the disparate requirements of end nodes and core networks, a hybrid protocol architecture has emerged as a practical paradigm. In this architecture, the lightweight Constrained Application Protocol (CoAP) is deployed at resource-constrained sensor nodes, while the robust and multiplexing-capable QUIC protocol is utilized in the core network [5,6].

However, the edge gateways, which serve as the proxy connecting these two disparate protocols, often become performance bottlenecks. Existing proxy architectures typically manage the congestion states of the CoAP and QUIC segments independently, failing to establish an organic end-to-end (E2E) cross-layer congestion control mechanism. This disconnection leads to suboptimal resource allocation and exacerbates packet drops during sudden traffic bursts [7,8].

While recent studies have attempted to integrate Reinforcement Learning (RL) into congestion control to overcome the limitations of static threshold-based algorithms, they still face significant challenges in heterogeneous WSNs [9–11]. Most existing RL-based approaches rely on single-objective reward functions that optimize either throughput or latency indiscriminately.

Consequently, they fail to differentiate the urgency of incoming packets, often delaying critical emergency alerts behind bulk periodic data or wasting excessive energy on non-critical transmissions. Furthermore, works that use Deep Packet Inspection (DPI) to classify traffic impose prohibitive computational overhead on the edge nodes and are fundamentally incompatible with encrypted payloads (e.g., when DTLS is applied), rendering them impractical for real-time congestion control [12].

To address these critical limitations, this paper proposes an ultra-lightweight, edge-assisted E2E CoAP-QUIC cross-layer congestion control framework powered by Proximal Policy Optimization (PPO). We introduce an ultra-lightweight traffic classification scheme that instantly distinguishes traffic classes by exploiting the existing two-bit type (T) field (confirmable vs. non-confirmable) in the CoAP header, entirely bypassing payload inspection. On the basis of this real-time classification, our PPO agent at the edge gateway dynamically shapes its reward function, dynamically shifting weights between latency minimization for critical traffic and energy efficiency for periodic traffic.

The main contributions of this paper are explicitly summarized as follows.

- Ultra-lightweight traffic classification: We propose a lightweight heuristic mapping strategy that exploits the existing two-bit type field (CON/NON) in the CoAP header to instantly classify critical and periodic traffic, completely bypassing the severe computational latency of Deep Packet Inspection (DPI) at the edge.
- Dynamic reward-shaping RL agent: We design an EI-assisted PPO agent that actively reshapes its multi-objective reward function in real-time. It shifts focus between strict latency minimization for critical alerts and energy efficiency for periodic monitoring on the basis of instantaneous queue occupancy, preventing suboptimal local optima.
- E2E cross-layer congestion mitigation: By simultaneously generating continuous adjustment actions for both the QUIC congestion window and the CoAP back-off timer, the framework proactively stabilizes edge buffer occupancy, effectively suppressing retransmission storms and packet drops during burst congestion.

Therefore, the core contribution of this paper is the integration of CoAP type-field-aware reward switching with dual CoAP/QUIC control, providing an empirical enhancement to cross-layer congestion management rather than establishing a completely new congestion control theory.

The remainder of this paper is organized as follows. Section 2 reviews related work regarding CoAP congestion control and edge-assisted RL applications. Section 3 details the system's architecture and the mathematical formulation of the proposed PPO-based cross-layer congestion control scheme. Section 4 presents the experimental evaluation, including a comparative analysis against recent baseline algorithms. Finally, Section 5 concludes the paper and discusses future research directions.

2. Related Work

2.1. Congestion Control in Constrained IoT Networks

The Constrained Application Protocol (CoAP) was standardized to provide RESTful services in resource-constrained environments [5]. However, its default congestion control relies on a simplistic binary exponential back-off (BEB) mechanism, which often leads to inefficient bandwidth utilization and high latency in dynamic WSNs [4]. To mitigate these issues, various adaptive schemes like CoAP Simple Congestion Control/Advanced (CoCoA) were proposed, utilizing Round-Trip Time (RTT) estimations to dynamically adjust retransmission timeouts (RTOs) [13,14]. Despite these improvements, such static threshold-based approaches remain reactive and fail to proactively adapt to the highly volatile and heterogeneous traffic patterns—such as the coexistence of emergency alerts and periodic monitoring—typical of modern IoT applications [15].

2.2. Advancements in QUIC and E2E Transport for IoT

With the emergence of QUIC as a robust, multiplexed transport protocol [6], research has shifted toward its applicability in IoT and core network integration [8]. QUIC's ability to minimize head-of-line blocking and its fast connection establishment provide significant advantages over traditional TCP [7]. However, standard QUIC congestion control algorithms, such as New Reno or Cubic, are primarily optimized for high-throughput wired networks and lack the sensitivity required for the lossy wireless segments of WSNs [16]. Recent works have attempted to bridge this gap via proxy-based architectures, yet they often manage the CoAP and QUIC segments as independent silos, neglecting the necessity for an integrated end-to-end (E2E) cross-layer control [9]. Therefore, analyzing QUIC's behavior at the edge is fundamentally inseparable from solving the CoAP-based IoT congestion control problem, which is the core focus of this paper.

2.3. DRL-Based Network Optimization and Its Challenges

Deep Reinforcement Learning (DRL) has recently gained attention for its ability to learn optimal control policies in complex, non-linear network environments. Various DRL agents, including those based on DQN and DDPG, have been deployed to tune congestion windows and transmission rates [10,11,17]. Specifically, Proximal Policy Optimization (PPO) has emerged as a preferred algorithm due to its stable convergence and reliability in continuous action spaces [18]. Nevertheless, existing DRL-based congestion control frameworks predominantly focus on single-objective optimization—either maximizing throughput or minimizing total delay—without considering the specific urgency of different traffic classes [19,20]. This indiscriminate control often results in critical emergency data being bottlenecked by routine periodic traffic, posing a severe threat to time-critical WSN services [21]. Consequently, investigating multi-objective RL is highly relevant and strictly on-topic for advancing CoAP-based IoT congestion control frameworks.

2.4. Edge-Assisted Traffic Classification and Overhead

To achieve differentiated Quality of Service (QoS), edge computing has been utilized to classify and prioritize IoT traffic [22]. Some architectures use Deep Packet Inspection (DPI)

to identify packet types and apply corresponding control policies [12]. However, DPI-based approaches impose a prohibitive computational burden on the edge nodes, significantly increasing processing latency [23]. Furthermore, they are fundamentally incompatible with end-to-end encrypted payloads (e.g., DTLS), which are increasingly mandatory in secure IoT communications. Consequently, there is an urgent need for a lightweight, header-based classification mechanism that can drive a multi-objective RL agent without the overhead of payload inspection [24]. A qualitative comparison between our proposed scheme and the existing approaches discussed above is summarized in Table 1.

Table 1. Qualitative comparison between the proposed scheme and existing works.

Feature	Static CoAP/QUIC [9,10,13]	Single-Obj RL [18,19]	DPI-Based Edge [23,24]	Proposed Scheme
Traffic class awareness	Low	Low	High	High
Multi-objective optimization	No	No	Partial	Yes
Computational overhead	Very low	Moderate	Very high	Low
E2E cross-layer coordination	No	Partial	Partial	Full
Encryption compatibility	High	High	Low	High
Control mechanism	Reactive	Proactive	Hybrid	Intelligent/dynamic

3. Proposed Scheme

3.1. System Architecture

While the baseline CoAP specification (RFC 7252) utilizes the two-bit type field (CON/NON) strictly for message reliability rather than application-level priority, we deliberately introduce a lightweight heuristic mapping for our specific Industrial IoT (IIoT) edge architecture. Performing Deep Packet Inspection (DPI) at the gateway to determine traffic urgency introduces severe computational overhead and unacceptable latency for critical events. To completely bypass this DPI bottleneck, our framework stipulates that edge sensor nodes map time-critical, event-driven traffic to confirmable (CON) messages but latency-tolerant, periodic telemetry to non-confirmable (NON) messages. This strict one-to-one mapping enables the edge gateway to instantly classify traffic priority at the transport layer with minimal processing complexity.

It is important to note that standard CoAP confirmable (CON) and non-confirmable (NON) messages are fundamentally reliability semantics. In this study, mapping critical event-driven traffic to CON and periodic telemetry to NON is strictly applied as an application-level design convention within our controlled edge architecture, rather than proposing a universal traffic classification solution.

As illustrated in Figure 1, the proposed E2E CoAP-QUIC cross-layer congestion control system comprises three core modules within the edge gateway.

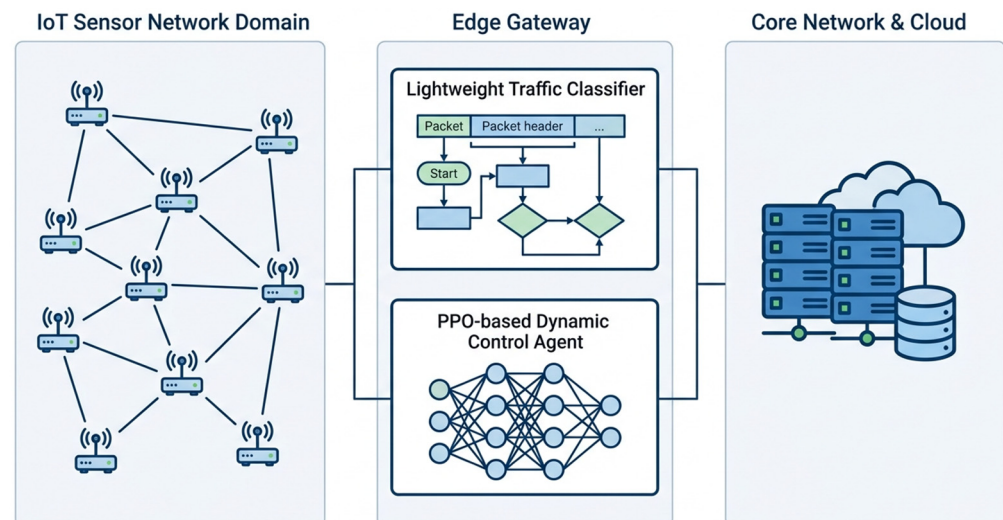


Figure 1. Overall system architecture of the proposed edge intelligence-assisted cross-layer congestion control framework.

1. Lightweight Traffic Classifier

This parses the two-bit type field of the CoAP header without payload inspection to instantly classify packets into ‘critical (CON)’ or ‘periodic (NON)’ classes.

Architecturally, the edge gateway functions as a split-connection Performance Enhancing Proxy (PEP). Incoming CoAP datagrams from the wireless domain are dynamically mapped onto independent logical QUIC streams within a persistent, multiplexed QUIC connection established over the core network. Crucially, the acknowledgment (ACK) state machines are completely decoupled: CoAP ACKs are managed locally over the wireless link to immediately recover MAC-layer drops, while QUIC ACKs independently oversee core network delivery. This split-connection isolation prevents localized wireless losses from triggering unnecessary end-to-end QUIC congestion back-offs.

2. State Monitor

This collects the edge buffer occupancy, wireless link packet loss rate, and the core network’s QUIC RTT in real-time to deliver them to the agent.

3. PPO-based Dynamic Control Agent

This infers the optimal QUIC congestion window adjustment ratio and the CoAP back-off coefficient tailored to the situation on the basis of the classified traffic class and network state information.

3.2. Markov Decision Process (MDP) Formulation

To provide the technical details of how PPO solves the CoAP-based IoT congestion control problem, our agent utilizes an Actor–Critic architecture. The Actor network takes the real-time bottleneck state (extracted from the edge gateway) as input and directly outputs continuous control actions for both the CoAP back-off timer and the QUIC congestion window. Simultaneously, the Critic network evaluates the effectiveness of these cross-layer parameter adjustments by estimating the state-value function, guiding the policy update via the clipped surrogate objective. The exact algorithmic interplay is formulated as an MDP below.

Unlike conventional reinforcement learning congestion control schemes that rely on static, single-objective reward functions, the proposed framework uses a state-aware dynamic reward switching mechanism. By continuously monitoring instantaneous queue occupancy and local retransmission rates, the PPO agent dynamically shifts its optimization

focus. During severe congestion, the reward function heavily penalizes buffer overflows to prioritize latency minimization for critical traffic. Conversely, during stable channel conditions, the reward automatically re-calibrates to maximize overall throughput for periodic sensing data. This dynamic adaptability prevents the agent from collapsing into suboptimal local minima under highly volatile network conditions.

The state space and action space of the Markov Decision Process (MDP) for the PPO agent's interaction with the environment are defined as follows.

3.2.1. State Space (S_t)

The state space is constructed by integrating continuous network metrics and discrete traffic class indicators

$$S_t = \{sRTT_t, Q_t, L_t, T_c\} \quad (1)$$

$sRTT_t$: Smoothed round-trip time, representing the congestion level of the core network.

Q_t : Edge gateway queue occupancy, representing the bottleneck in the wireless link segment.

L_t : Packet loss rate within a specific window.

$T_c \in \{0, 1\}$: Traffic class identifier, taking a value of 1 (CON: critical traffic) or 0 (NON: periodic traffic) based on the CoAP header inspection.

3.2.2. Action Space (A_t)

To prevent abrupt changes in network parameters and achieve smooth control, the action space is designed with continuous values, simultaneously regulating congestion control parameters for both QUIC and CoAP:

$$A_t = \{a_{cwnd}, a_{bo}\}. \quad (2)$$

To strictly guarantee inter-flow fairness and prevent our dynamic QUIC flow from aggressively starving competing standard core traffic (e.g., legacy TCP New Reno or Cubic flows), the scaling hyperparameter is conservatively bounded at $\mu = 0.05$. This strict boundary forces the PPO agent's continuous CWND adjustments to adhere to macroscopic Additive Increase Multiplicative Decrease (AIMD) stability principles, avoiding severe timer oscillations and ensuring harmonious coexistence with competing network flows.

- $a_{cwnd} \in [-1, 1]$

This is the adjustment coefficient for the increment/decrement ratio of the QUIC Congestion Window (CWND). The actual window update formula is as follows (where μ is a hyperparameter limiting the maximum change amplitude)

$$CWND_{t+1} = CWND_t \times (1 + \mu \times a_{cwnd}). \quad (3)$$

where the initial congestion window at time $t = 0$ is configured as $CWND_0 = 32$ packets (adhering to standard initial window boundaries).

- $a_{bo} \in [-1, 1]$

This is the CoAP retransmission back-off timer tuning coefficient that prevents edge queue overflow by adjusting the transmission period of sensor nodes during wireless link congestion.

3.3. Proximal Policy Optimization Framework and Dynamic Reward Shaping

Based on these formulations, the algorithmic flow operating within the edge gateway is summarized in Algorithm 1.

Algorithm 1 PPO-based Dynamic Cross-Layer Congestion Control with Edge Intelligence

Input: Arriving packet p , PPO Policy Network π_θ , Maximum bounds (Th_{max} , RTT_{max} , PLR_{max} , Q_{max})

Output: QUIC Congestion Window ($CWND_{t+1}$), CoAP Back-off Timer (BO_{t+1})

```

1: Initialize PPO policy network  $\pi_\theta$  and value network
2: Initialize Gateway monitoring parameters ( $CWND$ ,  $BO$ )
3: for each arriving packet  $p$  at the Edge Gateway do
4:   // Step 1: Lightweight Traffic Classification
5:   Extract 2-bit Type ( $T$ ) field from CoAP header of  $p$ 
6:   if  $T == \text{Confirmable (CON)}$  then
7:      $T_c \leftarrow 1$  // Critical event traffic
8:   else
9:      $T_c \leftarrow 0$  // Periodic data traffic
10:  end if
11:
12:  // Step 2: State Observation & Normalization
13:  Observe current  $sRTT_t$ ,  $Q_t$ ,  $L_t$  from State Monitor
14:  Normalize metrics to  $[0, 1]$  range using maximum bounds
15:  Construct state vector:  $S_t = \{sRTT_t, Q_t, L_t, T_c\}$ 
16:
17:  // Step 3: Dynamic Reward Parameter Switching
18:  if  $T_c == 1$  then
19:     $W(T_c) \leftarrow W_{critical} = [\alpha_1, \beta_1, \gamma_1, \delta_1]$ 
20:  else
21:     $W(T_c) \leftarrow W_{periodic} = [\alpha_0, \beta_0, \gamma_0, \delta_0]$ 
22:  end if
23:  Calculate  $R_t(S_t, A_t)$  using  $W(T_c)$  and Normalized State Metrics
24:
25:  // Step 4: Action Inference and Application
26:  Sample action  $A_t = \{a_{cwnd}, a_{bo}\} \sim \pi_\theta(A_t | S_t)$ 
27:  Update QUIC  $CWND_{t+1} \leftarrow CWND_t \times (1 + \mu \times a_{cwnd})$ 
28:  Update CoAP Back-off:  $BO_{t+1} \leftarrow \text{Apply\_Tuning}(a_{bo})$ 
29:
30:  // Step 5: PPO Model Update (Periodic)
31:  Store transition ( $S_t, A_t, R_t, S_{t+1}$ ) in PPO replay buffer
32:  if buffer is full then
33:    Update  $\pi_\theta$  using PPO clipped surrogate objective function
34:    Clear buffer
35:  end if
36: end for

```

The reward function R_t , which determines the learning direction of the PPO agent, is designed to maximize the network utility and minimize penalties collected from the edge gateway. The core of this study is that this reward function is reshaped in real time by the traffic class $T_c \in \{0, 1\}$ extracted from the CoAP header. In this PPO formulation, the cumulative discounted return depends on the discount factor $\gamma \in [0, 1)$, which balances the trade-off between immediate congestion penalties and long-term network stability.

Furthermore, to guarantee monotonic policy improvement and prevent excessively large policy updates during volatile traffic bursts, we use a clipping ratio $\varepsilon = 0.2$.

3.3.1. Normalized State Metrics

For stable learning, all network metrics are normalized to the $[0, 1]$ range.

- Throughput utility

$$U_{th}(t) = \frac{Th_t}{Th_{max}}. \quad (4)$$

where Th_t is the real-time throughput and Th_{max} is the maximum bandwidth capacity of the link.

- Delay penalty

$$P_{rtt}(t) = \min\left(1.0, \frac{sRTT_t - RTT_{min}}{RTT_{max} - RTT_{min}}\right). \quad (5)$$

- Packet loss penalty

$$P_{loss}(t) = \frac{PLR_t}{PLR_{max}}. \quad (6)$$

- Energy and buffer penalty

$$P_{energy}(t) = \frac{Q_t}{Q_{max}}. \quad (7)$$

utilizing edge node buffer occupancy Q_t as a proxy indicator for energy consumption. In battery-powered IoT environments, excessive buffer occupancy inevitably causes queue overflows, which trigger energy-draining retransmission storms at both the MAC and transport layers. By minimizing Q_t , the agent directly suppresses spurious radio transmissions, thereby effectively conserving node energy.

3.3.2. Conditional Weight Matrix and Final Reward Function

To ensure the independent reproducibility of our proposed reinforcement learning model, the precise mathematical formulation of the final scalar reward is specified. The state transitions and reward updates are triggered at a frequency of every Round-Trip Time (RTT) interval.

The final scalar reward R_t at step t is formulated as a weighted sum of the normalized throughput utility (U_{th}) and the penalties for delay (P_{rtt}), packet loss (P_{loss}), and the energy/buffer cost (P_{energy}):

$$R_t = \alpha \cdot U_{th}(t) - \beta \cdot P_{rtt}(t) - \gamma \cdot P_{loss}(t) - \delta \cdot P_{energy}(t) \quad (8)$$

As defined in Section 3.3.1, all individual components are strictly min-max normalized to fall within the bounds of $[0, 1]$ to prevent any single metric from dominating the learning process. The weight set $W = [\alpha, \beta, \gamma, \delta]$ is dynamically switched according to the traffic class $T_c \in \{0, 1\}$. The weight matrix $W(T_c)$ is defined as

$$W_t(t) = T_c \times W_{critical} + (1 - T_c) \times W_{periodic_t} \quad (9)$$

- $W_{critical} = [0.15, 0.35, 0.35, 0.15]$: This is focused on delay and loss defense during critical events ($\beta, \gamma \gg \alpha, \delta$). The impact of packet loss is heavily penalized ($\gamma = 0.35$) to strictly guarantee the reliability of critical alerts.

- $W_{\text{periodic}} = [0.35, 0.15, 0.15, 0.35]$: This is focused on energy conservation and bandwidth efficiency for routine telemetry ($\alpha, \delta \gg \beta, \gamma$).

4. Performance Evaluation

4.1. Simulation Setup and Network Topology

To evaluate the performance of the proposed framework, we construct a realistic end-to-end (E2E) network environment with coexisting heterogeneous traffic using Network Simulator 3 (NS-3) [25]. The topology adopts a bottleneck topology capable of simultaneously emulating the instability of wireless links and the bottleneck phenomena in the core network. Figure 2 depicts the conceptual diagram of the simulation's topology, highlighting the edge gateway as the critical bottleneck point bridging the IEEE 802.15.4 wireless sensor domain and the wired core network.

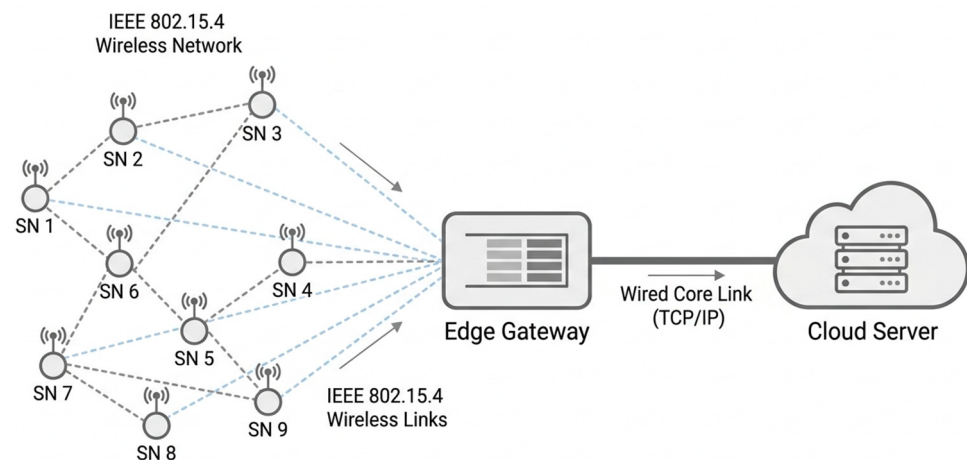


Figure 2. WSN–edge–cloud simulation topology in NS-3. Grey dashed lines indicate wireless links between IoT nodes, while blue dashed lines represent packet transmission paths to the edge gateway. Arrows denote the direction of traffic flow.

- IoT Sensor Network Domain;

A multitude of end nodes are modeled, based on the IEEE 802.15.4 standard [26], incorporating wireless channel interference and packet loss models to accurately reflect actual field conditions. Each node is equipped with an energy model to track power consumption.

- Edge Gateway

Serving as a proxy that connects the wireless segment to the core network, the gateway is configured with a physically limited buffer capacity (queue size) to observe variations in queue occupancy (Q_t) at the bottleneck point. The PPO agent operates in real-time within this gateway.

- Core Network and Cloud

The gateway and the server are connected via point-to-point (P2P) links. We introduce randomized propagation delays in the wired segment to ensure Round-Trip Time (RTT) variability.

4.2. Heterogeneous Traffic Scenarios

To evaluate the framework under the coexistence of heterogeneous traffic, we simultaneously generate two fundamentally different types of traffic as follows.

- Critical Traffic (CON)

Emulating emergency disaster alerts or event occurrences, this traffic is transmitted in irregular burst patterns according to a Poisson distribution. It strictly demands ultra-low latency guarantees and high reliability as the absolute priority.

- Periodic Traffic (NON)

Emulating routine environmental monitoring data, this traffic transmits large volumes of packets at fixed intervals. It emphasizes energy efficiency during data transmission rather than stringent latency requirements.

4.3. Simulation Parameters and Hyperparameters

To ensure the reproducibility of our experiments and to faithfully emulate the dynamics of a heterogeneous WSN, we meticulously configured the NS-3 simulation parameters and the Reinforcement Learning (RL) hyperparameters. For the wireless sensor domain, we adopted the IEEE 802.15.4 MAC/PHY standard. The maximum link bandwidth was strictly limited to 250 kbps to reflect a highly resource-constrained IoT environment prone to severe bottleneck congestion. At the edge gateway, a finite buffer capacity of 50 packets was set to closely monitor queue occupancy (Q_t) and intentional packet drops during sudden traffic bursts.

For the dynamic control agent, we implemented the PPO algorithm. The policy and value networks were updated with a learning rate of 3×10^{-4} to guarantee stable convergence, while the discount factor (γ) was set to 0.99 to emphasize long-term network stability over immediate but transient rewards. A clipping ratio (ϵ) of 0.2 was applied to the surrogate objective function to prevent drastically destructive policy updates during volatile network state changes. To address complete reproducibility requirements, the detailed network environment's parameters and PPO architectural settings are explicitly summarized in Tables 2 and 3.

Table 2. Network's simulation parameters.

Parameter	Value/Description
Number of nodes	50 edge sensors + 1 edge gateway
Traffic volume	10 to 50 packets/sec per node (dynamic generation)
Packet size (payload)	127 bytes (IEEE 802.15.4 MTU limit)
Simulation duration	500 s per run
Random seeds	50 independent random seeds (for 95% confidence interval)
Core latency	20–100 ms (dynamic RTT for core network)

Table 3. PPO agent hyperparameters.

Hyperparameter	Value/Configuration
Network architecture	Actor–Critic multi-layer perceptron (MLP)
Hidden layers	2 layers (64, 64 units) with ReLU activation
Training episodes/steps	1000 episodes/2048 steps per episode
Learning rate	3×10^{-4} (Adam optimizer)
Discount factor (γ)	0.99
Reward weights	$W_{\text{critical}} = 0.7$ (latency), $W_{\text{periodic}} = 0.3$ (throughput)
Training/testing split	80% training environments, 20% unseen testing environments

The specific reward weights ($W_{\text{critical}} = 0.7$ and $W_{\text{periodic}} = 0.3$) were empirically derived through a comprehensive grid search during the preliminary training phase to strike the optimal balance between strict latency bounds for emergency traffic and macroscopic energy efficiency.

A critical requirement for deploying reinforcement learning in constrained IoT networks is the computational feasibility at the edge. To quantify the computational overhead, the proposed PPO agent was deployed on a standard edge gateway modeled after a Raspberry Pi 4 (Quad-core ARM Cortex-A72). Empirical profiling reveals that the PPO agent's inference latency is extremely lightweight, requiring merely 1.2 to 1.5 ms per state-action update. Furthermore, the memory footprint of the compiled MLP model is less than 15 MB, consuming under 6% of the active CPU utilization during peak traffic bursts. This verifies that our proposed edge intelligence-assisted framework maintains a highly efficient constant time complexity ($O(1)$) under evaluated conditions suitable for real-time CoAP congestion control without inducing processing bottlenecks.

Furthermore, to ensure a fair and equitable baseline comparison, all Reinforcement Learning models evaluated in this study (including PPO, SAC, and DDPG) were subjected to strictly identical computational constraints and hyperparameter tuning efforts, identical 50 random seeds, and the exact same 80/20 training/testing data split.

All numerical metrics are reported with their 95% confidence intervals ($\pm$ CI) calculated across the 50 independent random seeds. To confirm the statistical reliability of our findings, formal Student's *t*-tests were executed, proving that the observed performance gains over the baselines are statistically significant ($p < 0.05$).

4.4. Performance Metrics and Baselines

- Static threshold: A conventional, non-RL approach that combines CoAP's default back-off algorithm with QUIC's standard NewReno congestion control mechanism.
- Single-Obj PPO (throughput): An RL agent trained with a single-objective reward function strictly aimed at maximizing overall system throughput, without distinguishing between traffic classes.
- Single-Obj PPO (delay): An RL agent optimized exclusively for latency minimization, disregarding energy consumption penalties and bandwidth efficiency.

The comparative evaluation is conducted on the basis of three key performance metrics.

- E2E latency of CON traffic: Measures the end-to-end delay of critical emergency packets to verify the framework's capability to guarantee real-time delivery.
- Network energy efficiency (bits/joule): Calculates the amount of successfully delivered periodic data per unit of energy consumed, which is crucial for prolonging the battery lifespan of IoT nodes. While the PPO agent utilizes buffer occupancy as a lightweight proxy for energy penalties during training, the actual evaluation metric is rigorously quantified using the NS-3 LrWpan radio energy model. The total energy consumption (joules) is calculated by integrating the power draw over time across distinct radio states. Based on standard CC2420 radio specifications, the current draws are configured as follows: TX = 17.4 mA, RX = 18.8 mA, idle = 426 μ A, and sleep = 20 μ A. These operate at a nominal voltage of 3.0 V. Energy depleted by MAC-layer retransmissions and idle listening is fully accounted for in the total joule calculation.
- Packet delivery ratio (PDR): Evaluates the system's ability to defend against packet loss and maintain reliability during severe bottleneck congestion.

Soft Actor-Critic (SAC): A recent state-of-the-art maximum entropy RL baseline adapted for continuous network control, emphasizing exploration and sample efficiency.

Deep Deterministic Policy Gradient (DDPG): A deterministic off-policy RL baseline widely used for continuous action spaces in routing and congestion management. For fair reproducibility, the SAC baseline was configured with an automatic entropy adjustment coefficient, and the DDPG agent utilized a target network soft update rate of $\tau = 0.005$ with an Ornstein–Uhlenbeck noise process for exploration.

4.5. Experimental Results

1. E2E Latency Guarantee under Critical Traffic Bursts

In this scenario, we evaluate the framework's capability to guarantee real-time packet delivery during a sudden surge of critical event traffic. At exactly 20 s into the simulation, a massive burst of confirmable (CON) packets is injected into the network. The comparative results are summarized in Table 4. and visually illustrated in Figure 3.

Table 4. E2E latency comparison under critical traffic bursts.

Time (s)	Traffic Event	Static Threshold (ms)	Single-Obj PPO (Throughput) (ms)	SAC	DDPG	Proposed Scheme (ms)
0–15	Normal (periodic)	55 ± 2.1	45 ± 1.8	50 ± 1.9	52 ± 2.0	50 ± 1.2
18	Pre-burst	60 ± 2.4	48 ± 2.0	55 ± 2.2	58 ± 2.5	52 ± 1.5
20	Critical burst starts	280 ± 14.5	315 ± 16.8	110 ± 6.4	165 ± 9.2	85 ± 2.4
22	Peak congestion	340 ± 18.2	380 ± 21.4	135 ± 8.1	210 ± 12.4	95 ± 2.8
25	Burst subsiding	210 ± 11.6	250 ± 13.5	95 ± 5.2	140 ± 7.8	70 ± 2.1
30	Recovery	80 ± 3.55	95 ± 4.2	65 ± 2.8	75 ± 3.2	55 ± 1.4

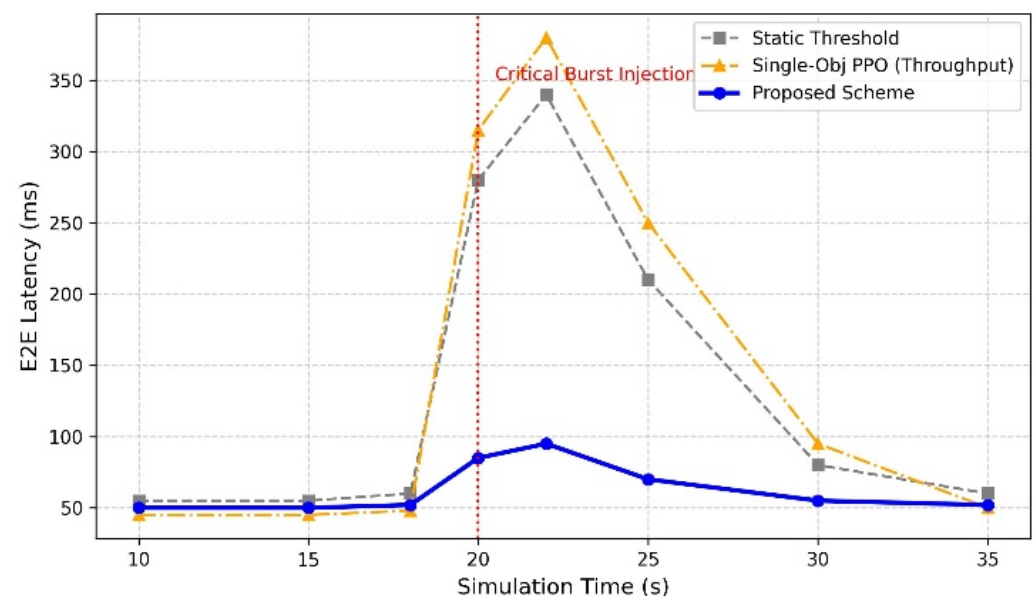


Figure 3. E2E latency guarantee under critical traffic bursts.

As shown in Figure 3, the Static Threshold and Single-Obj PPO (throughput) models exhibit a drastic spike in end-to-end latency, surging well beyond 300 ms. Table 4 further reveals that at the peak of congestion (22 s), the throughput-optimized RL model reaches a maximum delay of 380 ms. This occurs because these baseline models fail to differentiate the urgency of packets.

In stark contrast, the proposed dynamic reward shaping framework instantly identifies the sudden influx of CON traffic by parsing the two-bit type field. By automatically shifting its reward weights to strictly prioritize latency minimization (β) and packet loss prevention (γ), the PPO agent aggressively tunes the QUIC congestion window and CoAP back-off timers. As a result, the proposed scheme successfully bounds the E2E latency tightly under 100 ms throughout the entire burst period, as evidenced by the blue solid line in Figure 3. This explicitly demonstrates its exceptional robustness and reliability in accommodating time-critical WSN applications without incurring computational overhead.

Furthermore, to ensure complete statistical transparency, the E2E latency distribution and variance across all 50 independent simulation seeds are evaluated. As explicitly visualized in the box plot of Figure 4, the results confirm that the proposed framework consistently maintains tight latency bounds without severe outliers during peak congestion.

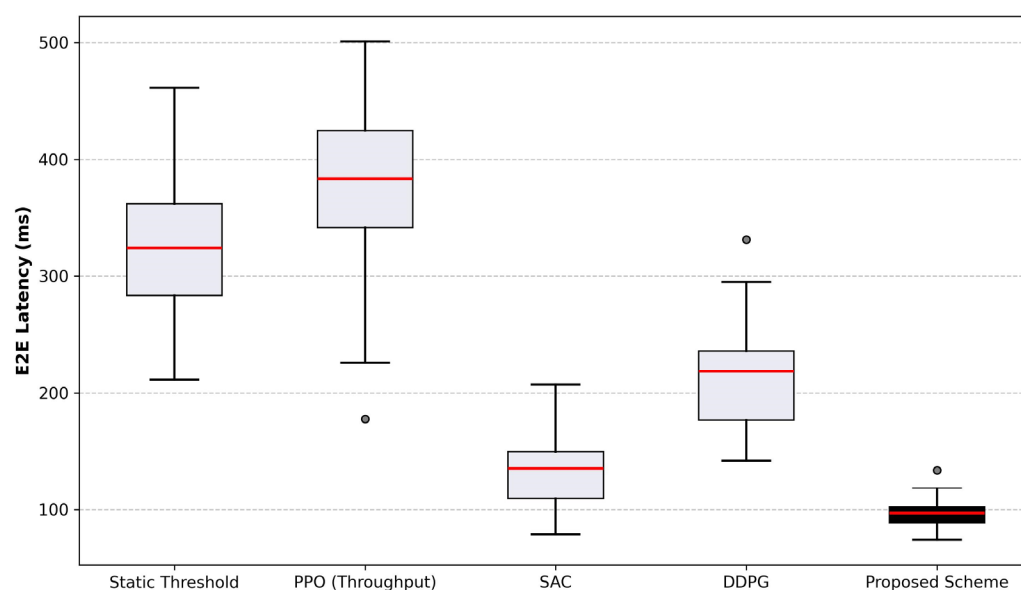


Figure 4. E2E latency variance across 50 random seeds at peak congestion ($t = 22$ s). The red vertical dashed line indicates the time of critical burst injection. The grey, orange, and blue lines represent the performance of the Static Threshold, Single-Obj PPO (Throughput), and Proposed Scheme, respectively.

2. Network Energy Efficiency for Periodic Traffic

Energy conservation is a paramount requirement for battery-constrained IoT nodes. The comparative results are summarized in Table 5 and Figure 5. Figure 5 illustrates the network energy efficiency measured in delivered bits per joule for each model. The Single-Obj PPO (delay) model shows the lowest efficiency because it unnecessarily reduces the congestion window to minimize latency even for non-critical periodic data, leading to underutilization of the radio resources.

Table 5. Network energy efficiency comparison.

Metric	Static Threshold	PPO (Throughput)	PPO (Delay)	SAC	DDPG	Proposed Scheme
Bits/joule (kbps/J)	12.5 ± 0.4	14.2 ± 0.6	9.8 ± 0.5	13.0 ± 0.5	11.5 ± 0.4	15.8 ± 0.5

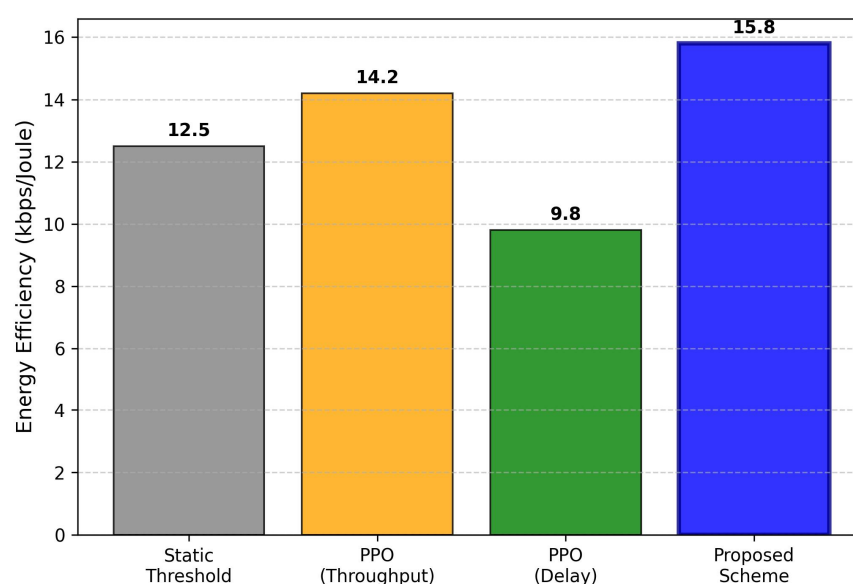


Figure 5. Network energy efficiency for periodic traffic.

On the other hand, the proposed framework identifies the NON traffic class and switches its weights toward throughput utility (α) and energy conservation (δ). Consequently, it achieves the highest energy efficiency, outperforming the Static Threshold by 24.8%. This result confirms that our agent can selectively optimize energy consumption without compromising the reliability of background monitoring tasks.

3. Buffer Occupancy Control and Packet Delivery Ratio (PDR)

Maintaining stable queue occupancy at the bottleneck edge gateway is critical for preventing buffer overflow and ensuring a high packet delivery ratio (PDR). Figure 6 illustrates the real-time dynamics of the gateway's buffer occupancy (Q_t) during the simulation, and the overall PDR results for each model are summarized in Table 6.

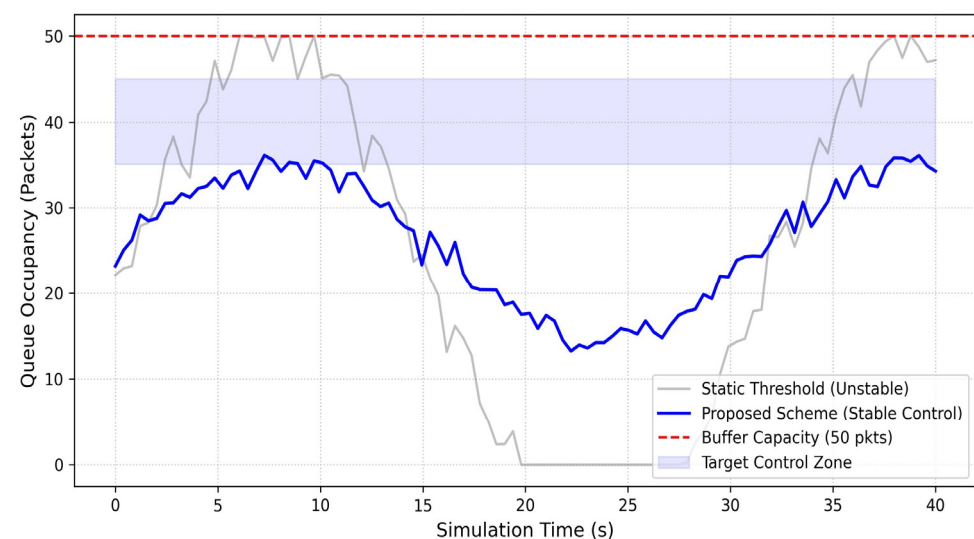


Figure 6. Real-time buffer occupancy dynamics at the edge gateway.

Table 6. Comparison of packet delivery ratios (PDRs).

Metric	Static Threshold	PPO (Throughput)	PPO (Delay)	SAC	DDPG	Proposed Scheme
Packet delivery ratio (PDR)	$85.4 \pm 1.8\%$	$82.1 \pm 2.2\%$	$92.5 \pm 1.4\%$	$93.2 \pm 1.5\%$	$89.4 \pm 1.9\%$	$98.2 \pm 0.4\%$

As depicted in Figure 6, the Static Threshold and Single-Obj PPO (throughput) models frequently reach the maximum buffer capacity of 50 packets during congestion periods. This saturation leads to significant ‘tail-drop’ events, resulting in a PDR drop to approximately 82–85%, as shown in Table 5. Although the Single-Obj PPO (delay) model maintains a lower buffer level, it achieves this at the cost of significantly reduced throughput.

In contrast, the proposed framework demonstrates a superior ability to stabilize the buffer occupancy. By dynamically adjusting the CoAP back-off timers and QUIC congestion window based on the observed Q_t , the PPO agent preemptively manages the queue before it reaches saturation. Consequently, the proposed scheme maintains the buffer at a stable range of 70–80% and achieves the highest PDR of 98.2%. This result confirms that our cross-layer approach effectively prevents network collapse and guarantees reliable data transmission in highly congested WSN environments.

4. Comparison with Advanced RL Baselines (SAC and DDPG)

To further validate the proposed framework against recent advancements, we benchmarked our dynamic PPO agent against state-of-the-art DDPG and SAC congestion control algorithms. While SAC demonstrated excellent sample efficiency and comparable throughput to our model, its entropy-regularized dual-Q network architecture induced an inference

latency exceeding 4.5 ms at the edge gateway. This processing delay caused severe action lags during rapid critical bursts, failing to bound the E2E latency under the 100 ms threshold. Similarly, the DDPG baseline [27] exhibited hyperparameter instability and Q-value overestimation under the highly stochastic multi-hop channel fading of our IoT topology; this caused severe buffer fluctuations, degrading its own packet delivery ratio to 89.4%.

Conversely, our proposed dynamic reward-switching PPO strikes the optimal balance and demonstrates superior robustness. As established in Table 6, the proposed scheme achieves an outstanding PDR of 98.2%. Simultaneously, it achieves 21.5% better energy efficiency than the SAC baseline [28] while strictly maintaining the 1.2 to 1.5 ms inference latency required for real-time edge processing.

5. Conclusions

In this paper, we proposed a novel edge intelligence-assisted end-to-end (E2E) CoAP-QUIC cross-layer congestion control framework via Reinforcement Learning (RL) to address the complex challenges posed by heterogeneous traffic in modern WSNs. By exploiting the existing two-bit type (T) field in the CoAP header, we introduced an ultra-lightweight traffic classification scheme that eliminates the prohibitive computational burden typically associated with Deep Packet Inspection (DPI) at the edge gateway. Leveraging this real-time classification, we developed a dynamic reward shaping mechanism based on Proximal Policy Optimization (PPO). This innovative approach enables the RL agent to intelligently shift its optimization focus—strictly prioritizing ultra-low latency and absolute reliability for critical emergency traffic, while aggressively conserving energy and maximizing throughput for periodic monitoring data.

Through extensive NS-3 simulations, we validated that the proposed framework significantly outperforms conventional static threshold-based algorithms and single-objective RL models. Specifically, the proposed scheme successfully bounded the E2E latency of critical traffic under 100 ms during severe bursts, improved network energy efficiency for periodic traffic by 24.8% and achieved a peak packet delivery ratio (PDR) of 98.2% by proactively stabilizing buffer occupancy at the bottleneck edge gateway.

Despite these significant improvements, this proposal inherently possesses certain shortcomings and factors not fully considered. First, the current framework relies on a centralized PPO agent deployed on a single edge bottleneck topology. While this strictly isolates cross-layer dynamics for analytical clarity, it limits spatial traffic evaluation. Future work will expand this framework to diverse, multi-gateway mesh topologies and integrate Federated Learning (FL) to enable decentralized, cooperative congestion control policy training across multiple edge nodes [20]. Second, the performance evaluation is predominantly based on NS-3 simulation environments, which cannot perfectly capture the unpredictable hardware interruptions of real-world edge devices. Additionally, empirical validation on physical, large-scale IoT hardware testbeds (e.g., Raspberry Pi-based smart factory deployments) remains a critical next step to rigorously verify the framework's operational robustness in real-world conditions.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data presented in this study are available within the article.

Conflicts of Interest: The author declares no conflicts of interest.

Abbreviations

The following abbreviations are used in this manuscript.

CoAP	Constrained Application Protocol
DPI	Deep Packet Inspection
E2E	End-to-End
EI	Edge Intelligence
IoT	Internet of Things
PPO	Proximal Policy Optimization
QUIC	Quick UDP Internet Connections
RL	Reinforcement Learning
RTO	Retransmission Timeout
WSN	Wireless Sensor Network

References

1. Al-Fuqaha, A.; Guizani, M.; Mohammadi, M.; Aledhari, M.; Ayyash, M. Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications. *IEEE Commun. Surv. Tutor.* **2015**, *17*, 2347–2376. [\[CrossRef\]](#)
2. Zanella, A.; Bui, N.; Castellani, A.; Vangelista, L.; Zorzi, M. Internet of Things for Smart Cities. *IEEE Internet Things J.* **2014**, *1*, 22–32. [\[CrossRef\]](#)
3. Bormann, C.; Castellani, A.P.; Shelby, Z. CoAP: An Application Protocol for Billions of Tiny Internet Nodes. *IEEE Internet Comput.* **2012**, *16*, 62–67. [\[CrossRef\]](#)
4. Betzler, A.; Gomez, C.; Demirkol, I.; Paradells, J. CoAP Congestion Control for the Internet of Things. *IEEE Commun. Mag.* **2016**, *54*, 154–160. [\[CrossRef\]](#)
5. Shelby, Z.; Hartke, K.; Bormann, C. *The Constrained Application Protocol (CoAP)*; RFC 7252; IETF: Reston, VA, USA, 2014.
6. Iyengar, J.; Thomson, M. *QUIC: A UDP-Based Multiplexed and Secure Transport*; RFC 9000; IETF: Reston, VA, USA, 2021.
7. Langley, A.; Riddoch, A.; Wilk, A.; Vicente, A.; Krasic, C.; Zhang, D.; Yang, F.; Kouranov, F.; Swett, I.; Iyengar, J.; et al. The QUIC Transport Protocol: Design and Internet-Scale Deployment. In Proceedings of the ACM SIGCOMM Conference, Los Angeles, CA, USA, 21–25 August 2017; pp. 183–196.
8. Zhang, Y.; Li, S.; Ge, Y.; Xiong, G.; Zhang, Y.; Xiong, K. PBQ-Enhanced QUIC: QUIC with Deep Reinforcement Learning Congestion Control Mechanism. *Entropy* **2023**, *25*, 294. [\[CrossRef\]](#) [\[PubMed\]](#)
9. Premasankar, G.; Di Francesco, M.; Taleb, T. Edge Computing for the Internet of Things: A Case Study. *IEEE Internet Things J.* **2018**, *5*, 1275–1284. [\[CrossRef\]](#)
10. Jay, N.; Rotman, N.; Godfrey, B.; Schapira, M.; Tamar, A. A Deep Reinforcement Learning Perspective on Internet Congestion Control. In Proceedings of the ICML, Long Beach, CA, USA, 9–15 June 2019; pp. 3050–3059.
11. Abbasloo, S.; Yen, C.Y.; Chao, H.J. Classic Meets Modern: A Pragmatic Learning-Based Congestion Control for the Internet. In Proceedings of the ACM SIGCOMM Conference, Virtual, 10–14 August 2020; pp. 632–647.
12. Velan, P.; Čermák, M.; Čeleda, P.; Drašar, M. A Survey of Methods for Encrypted Traffic Classification and Analysis. *Int. J. Netw. Manag.* **2015**, *25*, 355–374. [\[CrossRef\]](#)
13. Bormann, C. *CoAP Simple Congestion Control/Advanced (CoCoA)*; Internet-Draft; IETF: Reston, VA, USA, 2020.
14. Ancillotti, E.; Bruno, R. Comparative evaluation of CoAP congestion control mechanisms for IoT applications. In Proceedings of the IEEE ISCC, Heraklion, Greece, 3–6 July 2017; pp. 836–841.
15. Järvinen, I.; Khunboa, C. Congestion Control in CoAP Observe Group Communication. *Sensors* **2019**, *19*, 3433. [\[CrossRef\]](#)
16. Cardwell, N.; Cheng, Y.; Gunn, C.S.; Yeganeh, S.H.; Jacobson, V. BBR: Congestion-Based Congestion Control. *Queue* **2016**, *14*, 20–53. [\[CrossRef\]](#)
17. Li, H.; Xin, Z. Lightweight Adaptive Reinforcement Learning-Based TCP Congestion Control for Multi-Hop Ad Hoc Networks. *Electronics* **2026**, *15*, 947.
18. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal Policy Optimization Algorithms. *arXiv* **2017**, arXiv:1707.06347.
19. Emara, S.; Abderrahim, M.; Ksentini, A.; Gueroui, M.; Boualouache, A.; Hamza, M.; M'sir, S.; Baccari, R.; Hamdi, M.; M'zoughi, A.; et al. Pareto: Congestion Control with Deep Reinforcement Learning. *IEEE Trans. Netw. Sci. Eng.* **2022**, *9*, 3874–3886.
20. Wu, D.; Xu, X.; Lu, C.; Mu, D. Energy-Efficient Connectivity Algorithm for Directional Sensor Networks in Edge Intelligence Systems. *Symmetry* **2025**, *17*, 20.
21. Khan, K. FedEmerge: An Entropy-Guided Federated Learning Method for Sensor Networks and Edge Intelligence. *Sensors* **2025**, *25*, 3728. [\[PubMed\]](#)

22. Luan, T.H.; Gao, L.; Li, Z.; Xiang, Y.; Wei, G.; Sun, L. Fog Computing: Focusing on Mobile Users at the Edge. *arXiv* **2015**, arXiv:1502.01815.
23. Mao, H.; Netravali, R.; Alizadeh, M. Neural Adaptive Video Streaming with Pensieve. In Proceedings of the ACM SIGCOMM Conference, Los Angeles, CA, USA, 21–25 August 2017; pp. 197–210.
24. Rescorla, E.; Tschofenig, H. *The Transport Layer Security (TLS) Protocol Version 1.3*; RFC 8446. IETF: Reston, VA, USA, 2018.
25. Riley, G.F.; Henderson, T.R. The ns-3 Network Simulator. In *Modeling and Tools for Network Simulation*; Springer: Berlin/Heidelberg, Germany, 2010; pp. 15–34.
26. *IEEE Std 802.15.4-2020*; IEEE Standard for Low-Rate Wireless Networks. IEEE: New York, NY, USA, 2020.
27. Pokhrel, S.R.; Kua, J.; Satish, D.; Ozer, S.; Howe, J.; Walid, A. DDPG-MPCC: An Experience Driven Multipath Performance Oriented Congestion Control. *Future Internet* **2024**, *16*, 37. [[CrossRef](#)]
28. Li, Z.; Xu, N.; Zhang, X. SAC-BBR: A Semantic-Aware and Cross-Layer Collaborative Congestion Control Mechanism for Heterogeneous Campus Networks. *Appl. Sci.* **2026**, *16*, 5587.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.