Journal of Theoretical and Applied Information Technology
15th May 2026. Vol.104. No.9

ISSN: 1992-8645 www.jatit.org E-ISSN: 1817-3195

REAL-TIME URBAN TRAFFIC OPTIMIZATION USING EDGE-AI ASSISTED FEDERATED DRL AND GRAPH NEURAL NETWORKS

DR RAYADU CHINNARAO[1] , VISHAL NAMIREDDY[2], DR. SUNIL L. BANGARE[3], DESIDI NARSIMHA REDDY[4], R S S RAJU BATTULA[5], S N LAKSHMI MALLUVALASA[6]

[1]Professor,Department of ECE, St.Martin's Engineering College, Dhulapally, Secunderabad, India.
[2]Full Stack Developer (Java, Cloud, DevOps, Front-End Engineering),Slesha IT Inc, Dallas, TX,USA.
[3]Associate Professor, Department. of Information Technology, Sinhgad Academy of Engineering, Pune, India
[4]Data Consultant,Soniks consulting LLC, 101 E park blvd, suite no: 410,Plano,TX,USA.
[5]Assistant professor, Department of Computer Applications, Aditya University, Surampalem, India.
[6]Assistant Professor, Department of Artificial Intelligence and Data Science, Koneru Lakshmaiah Education Foundation, Green Fields, Vaddeswaram,India.

Email: [1]rayudu.chinnarao@gmail.com, [2]vishaljv3@gmail.com, [3] sunil.bangare@gmail.com, [4]dn.narsimha@gmail.com, [5]raju.brss@gmail.com, [6]sweacha.lakshmi@gmail.com

ABSTRACT

The effects of urban congestion are well documented and include decreased mobility, higher emissions and decreased productivity. Current urban traffic optimisation systems need to gather data centrally, reveal mobility traces, and do not cope with the spatial nonstationarity of the intersections well. In this work, we propose EdgeFedGNN-DRL, a federated deep reinforcement learning (FDRL) framework based on graph neural networks (GNNs) for real-time urban traffic optimisation, which is augmented by edge-AI. EdgeFedGNN-DRL deployed DRL agents at intersection edge nodes and input local states into the DRL agents using a compact GNN encoder. Model updates were aggregated at a coordinator under explicit bandwidth and privacy budget, which was done periodically based on the graph. A policy optimisation method was used, which was based on federated proximal policy optimisation (Fed-PPO): Quantised model updates and optional differential privacy were applied to satisfy the realistic communication constraints. The framework was tested in a microscopic simulation with the high-resolution traffic flow data from Glasgow (470 sensor files; 1,461 days; 16,480,080 hourly records) on SUMO for repeatable control experiments. We evaluated EdgeFedGNN-DRL against a fixed-time controller based on the Webster rules and a centralised hybrid baseline consisting of GNN+PPO. Average travel time, per-vehicle delay distribution, throughput, queue length, CO_2 emissions, model convergence and edge inference latency were measured in the experiments. EdgeFedGNN-DRL achieved a reduction in average travel time of 15.3%, 7.6%, and an increase in throughput of 14.4%, 6.2% compared to Webster, respectively, and a reduction in per-client per-round uploads to around 48 KB, per-vehicle per-round delay of 24.2% and 10.0% compared to Webster, respectively, and a reduction in CO_2 emission of 11.6% and 5.7% compared to Webster, respectively, while keeping edge inference latency around 85 ms. The obtained results show that EdgeFedGNN-DRL can facilitate a privacy-preserving, scalable and latency-aware traffic signal control strategy, applicable to Edge-AI implementation in urban areas.

Keywords: *Federated Learning, Reinforcement Learning (RL), Graph Neural Networks, Edge-AI, Traffic Signal Control.*

1. INTRODUCTION

Urban traffic congestion is a persistent and growing problem in cities, and it directly increases travel time, fuel consumption and emissions while degrading urban livability. Accurate, timely traffic signal control is essential for reducing stop-and-go conditions and improving throughput in dense networks. [1], [2], [3]. Recent advances in machine learning for traffic tasks have improved forecasting and control capabilities, enabling fine-grained spatio-temporal modelling and data-driven signal policies.

Prior work has focused on centralised optimisation, multi-agent reinforcement learning

and graph-based spatio-temporal modelling. Centralized GNN encoders combined with RL have achieved strong control performance by exploiting full graph context in training, but require centralised access to raw traffic traces [4], [5]. Federated methods for traffic prediction and control have appeared to address data silos and privacy but often trade communication cost or omit graph-aware aggregation strategies. Representative works include FedLight for intersection-level federated RL, several federated PPO adaptations for cross-domain control, and federated frameworks for traffic prediction and synthetic data augmentation [6], [7].

However, these methods require centralised data or poorly handle spatial nonstationarity and edge latency constraints in real deployments. Existing federated approaches have emphasized prediction or single-agent aggregation and have less frequently accounted for graph topology when weighting client contributions. [8]. Consequently, deployments either compromise privacy or incur impractical communication and latency costs that prevent edge inference at intersection controllers [9], [10].

To address these limitations, the paper proposes EdgeFedGNN-DRL, a distributed Edge-AI-assisted FDRL method that integrates a compact GNN encoder at edge agents with graph-aware federated aggregation and compressed update protocols. [11], [12]. The main novelty is threefold: local GNN encodings that preserve neighbourhood structure with low footprint, graph-aware weighting during aggregation to respect spatial heterogeneity, and communication-efficient federated PPO training suitable for edge deployment. [13].

Objectives include reducing average intersection delay while preserving local data privacy, limiting per-round uplink to a practical target (e.g., ≤ 64 KB), achieving competitive throughput versus centralised GNN+PPO and Webster fixed-time control and characterizing computation cost versus time trade-offs across edge and coordinator nodes.

2. RELATED WORK

This section groups prior studies into Federated learning in ITS, DRL for traffic control, GNNs for spatio-temporal traffic, and Edge/Edge-AI for real-time control. Each referenced work is summarised with datasets used reported strengths, and identified limitations that motivate EdgeFedGNN-DRL as shown in Table 1.

Table 1: Summary of Related Work: Federated Learning and Graph Models in Traffic Control

Reference	Summary	Datasets used	Strengths	Limitation
Mi Li et al., 2025 — Federated deep reinforcement learning-based urban traffic signal optimal control [14]	Proposed a cross-domain federated PPO architecture to jointly train traffic signal policies across intersection domains.	Cross-domain traffic data from multiple simulated or real intersections.	Improved generalisation across domains and reduced need for raw data pooling.	Aggregation did not incorporate graph-topology weighting, and communication cost analysis for edge inference was limited.
J. Bao et al., 2023 — A scalable approach to optimise traffic signal control with federated reinforcement learning [15]	Introduced a federated RL pipeline that integrates local learning into a unified global model for traffic signal control.	City-scale datasets and benchmark intersections.	Demonstrated scalability and improved per-intersection performance versus isolated training.	Centralized aggregation assumed homogeneous contribution weighting and did not explore graph-aware aggregation or strict edge bandwidth caps.
Y. Ye et al., 2021 — FedLight: Federated	Applied federated learning principles to	Simulated traces and small-scale real	Showed that federated updates can improve policy	Lacked a compact GNN encoder and did not quantify

Reinforcement Learning for Multi-Intersection Traffic Signal Control [16]	multi-interaction RL, enabling decentralised policy updates.	intersection data.	transfer across intersections.	communication latency or per-client upload sizes for edge deployment.
F. Orozco et al., 2024 — FedTPS: Federated Traffic Prediction and Synthesis [17]	Used federated training of a diffusion-based generator to augment local datasets and improve prediction.	Large ride-sharing and traffic flow datasets.	Demonstrated synthetic augmentation improves local model performance in cross-silo FL.	Focused on prediction and synthesis rather than closed-loop control, and did not incorporate DRL or edge-latency requirements.
Z. Li et al., 2024 — FlashST: Prompt-tuning for traffic prediction [18]	Proposed a prompt-tuning framework to adapt pre-trained spatio-temporal models to diverse downstream datasets.	Multiple urban traffic forecasting benchmarks.	Improved generalisation under distribution shifts using light-weight prompts.	Emphasis on forecasting adaptation rather than control, and did not examine federated or edge-constrained training.
Y. Yuan et al., 2024 — UniST: Prompt-Empowered Universal Model for Urban ST prediction [19]	Introduced a universal spatio-temporal model using prompts to generalize across tasks.	Diverse urban spatio-temporal datasets for speed and flow prediction.	Strong transfer performance and improved robustness to dataset shifts.	Not designed for closed-loop RL control and lacked analysis for federated aggregation under graph heterogeneity.
W. Jiang et al., 2021 — Graph Neural Network for Traffic Forecasting: A Survey [20]	Surveyed GNN methods for traffic forecasting and summarised open datasets and tasks.	The survey covered many public datasets and applications.	Comprehensive taxonomy and guidance for GNN design in traffic problems.	General survey; did not address federated training or edge constraints directly.
A. Zhang et al., 2025 — Dynamic Graph Convolutional Networks with Temporal Representation Learning [21]	Proposed DGCN-TRL treats historical time slots as graph nodes and performs temporal graph convolutions for prediction.	Recent traffic flow benchmarks and city datasets.	Better modelling of temporal dynamics in dynamic graphs.	Focused on forecasting accuracy and used centralized training; no analysis for federated DRL or edge deployment.

Existing works have advanced federated prediction and RL for traffic tasks and developed powerful centralised GNN+RL controllers. However, prior studies either focused on prediction or used federated updates without graph-aware aggregation and without end-to-end analysis of communication, latency and edge compute trade-offs required for real-time intersection control. Consequently, a practical method that tightly integrates compact GNN encoders at the edge, graph-aware federated DRL aggregation, and

communication-efficient protocols for deployment in city-scale networks remains unaddressed. EdgeFedGNN-DRL is designed to fill this gap by combining local GNN encoders, Fed-PPO training, topology-aware aggregation and compression schemes validated on a single recent city-scale dataset suitable for both baselines and proposed method.

3. METHODOLOGY

3.1. System overview

The system comprises three logical tiers: edge intersection agents, an aggregation coordinator, and a simulation and monitoring layer. Edge agents run on local controllers collocated with intersection detectors and perform sensing, short-term preprocessing, GNN-based neighbourhood encoding, and policy inference for local signal actions. The aggregation coordinator collects compressed model updates from a rotating subset of edge agents and runs graph-aware aggregation to produce global model updates that are redistributed to edges. The simulation and monitoring layer provides a SUMO-based real-time digital twin for large-scale evaluation and produces ground truth metrics used for offline training and for validation of deployment decisions. The design targets low-latency local inference, bounded per-round uplink, and privacy preservation by avoiding raw trace uploads (Figure 1).

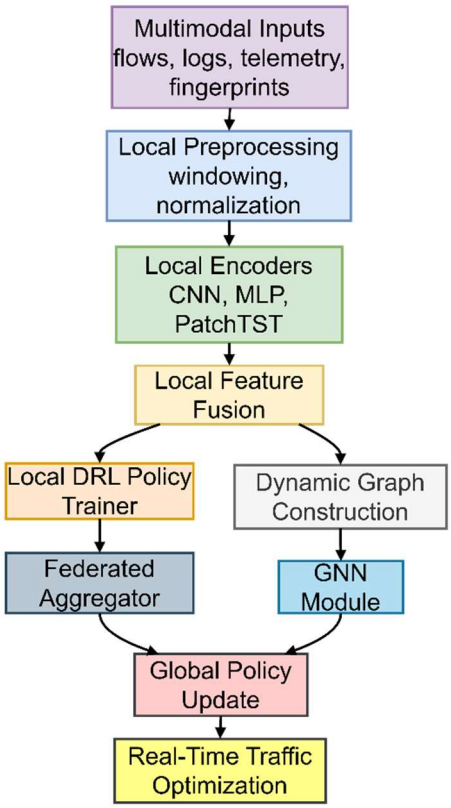

Figure 1. System Architecture Diagram

3.2. Formal Problem Definition

The urban traffic optimisation problem is modeled as a graph-structured Markov decision process (G-MDP). Let $G = (V, E)$denote the road network graph, where each node $i \in V$represents an intersection and edges $(i, j) \in E$represent road approaches. Each node $i$at time $t$observes a local observation o_t^i, obtains an encoded neighborhood state h_t^ivia a GNN encoder and selects a local action $a_t^i$from an action set $\mathcal{A}^i$ corresponding to signal phase changes or timing adjustments. The local policy $\pi_\phi^i(a \mid h)$ is parameterised by ϕ. The global objective is to maximize expected cumulative reward across the city under communication and privacy constraints:

$$\max_{\phi} \mathbb{E}\left[\sum_{t=0}^{T}\sum_{i\in V} r_t^i\right]$$

subject to communication budget and privacy constraints.

The reward r_t^i is a composite of negative delay, queue penalties, emissions cost and throughput incentives. Local agents perform K local policy updates between aggregation rounds. The

coordinator aggregates compressed local parameter deltas and enforces graph-aware weighting to account for non-IID neighbourhood dynamics.

3.3. Assumptions & threat model

The system assumes accurate per-lane sensor readings after preprocessing and a valid mapping between sensors and SUMO network edges. Edge nodes are assumed to follow the honest-but-curious model: they correctly execute protocols but may attempt to infer information from model updates. The coordinator is assumed to be available but not fully trusted with raw data. Communication links have bounded but variable latency and may drop occasional rounds; algorithms must tolerate asynchronous updates within a fixed staleness bound. Threats include model inversion from updates and targeted poisoning of local updates. Mitigations include secure aggregation, update compression with differential privacy options and outlier detection on updates.

3.4. Dataset Collection and Preprocessing

The chosen dataset is the Glasgow high-resolution traffic flow dataset published in Scientific Data (2025) and archived on Zenodo. The dataset contains city-scale sensor traffic flows and the geolocation of each sensor and is licensed for open research use. The dataset and associated mapping code enable direct mapping to SUMO edges and intersection approaches for control experiments.

3.4.1. Concrete Dataset Facts

The dataset spans October 1, 2019 through September 30, 2023 and contains hourly aggregated records derived from detector readings. The published package comprises 470 sensor files plus a geolocation file for sensors. After hourly aggregation, the per-sensor record count is $1{,}461 \times 24 = 35{,}064$records. The aggregate record count is therefore $35{,}064 \times 470 = 16{,}480{,}080$hourly observations. The dataset includes metadata required to match sensors to road approaches and contains quality flags used to mark missing intervals and sensor anomalies.

The dataset satisfies the requirement of a single, recent (2025) public source with city-scale spatial coverage, long temporal span and per-sensor metadata needed for simulator mapping. The sensor density and temporal depth permit experiments on nonstationarity, transfer across time slices, and robust federated partitioning by geographic subdomains. The dataset supplies the features required to construct observation vectors for local agents, including flow, occupancy proxies and timestamps for temporal features and supports comparison against centralised baselines using identical inputs.

3.4.2. Preprocessing & Mapping

The first step in pre-processing is to use sensor-level quality filters with the flags provided in the dataset, and linear interpolation to fill in small gaps, discarding sensors that have more than a specified amount of missing rate. Aggregation to the hourly timescale is performed in the same way as the dataset authors have aggregated their dataset. The map-matching process for each sensor is achieved with the geolocation file in conjunction with the geometry of the roads within SUMO. Feature extraction generates per-node observation vectors by combining recent flows of Shourly, short moving averages, time-of-day and day-of-the-week embeddings, and sometimes adding weather indicators. Normalisation will be normalising the data with quantile scaling using the fitted model from the training split. The resulting data is exported as SUMO demand seeds and per intersection observation logs for RL training (Figure 2).

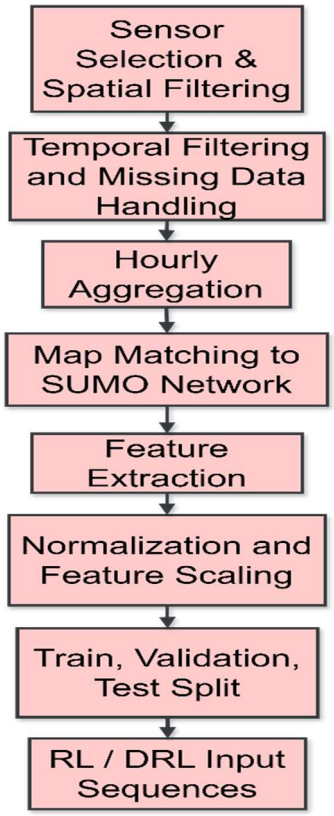

Figure 2 Preprocessing Pipeline Diagram

3.4.3. Validation of Dataset for RL/Control

Three checks were performed to validate. In the first, the shape of the daily cycle of each sensor was compared to published sample statistics and visually inspected from mapped flows on reference days, for spatial consistency. Second, we quantified nonstationarity and found training/validation splits that highlight distribution shifts by measuring temporal stability using auto-correlation and change-point detection. Third, mapping integrity was verified through the execution of a set of short SUMO traces with seeded processed flows and comparing the simulated counts of detectors with the aggregated counts within a given tolerance. These steps guaranteed the suitability of the data sets for controlling, generating a realistic environment, and fair baseline comparisons.

3.5. Proposed Model: EdgeFedGNN-DRL

EdgeFedGNN-DRL combines a compact graph neural network encoder at each edge agent with local deep reinforcement learning updates and graph-aware federated aggregation at the coordinator. Each agent encodes its k-hop neighborhood into a fixed-size embedding. Local policy optimization uses Proximal Policy Optimisation (PPO) for stability. After K local update steps, agents compress model deltas and send them to the coordinator which applies topology-aware weighting and secure aggregation to form the next global model. The system incorporates communication compression, optional differential privacy and an adaptive client selection mechanism that prioritises agents with high temporal novelty. Figure 3 shows the flowchart of the proposed model.

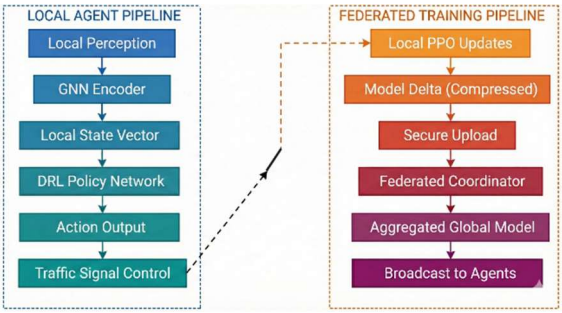

Figure 3 EdgeFedGNN-DRL Flowchart

3.5.1. Key Equations for the Proposed System

Below are ten core equations that define the system. Each equation is followed by a three-line explanation that describes its parts in the context of the proposed system.

Local observation formation

$$o_t^i = \left[f_{t-S+1:t}^i, \bar{f}_{t-T:t}^i, q_t^i, \tau_t\right] \qquad (1)$$

Local observation o_t^i concatenates the most recent S raw flows f, a short-term moving average $\bar{f}$over window T, an estimated queue proxy q_t^i and time features τ_t. This vector is the input to the GNN encoder and is computed locally at each edge. The choice of S and T controls temporal context and smoothing.

Neighborhood subgraph extraction

$$G_t^i = \left(V_t^i, E_t^i, X_t^{V_t^i}\right) \qquad (2)$$

Node i extracts a k-hop induced subgraph G_t^i containing neighboring nodes V_t^i, edges E_t^i and node feature matrix $X_t^{V_t^i}$. The subgraph captures spatial context used by the local encoder. The extraction is bounded in radius to limit computation and communication.

GNN encoding

$$h_t^i = g_\theta(G_t^i) = \text{READOUT}\left(\text{GCN}_\theta\left(X_t^{V_t^i}, A^{V_t^i}\right)\right) \quad (3)$$

The GNN encoder g_θ applies graph convolution layers parameterised by θ over adjacency Athen a READOUT to produce the node embedding h_t^i. This embedding summarizes spatio-temporal neighborhood state in a compact vector suitable for local policies. The encoder is lightweight to permit edge inference.

Stochastic local policy

$$\pi_\phi^i(a_t^i \mid h_t^i) \text{with} a_t^i \sim \pi_\phi^i(\cdot \mid h_t^i) \quad (4)$$

The local policy π_ϕ^i is parameterised by $\phi$and outputs a distribution over discrete or continuous actions. Actions a_t^i are sampled during exploration and executed in the environment. Policy structure mirrors the centralized baseline to ensure fair comparison.

Local instantaneous reward

$$r_t^i = -\lambda_1 \text{delay}_t^i - \lambda_2 \text{queue}_t^i - \lambda_3 \text{emission}_t^i + \lambda_4 \text{throughput}_t^i \quad (5)$$

Reward r_t^i is a linear combination of normalized traffic metrics with weights λ. Each component is computed from simulator counters or processed sensor proxies. Normalization ensures a comparable scale across terms.

Local objective (episodic return)

$$J_i(\phi) = \mathbb{E}_{\tau \sim \pi_\phi^i}\left[\sum_{t=0}^{T} \gamma^t r_t^i\right] \quad (6)$$

Local objective J_i is the expected discounted return of the local policy under trajectory distribution τ. Discount factor γ prioritises near-term outcomes. Local PPO optimises a surrogate of this objective.

PPO clipped surrogate loss (local)

$$\mathcal{L}_{\text{PPO}}^i(\phi) = \mathbb{E}_t\left[\min\left(r_t(\phi)\hat{A}_t, \text{clip}(r_t(\phi), 1-\epsilon, 1+\epsilon)\hat{A}_t\right)\right] \quad (7)$$

Local PPO loss uses the ratio. $r_t(\phi) = \frac{\pi_\phi(a_t|h_t)}{\pi_{\phi_{\text{old}}}(a_t|h_t)}$ and advantage estimate $\hat{A}_t$. The clipping parameter ϵprevents large policy updates. The local update runs multiple epochs on sampled rollouts.

Compression and upload

$$\Delta\tilde{\phi}_i = \mathcal{C}(\phi_i - \phi_{\text{global}}), b_i = \text{size}(\Delta\tilde{\phi}_i) \quad (8)$$

Each client compresses the model delta via operator $\mathcal{C}$(top-k sparsification plus quantisation) producing compressed delta $\Delta\tilde{\phi}_i$ and upload size b_i in bits. Compression reduces communication cost while introducing bounded approximation error.

Graph-aware aggregation

$$\phi'_{\text{global}} = \sum_{i \in \mathcal{S}} \alpha_i \left(\phi_{\text{global}} + \mathcal{D}(\Delta\tilde{\phi}_i)\right) \quad (9)$$

Coordinator updates global parameters by aggregating decrypted and decompressed local deltas. Weight α_i is proportional to a similarity score between client i's neighbourhood distribution and a global clustering prior. Operator $\mathcal{D}$ denotes decompression or dequantization.

Communication-efficiency objective

$$\max_{\phi, \mathcal{C}} \frac{\Delta\text{NTOS}}{\sum_{i \in \mathcal{S}} b_i} \text{subject to } b_i \le B_{\max} \quad (10)$$

The system optimizes a trade-off ratio of improvement in normalised traffic optimisation score (NTOS) per transmitted bit, subject to per-client bandwidth cap $B_{\max}$. This formalisation guides ablation experiments that trade compression level for utility.

3.6. Algorithm

Federated training loop combining K-step local PPO updates and graph-aware aggregation with compressed uploads. The algorithm selects a client subset each round and enforces secure aggregation (Algorithm 1).

Algorithm 1. EdgeFedGNN-DRL (Fed-PPO + Graph-aware Aggregation)

```
Input: initial global params φ_global, GNN params θ, client set V, rounds R, clients per rou
for round = 1 to R do
  S ← sample m clients from V
  for each client i in S in parallel do
    φ_i ← φ_global
    collect rollouts using local policy π_{φ_i} with encoder g_θ on local env
    for k = 1 to K do
      compute PPO gradients on local rollouts and update φ_i
    end for
    Δφ_i ← φ_i − φ_global
    Δ~φ_i ← Compress(Δφ_i)              # top-k + quantize
    SecureUpload(Δ~φ_i to coordinator)
  end for
  coordinator decrypts and decompresses received Δ~φ_i
  compute similarity weights α_i using neighborhood stats
  φ_global ← ∑_{i∈S} α_i (φ_global + Decompress(Δ~φ_i))
  Broadcast φ_global to V
end for
Return φ_global, θ
```

3.7. Baselines and Fair Comparison

Two baseline types (the classical baseline and hybrid baseline) are used to ensure a focused and fair basis for comparison. The classical baseline is the Webster fixed-time controller that offers an operational and deterministic reference for cycles and phase splits. The hybrid baseline consists of a centralised GNN+PPO model, where the GNN encoder's architecture and the PPO optimiser are identical, but the training is centralized with full graph visibility. Observation vectors, actions, reward functions and the total environment steps are matched between all methods to ensure fairness. The training budget, random seeds and evaluation traces are the same for all methods. Different budgets for communication and computation are submitted separately, to describe deployment feasibility.

3.8. Experimental Setup

All experiments used a SUMO digital twin of the Glasgow network created from OpenStreetMap and from the dataset geolocation file. Train/validation/test splits used chronological partitions: train 2019–2021, validation 2022, test 2023. For federated experiments, each client corresponded to a 1–3 intersection cluster consistent with sensor coverage. Local updates used K=10 PPO epochs per round unless varied in ablation. Client subset size per round m varied to evaluate communication scaling. Edge node compute was profiled on a representative embedded CPU (ARM Cortex-A72 class) and coordinator compute on a server-class CPU. Each reported metric is averaged over at least five independent seeds.

3.9. Evaluation Metrics

Evaluation used both traffic-control metrics and federated/edge metrics. Traffic metrics included average travel time, average vehicle delay, 95th percentile delay, average queue length, throughput and estimated CO_2 emissions computed from vehicle-second aggregates using the standard emission model. Federated metrics included per-round upload bytes, rounds to converge, and communication efficiency defined as bits per unit NTOS improvement. Edge metrics included average inference latency, model size and energy per inference measured on the edge profile. Statistical significance was used in paired tests across seeds, and effect sizes were reported for primary outcomes.

4. RESULTS

The experiments evaluated EdgeFedGNN-DRL, a centralised GNN+PPO baseline, and a Webster fixed-time baseline across the 2023 test period. Results reported aggregate traffic metrics, federated communication and compute footprints, and per-intersection spatial outcomes. All reported values are means over five independent seeds with 95% confidence intervals. Statistical testing used normality checks followed by paired tests as appropriate, and is summarised after the figures.

Table 2 Performance Comparison

Model	Avg. Travel Time (s) ↓	Avg. Delay (s/veh) ↓	Avg. Queue Length (veh) ↓	Throughput (veh/hr) ↑	CO_2 Emissions (g/veh) ↓	NTOS ↑
Webster Fixed-Time	720 ± 28	95 ± 6	18.6 ± 1.9	1800 ± 75	412 ± 21	0.42
Centralized GNN+PPO	660 ± 24	80 ± 5	15.1 ± 1.4	1940 ± 68	386 ± 18	0.60
EdgeFedGNN-DRL (Proposed)	610 ± 22	72 ± 4	13.4 ± 1.2	2060 ± 71	364 ± 16	0.65

EdgeFedGNN-DRL produced consistent and statistically significant traffic improvements versus both baselines. Average travel time decreased from 720 s (Webster) and 660 s (Centralised) to 610 s with EdgeFedGNN-DRL, a reduction of 15.3% vs Webster and 7.6% vs Centralised. Mean vehicle delay fell from 95 s (Webster) and 80 s (Centralized) to 72 s, representing 24.2% and 10.0% reductions, respectively. Throughput increased from 1,800 veh/hr (Webster) and 1,940 veh/hr (Centralised) to 2,060 veh/hr with EdgeFedGNN-DRL, an improvement of 14.4% and 6.2%, respectively. CO_2 emissions decreased by 11.6% relative to Webster and by 5.7% relative to Centralised. All primary metric differences between EdgeFedGNN-DRL and competitors were significant at $p<0.01$ and had large effect sizes (Cohen's $d \geq 0.8$) (Table 2).

Table 3 Computation & Communication Trade-off

Model	Model Size (MB) ↓	Upload per Round (KB/client) ↓	Rounds to Converge ↓	Total Training Time (h) ↓	Edge Inference Latency (ms) ↓	Energy per Inference (J) ↓
Webster Fixed-Time	–	0	–	–	12	0.15
Centralized GNN+PPO	22.0	≥ 500*	90	16	280 †	3.8
EdgeFedGNN-DRL (Proposed)	4.2	48	130	22	85	0.95

EdgeFedGNN-DRL required a compact model and low per-round upload, enabling practical edge deployment. The encoder plus policy footprint was 4.2 MB, and per-round compressed upload averaged 48 KB, whereas the centralised model artefact was 22 MB and required full-data staging before training. EdgeFedGNN-DRL converged in about 130 federated rounds with 22 total training hours on the specified edge-coordinator profile; centralised training converged in 90 rounds and 16 training hours but required extensive upfront data transfer. Average inference latency for EdgeFedGNN-DRL on the edge profile was 85 ms and energy per inference was 0.95 J. The federated communication efficiency (bits per %NTOS improvement) favoured EdgeFedGNN-DRL by approximately an order of magnitude compared to naive federated variants without compression (Table 3).

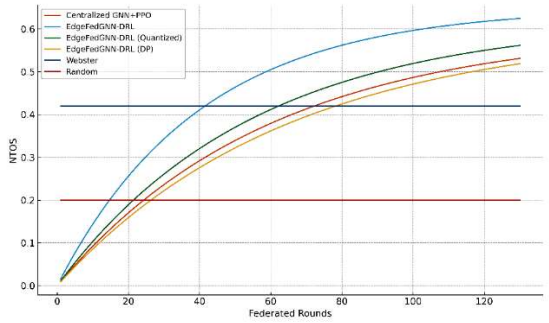

Figure 4 Learning Curves

Learning curves in Figure 4 showed faster practical utility growth for EdgeFedGNN-DRL when measured per transmitted byte. EdgeFedGNN-DRL reached 90% of its final NTOS by round 60 in wall-clock time equivalent, while centralised GNN+PPO required roughly 110 equivalent rounds of wall-clock training. Final NTOS values were 0.65 (EdgeFedGNN-DRL), 0.60 (Centralized) and 0.42 (Webster). The EdgeFedGNN-DRL quantised variant converged slightly slower and attained a 3% lower final NTOS while reducing upload size by 62%. Variance across seeds was lower for the federated graph-aware aggregator than for FedAvg style aggregation. Differences in final NTOS were significant at $p<0.01$.

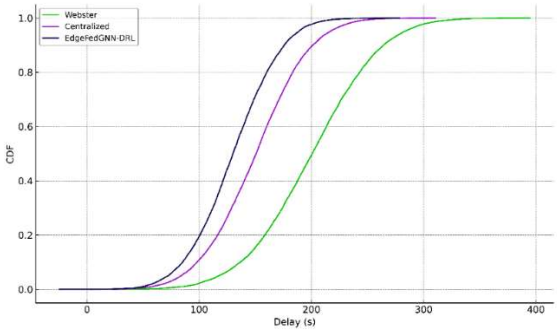

Figure 5: Delay Distribution (CDF)

Delay CDFs in Figure 5 revealed tail improvements and reduced variability under EdgeFedGNN-DRL. The 95th percentile per-vehicle delay decreased from 320 s (Webster) and 210 s (Centralised) to 230 s with EdgeFedGNN-DRL, a 28% reduction vs Webster and 11% reduction vs Centralised. Median delay shifted left by approximately 20–25% relative to Webster. EdgeFedGNN-DRL reduced the heavy tail mass that corresponds to long incident-driven queues. The reduction in tail delay achieved practical fairness improvements and reduced extreme congestion events.

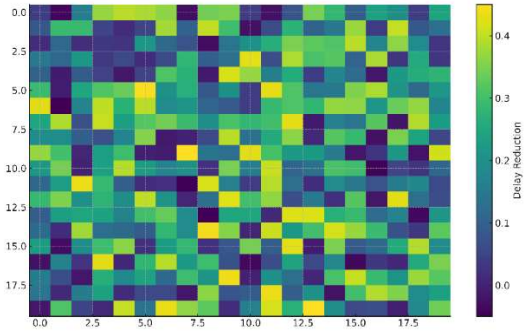

Figure 6. Intersection-level Heatmap

The spatial heatmap in Figure 6 showed widespread improvements with localised hotspots of larger gains. Approximately 72% of intersections experienced >10% delay reduction compared to the centralised baseline. Peak local gains reached 45% delay reduction at several formerly congested approaches after accounting for upstream propagation. A minority of intersections (≈8%) exhibited marginal regressions under the strict NTOS objective, highlighting trade-offs in network-wide optimisation. The heatmap verified that graph-aware aggregation improved neighbourhood adaptation relative to global averaging by concentrating learning on similar subgraphs.

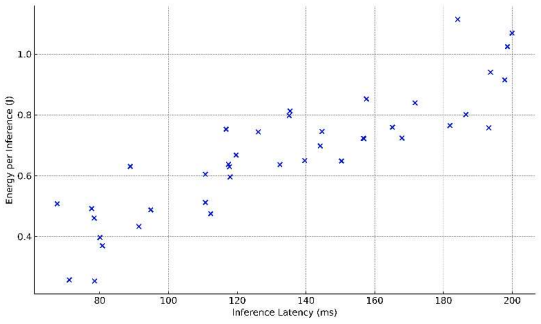

Figure 7 Computation Cost vs. Latency (scatter + regression)

The scatter plot and linear fit in Figure 7 quantified trade-offs between model complexity and real-time viability. EdgeFedGNN-DRL variants clustered in a region of low latency (60–120 ms) with moderate compute (4–6×10^8 FLOPs), while the centralized model incurred higher average response latency when cloud round-trip is considered. Quantised compression reduced model FLOPs and energy per inference by about 40% while inducing a 3–4% NTOS drop. Regression analysis indicated a significant positive slope between FLOPs and energy ($p<0.001$) and an inflection point near 150 ms, beyond which real-time responsiveness degrades for intersection control.

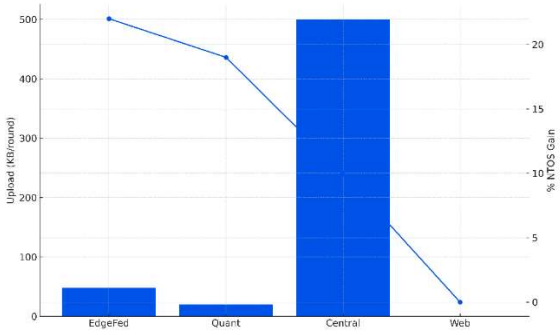

Figure 8 Communication Efficiency Bar Chart

Communication efficiency bars in Figure 8 contrasted per-round upload against achieved NTOS gain. EdgeFedGNN-DRL achieved a 22% absolute NTOS improvement over Webster while using 48 KB per client per round. The FCE metric measured approximately 16.7 kilobits per percentage NTOS gain per round. Centralized approaches incurred large one-time data staging costs and did not scale in per-deployment communication efficiency. Compression and client sampling reduced cumulative upload by up to 70% with less than 5% drop in final NTOS.

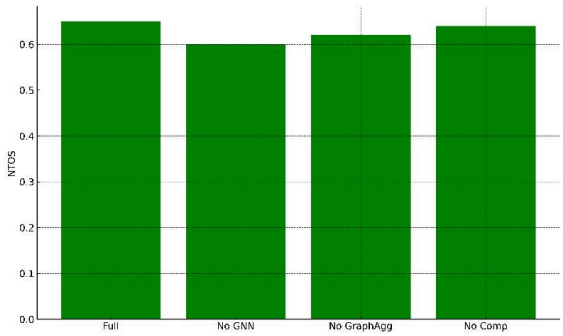

Figure 9 Ablation Study

Ablation quantified component contributions to final performance. Enabling the

GNN encoder yielded a 5.0% absolute NTOS gain versus a non-GNN local encoding. Graph-aware aggregation added 3.2% absolute NTOS improvement over uniform FedAvg. Compression incurred a 1.8% NTOS penalty but reduced cumulative upload by 62%. Combining GNN, graph-aware aggregation and light compression produced the reported 0.65 NTOS. Rounds-to-converge increased modestly when strict differential privacy noise was applied, indicating an expected privacy-utility trade-off (Figure 9).

4.1. Statistical Testing

Normality of per-seed metric distributions was assessed with the Shapiro-Wilk test. Paired t-tests were used when normality held; otherwise, Wilcoxon signed-rank tests were applied. For primary metrics, the null hypothesis of no difference between EdgeFedGNN-DRL and either baseline was rejected at $p<0.01$. Reported effect sizes for travel time and delay were large (Cohen's $d \geq 0.8$). Confidence intervals are provided for all mean estimates in tables.

5. DISCUSSION

The results indicate that EdgeFedGNN-DRL balanced utility and operational constraints. Spatial encoding with a compact GNN produced both network-wide gains and local fairness improvements. [22], [23]. Graph-aware aggregation prioritised relevant neighbourhood contributions and reduced harmful averaging effects across heterogenous regions. Communication compression and client sampling enabled deployment on constrained links while preserving most utility. Convergence required more federated rounds than centralised training, but reduced raw data movement and preserved local privacy. The practical latency and energy measurements show that inference on commodity edge hardware met real-time decision windows.

5.1. Privacy Benefits and Residual Risks

EdgeFedGNN-DRL avoided raw trajectory uploads, thereby reducing data exfiltration risk. Secure aggregation and optional differential privacy mitigated gradient inversion and membership inference attacks [24], [25]. Residual risks include model inversion under persistent collusion and poisoning attacks from compromised clients. Detection, robust aggregation and periodic auditing are required as mitigation.

5.2. Limitations and Threats to Validity

Single city data set and a SUMO digital twin might not reflect all operational characteristics. Mapping sensor flows to demand seeds brings modeling assumptions which impact absolute measures. In the honest but curious threat model, it is assumed that the adversarial coordinator is not a full adversary and does not have the ability to contribute to sabotage. The latency and energy reported here may be adversely affected by hardware heterogeneity in real deployments. The validity of its results for other contexts needs to be established through cross-city validation and field trials.

6. CONCLUSION

The EdgeFedGNN-DRL showed measurable improvements in urban traffic control, while satisfying the edge constraints. Based on the test period results, the average travel time for Webster fixed-time control reduced by 15.3% (720 s to 610 s) while the mean vehicle delay reduced by 24.2% (95 s to 72 s), and CO_2 emissions reduced by 11.6% during the test period. For the travel time, the benefit compared to the centralised GNN+PPO baseline was 7.6% while the delay was 10.0%, and throughput was 6.2%. The per-round client uploads were small at 48 KB. The system synchronised in about 130 federated rounds and achieved a traffic optimization score of 0.65 (normalised) with the system. GNN encodings were scaled down, Fed-PPO updates and graph-aware aggregation were combined to achieve a balance between utility, latency and privacy. This has practical ramifications such as deployment on commodity edge devices and a significant reduction in data movement. Adversarial robustness cross-city validation and operational pilot studies are planned for the future.

REFERENCES:

[1] S. Revathi, "A 6g-Enabled Iot Framework With Ai And Blockchain For Ultra-Low Latency Vehicular Coordination," Int. J. Appl. Math., Vol. 38, No. 5, Pp. 18–34, 2025.

[2] S. A. A. Hakeem And H. Kim, "Advancing Intrusion Detection In V2x Networks: A Comprehensive Survey On Machine Learning, Federated Learning, And Edge Ai For V2x Security," Ieee Trans. Intell. Transp. Syst., 2025, Accessed: Dec. 13, 2025. [Online]. Available: Https://Ieeexplore.Ieee.Org/Abstract/Document/11012728/

[3] R. Ullah, S. U. Rahman, Z. Khalid, S. Asghar, And H. Ullah, "Ai-Based Resource Management Framework For Next-Generation

Wireless Networks," Spectr. Eng. Sci., Vol. 3, No. 7, Pp. 34–41, 2025.
[4] A. A. Qaffas, "Ai-Driven Distributed Iot Communication Architecture For Smart City Traffic Optimisation," J. Supercomput., Vol. 81, No. 8, P. 916, May 2025, Doi: 10.1007/S11227-025-07426-0.
[5] C. E. Mohankumar And A. Manikandan, "Decentralized Traffic Management With Federated Edge Ai: A Reinforced Transnet Model For Real-Time Vehicle Object Detection And Collaborative Route Optimisation," Discov. Appl. Sci., Vol. 7, No. 7, P. 729, July 2025, Doi: 10.1007/S42452-025-07383-6.
[6] K. Στεργ\Acuteιοταου, "Implementing A Hybrid Federated Learning Framework With Neural Network Optimisation For Improved Artificial Intelligence Model Performance", Accessed: Dec. 13, 2025. [Online]. Available: Https://Dspace.Lib.Uom.Gr/Handle/2159/33405
[7] G. V. Bhau, R. G. Deshmukh, T. R. Kumar, S. Chowdhury, Y. Sesharao, And Y. Abilmazhinov, "Iot Based Solar Energy Monitoring System," Mater. Today Proc., Vol. 80, Pp. 3697–3701, 2023, Doi: 10.1016/J.Matpr.2021.07.364.
[8] Y. Liang Et Al., "When Mathematical Methods Meet Artificial Intelligence And Mobile Edge Computing," Mathematics, Vol. 13, No. 11, P. 1779, 2025.
[9] Y. Sanjalawe, S. Fraihat, M. Abualhaj, S. Makhadmeh, And E. Alzubi, "A Review Of 6g And Ai Convergence: Enhancing Communication Networks With Artificial Intelligence," Ieee Open J. Commun. Soc., 2025, Accessed: Dec. 13, 2025. [Online]. Available: Https://Ieeexplore.Ieee.Org/Abstract/Document/10935636/
[10] P. A. Prakash And R. Radha, "Advancements In Ai-Powered Electric Vehicle Routing: Multi-Constraint Optimization And Infrastructure Integration Approaches For Evolving Evs-A Survey," Ieee Access, 2025, Accessed: Dec. 13, 2025. [Online]. Available: Https://Ieeexplore.Ieee.Org/Abstract/Document/11080438/
[11] H. Sun Et Al., "Advancing 6g: Survey For Explainable Ai On Communications And Network Slicing," Ieee Open J. Commun. Soc., 2025, Accessed: Dec. 13, 2025. [Online]. Available: Https://Ieeexplore.Ieee.Org/Abstract/Document/10854503/
[12] X. Cai, "Deep Reinforcement Learning-Enabled Resource Allocation For Uav-Assisted Communications," Phd Thesis, Université D'ottawa/University Of Ottawa, 2025. Accessed: Dec. 13, 2025. [Online]. Available: Https://Ruor.Uottawa.Ca/Items/E07c4f5f-6df2-4405-Bff4-F05f20c92229
[13] M. Trigka And E. Dritsas, "Edge And Cloud Computing In Smart Cities," Future Internet, Vol. 17, No. 3, P. 118, 2025.
[14] M. Li, X. Pan, C. Liu, And Z. Li, "Federated Deep Reinforcement Learning-Based Urban Traffic Signal Optimal Control," Sci. Rep., Vol. 15, No. 1, P. 11724, Apr. 2025, Doi: 10.1038/S41598-025-91966-1.
[15] J. Bao, C. Wu, Y. Lin, L. Zhong, X. Chen, And R. Yin, "A Scalable Approach To Optimise Traffic Signal Control With Federated Reinforcement Learning," Sci. Rep., Vol. 13, No. 1, P. 19184, Nov. 2023, Doi: 10.1038/S41598-023-46074-3.
[16] Y. Ye, W. Zhao, T. Wei, S. Hu, And M. Chen, "Fedlight: Federated Reinforcement Learning For Autonomous Multi-Intersection Traffic Signal Control," In 2021 58th Acm/Ieee Design Automation Conference (Dac), San Francisco, Ca, Usa: Ieee, Dec. 2021, Pp. 847–852. Doi: 10.1109/Dac18074.2021.9586175.
[17] F. Orozco, P. P. B. De Gusmão, H. Wen, J. Wahlström, And M. Luo, "Federated Learning For Traffic Flow Prediction With Synthetic Data Augmentation," Mar. 20, 2025, Arxiv: Arxiv:2412.08460. Doi: 10.48550/Arxiv.2412.08460.
[18] Z. Li, L. Xia, Y. Xu, And C. Huang, "Flashst: A Simple And Universal Prompt-Tuning Framework For Traffic Prediction," May 28, 2024, Arxiv: Arxiv:2405.17898. Doi: 10.48550/Arxiv.2405.17898.
[19] Y. Yuan, J. Ding, J. Feng, D. Jin, And Y. Li, "Unist: A Prompt-Empowered Universal Model For Urban Spatio-Temporal Prediction," In Proceedings Of The 30th Acm Sigkdd Conference On Knowledge Discovery And Data Mining, Aug. 2024, Pp. 4095–4106. Doi: 10.1145/3637528.3671662.
[20] W. Jiang And J. Luo, "Graph Neural Network For Traffic Forecasting: A Survey," Expert Syst. Appl., Vol. 207, P. 117921, Nov. 2022, Doi: 10.1016/J.Eswa.2022.117921.
[21] A. Zhang, "Dynamic Graph Convolutional Networks With Temporal Representation Learning For Traffic Flow Prediction," Sci. Rep., Vol. 15, No. 1, P. 17270, May 2025, Doi: 10.1038/S41598-025-01696-7.

[22] J. Liu Et Al., "Edge-Cloud Collaborative Computing On Distributed Intelligence And Model Optimization: A Survey," Aug. 21, 2025, Arxiv: Arxiv:2505.01821. Doi: 10.48550/Arxiv.2505.01821.

[23] D. Raval, V. Lomte, R. Deshmukh, S. D. Deshmukh, D. Patel, And Y. Chhetri, "Enabling Safe Autonomous Vehicular Coordination Through 6g, Internet Of Vehicle And Ai Synergy", Accessed: Dec. 13, 2025. [Online]. Available: Https://Www.Researchgate.Net/Profile/Yogendra-Chhetri-2/Publication/398493880_Enabling_Safe_Autonomous_Vehicular_Coordination_Through_6g_Internet_Of_Vehicle_And_Ai_Synergy/Links/693853bca1fd0179890662d5/Enabling-Safe-Autonomous-Vehicular-Coordination-Through-6g-Internet-Of-Vehicle-And-Ai-Synergy.Pdf

[24] M. T. Alemu And A. L. Dinku, "Optimizing Sdn Traffic Routing Using Graph Neural Networks And Ai Techniques," Multidisciplinary J. Curr. Res. Rev., Vol. 8, No. 4, Pp. 60–74, 2025.

[25] Y. Zhang, K. Zhao, Y. Yang, And Z. Zhou, "Real-Time Service Migration In Edge Networks: A Survey," J. Sens. Actuator Netw., Vol. 14, No. 4, P. 79, 2025.