Computers, Materials & Continua

Tech Science Press

Doi:10.32604/cmc.2026.075250

ARTICLE

Mobility-Aware Federated Learning for Energy and Threat Optimization in Intelligent Transportation Systems

Hamad Ali Abosaq[1], Jarallah Alqahtani[1,*], Fahad Masood[2], Alanoud Al Mazroa[3], Muhammad Asad Khan[4] and Akm Bahalul Haque[5]

[1]Computer Science Department, College of Computer Science and Information Systems, Najran University, Najran, 61441, Saudi Arabia
[2]Department of Computer Science, CECOS University of IT and Emerging Sciences, Peshawar, 25000, Pakistan
[3]Department of Information Systems, College of Computer and Information Sciences, Princess Nourah bint Abdulrahman University, Riyadh, 11671, Saudi Arabia
[4]Department of Telecommunication, Hazara University, Mansehra, 21120, Pakistan
[5]The Faculty of Science and Engineering, Abo Akademi University, Turku, 20520, Finland

*Corresponding Author: Jarallah Alqahtani. Email: jaalqahtani@nu.edu.sa

Received: 28 October 2025; Accepted: 26 December 2025; Published: 12 March 2026

ABSTRACT: The technological advancement of the vehicular Internet of Things (IoT) has revolutionized Intelligent Transportation Systems (ITS) into next-generation ITS. The connectivity of IoT nodes enables improved data availability and facilitates automatic control in the ITS environment. The exponential increase in IoT nodes has significantly increased the demand for an energy-efficient, mobility-aware, and secure system for distributed intelligence. This article presents a mobility-aware Deep Reinforcement Learning based Federated Learning (DRL-FL) approach to design an energy-efficient and threat-resilient ITS. In this approach, a Policy Proximal Optimization (PPO)-based DRL agent is first employed for adaptive client selection. Second, an autoencoder-based anomaly detection module is considered for malicious node detection. Results reveal that the proposed framework achieved an 8% higher accuracy increase, and 15% lower energy consumption. The model also demonstrates greater resilience under adversarial conditions compared to the state of the art in federated learning. The adaptability of the proposed approach makes it a compelling choice for next-generation vehicular networks.

KEYWORDS: Intelligent Transportation Systems (ITS); energy efficiency; mobility management; federated learning; deep reinforcement learning

1 Introduction

The vehicular Internet of Things (IoT) has seen massive growth across an extensive range of features in Intelligent Transport Systems (ITSs). The incorporation of IoT into ITS has transformed it into next-generation ITS with higher connectivity and intelligence [1,2]. This transformation can be leveraged to enhance traffic management, route optimization, and accident prevention. Federated Learning (FL) has emerged as an innovative learning paradigm to address the challenges of high communication cost, privacy risks, and latency. The cooperative model training is enabled on edge nodes to improve data privacy and reduce network congestion [3].

FL deployment in heterogeneous environments is a challenging task regardless of its potential [4]. The model's convergence and accuracy can be significantly degraded by high vehicular mobility and

frequent disconnections. The vehicular nodes continuously move in ITSs, which affects learning stability. Studies have shown that learning performance can be improved with reliable connections and timely node engagements [5]. The direct influence of mobility patterns has been ignored in existing FL frameworks, leading to aggregation delays and suboptimal accuracy in dense scenarios [6].

Energy efficiency is another challenging concern in FL-oriented IoT environments [7]. In vehicular networks, IoT devices are deployed and powered by batteries, which limits their power reserves. A significant amount of computational power is consumed during communication and model training. It not only shortens the device's lifetime but also disrupts the network connectivity. The proposed FL algorithms focus on energy consumption to optimize communication cost. The impact of node mobility on energy consumption in the ITS-FL environment remains an open challenge [8,9].

Privacy and trust management are other critical issues in addition to the energy and mobility constraints [10–12]. FL systems have a distributed architecture, which makes them vulnerable to data manipulation and to attacks on attack detection. Uncompromised nodes in vehicular networks can introduce malicious information that disrupts the global model. Various detection methods and trust-based schemes have been proposed to address these risks, but most techniques rely on centralized verification [13,14]. There is a need to develop an intelligent mechanism to detect mobile characteristics for secure and efficient FL procedures.

This research proposes an enhanced IoT-based DRL-FL framework for ITS. The main contribution of the paper includes a DRL agent for optimizing accuracy, energy usage, and trust level. An analytical mobility model is considered to estimate the client stability and minimize data loss. It also includes continuous monitoring of the node's energy and local parameter adjustment. Malicious node detection has been performed using an autoencoder mechanism for secure model aggregation.

Mobility-aware associate learning frameworks (such as MOB-FL, ESAFL, and MDFL) have focused on synchronous or semi-asynchronous aggregation under vehicle mobility. This framework optimizes energy efficiency, mobility stability, and robustness against malicious nodes in a unified architecture. A node selection mechanism and trust-weighted aggregation based on deep reinforcement learning (DRL) have been introduced. This integration includes a policy proximal optimization (PPO) based DRL agent for adaptive client selection, an autoencoder-driven anomaly detection, and a trust-weighted aggregation scheme for predicted connectivity, trust score, threat mitigation, and reliable global model updates. The novelty of the proposed model lies in integrating DRL and FL functionality in a mobile environment. This framework will not only improve the efficiency of intelligent traffic management but also enhance the smart vehicular network.

The remainder of this paper is organized as follows. Section 2 reviews related studies on federated learning in ITS, energy optimization, and security enhancement. Section 3 presents the system model and detailed methods of the proposed framework. Section 4 discusses the experimental setup and evaluation, followed by a conclusion and future work in Section 5.

2 Literature Review

Federated learning (FL) for vehicular and intelligent connected vehicle (ICV) environments is gaining increasing attention because it enables collaborative model training without sharing raw data within the vehicle. Much recent literature focuses on addressing the unique challenges posed by vehicle mobility—short contact times, frequent handovers, and intermittent connectivity—and on improving FL convergence and robustness under these dynamic conditions. Below, we summarize and critically assess recent representative

contributions addressing mobility, synchronization/aggregation strategies, decentralized learning, client selection, and self-supervised pre-training in vehicular FL settings.

Xie et al. proposed MOB-FL, a mobility-aware federated learning framework that explicitly optimizes the duration of each training round and the number of local iterations to maximize resource utilization in short-term wireless connections [13]. MOB optimizations to effectively utilize available contact time in each vehicle, thereby reducing wasted computation/communication opportunities and accelerating convergence. This approach has been validated on beam selection and trajectory prediction tasks. The results show that adapting round duration and workload per client to mobility substantially improves FL convergence in high-mobility scenarios. The strength of MOB-FL lies in operationally calculating contact times and mapping them to FL parameters; however, this approach primarily targets convergence speed through scheduling and does not address complementary issues such as client device energy sustainability or adversarial resilience in updates.

Jin et al. recognized the complementary issues arising from mobility and heterogeneity and proposed ESAFL, a semi-asynchronous FL scheme explicitly designed for vehicular networks [14]. ESAFL mitigates the straggler effect by grouping connected vehicles into layers based on arrival order and employing an Age-of-Information (AoI) aggregation strategy to balance delayed contributions. This semi-asynchronous design mitigates the negative impact of late or missing client uploads and empirically improves accuracy and convergence on standard image datasets. ESAFL is renowned for its practical focus on asynchronous aggregation and its intelligent use of AoI to weight delayed updates. However, ESAFL assumes a certain level of layering and coordination in RSUs. At the same time, it addresses availability and delay, but it does not explicitly optimize energy usage per client or incorporate vehicle reliability into aggregation weights.

Recent research has focused on decentralized or leader-based formulations that are more resilient to infrastructure failures and scale better for dense vehicular deployments. The Mobility-Aware Decentralized Learning (MDFL) framework formulates local iteration and leader election as joint optimization and solves them using a multi-agent RL (MAPPO) approach under the Dec-POMDP formulation [15]. MDFL aims to improve training efficiency in vehicular networks by enabling neighboring vehicles to collaborate in a decentralized manner and selecting a leader that optimally coordinates local aggregation. The multi-agent perspective is powerful for fully distributed settings and can better leverage local vehicle clusters; however, MDFL primarily focuses on iteration/leader selection and on decentralized coordination, raising questions about continuous energy monitoring, client safety thresholds, and security against malicious updates in realistic ITS environments.

Client selection has also been studied from a more classical optimization perspective. Chang et al. proposed a mobility-aware vehicle selection strategy that jointly considers geographic location, speed, and data quality to select vehicles capable of completing training and upload within the available time frame [16]. This dynamic selection approach demonstrated that combining mobility metrics with data utility and resource capability yields faster convergence and greater accuracy compared to naive selection. The strength of this line of work lies in its focus on selecting the highest-value participants under time constraints; its limitation is that selection heuristics are generally reactive and do not learn from long-term results (e.g., they do not utilize DRL to optimize the long-term trade-off between energy, confidence, and accuracy).

The integration of self-supervised learning (SSL) with FL for vehicle perception tasks is another complementary avenue. A mobility-adaptive federated self-supervised learning scheme has been presented using FLSimCo [17]. This scheme uses image blur levels as a quality metric for pre-training aggregation. It also addresses the practical problem that high vehicle speeds can produce blurred images that, if naively aggregated, would corrupt the overall model. FLSimCo improves the stability and convergence of SSL in the vehicular context by weighting or filtering updates based on modality-specific quality. This

research underscores the importance of adapting data quality for FL aggregation. Still, it focuses on the pre-training/SSL task rather than broader issues such as scheduleability, energy consumption, or security.

Energy-efficient and privacy-preserving federated learning (FE) are recent advances that have significantly impacted the development of ITS and smart vehicle networks [18]. A hybrid FE-based model for energy-efficient IoT systems demonstrates energy-aware aggregation and lightweight local optimization to reduce communication overhead. Improved model accuracy and device sustainability, reaching accuracy above 93% with lower communication latency, is also a further achievement. An Explainable Federated Learning (XFL) framework has been introduced for vehicular energy control in smart cities that combines hierarchical FE with explainable AI to improve transparency and achieve superior predictive accuracy (R^2 up to 99.83%) [19]. It has been complemented by a Weighted Explainable FL (WEFL-XAI) approach that adaptively assigns weight to client updates based on data relevance and local model performance to improve privacy and scalability.

A collaborative air-to-ground ground transportation AF system was proposed that integrates Bayesian prediction mechanisms and incentives [20]. The objective was to manage large-scale, energy-efficient, privacy-protecting intelligent traffic networks. The growing convergence of energy, privacy, and urban mobility has gained valuable space in these studies [21–24]. Adaptive decision-making in critical mobility scenarios has also been lacking, underscoring the need for an improved, energy-aware, and threat-resilient urban mobility framework for next-generation vehicular networks. Table 1 compares various federated learning approaches in ITS environments.

Table 1: Comparison of various federated learning approaches in ITS environments

Approach	Key features	Objectives/Metrics improved	Limitations
MOB-FL [13]	Mobility-aware scheduling, adaptive local epochs, contact-time prediction	Reduce wasted training time, improve FL convergence under dynamic mobility	Ignores malicious updates, energy constraints, client safety guarantees
ESAFL [14]	Semi-asynchronous aggregation, Age-of-Information-based update weighting	Mitigate stragglers, reduce synchronization delay, improve accuracy	Requires layered grouping; does not optimize energy or trust parameters
MDFL [15]	Decentralized learning, leader election, multi-agent RL coordination	Improve scalability and decentralized robustness in dense networks	Focuses on routing/coordination, limited security, complexity increases with node density
WEFL/XFL [19]	Weighted explainable update aggregation, privacy-preserving training	Enhance data relevance, transparency, and privacy without raw data sharing	Does not address mobility or resource availability in ITS
Energy-Aware FL [18]	Lightweight optimization, energy consumption, and communication reduction	Improve device lifetime and reduce overhead for IoT/vehicular devices	Does not incorporate threat-resilience or mobility-aware scheduling
FLSimCo [17]	Self-supervised federated pre-training, blur-aware update filtering	Improve SS pre-training quality and convergence for vehicular cameras	Focus limited to SSL; lacks client selection and energy/trust modeling

3 Methodology

This research aims to develop a mobility-based intelligent Deep Reinforcement Learning, Federated Learning (DRL-FL) framework in an IoT-assisted intelligent transportation system (ITS), shown in Fig. 1. The data has been collected for the IoT nodes, including mobility and energy constraints derived from the Udacity Self-Driving Car Dataset. The global objective and local updates are calculated via federated optimization, followed by the mobility prediction and stability score. The Energy Modeling is performed for energy consumption per client, calculated as the sum of computational and communication energy. Anomaly detection and trust updates have been done for device-level anomaly using a local autoencoder. The combined energy-trust-mobility selection has been posed as a round-by-round constrained optimization problem. This framework allows the DRL-FL components to explore a broader range of possible schemes. It also develops the framework's ability to enhance learning performance, optimizing ITS solutions in dynamic situations. The key parameters used in the proposed framework have been shown in Table 2.

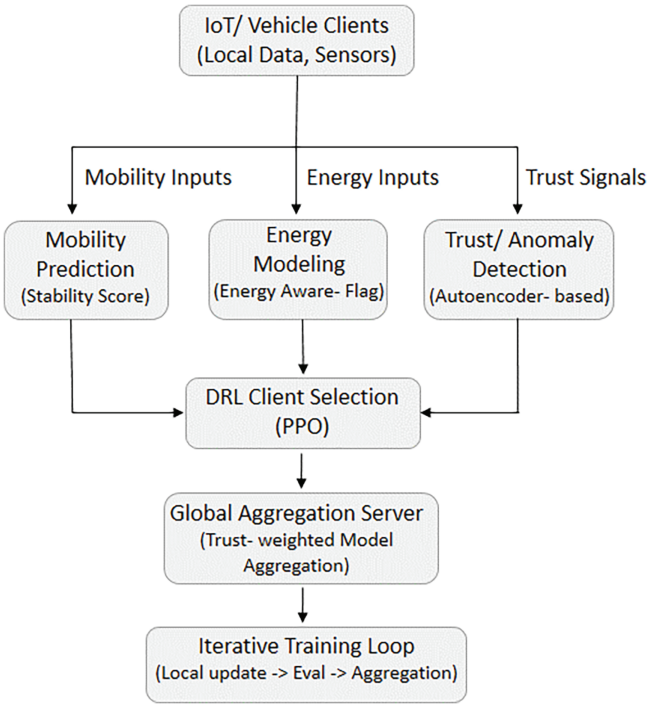

Figure 1: DRL-FL architecture

Table 2: Key parameters and their assigned values

Symbol	Description	Value/Range
w_t	Global FL model updated at round t	Updated every round
Δw_i	Local model update of client i	Depends on model dimensions
E_i^{rem}	Remaining energy level of client i	50–100 J
$E_{\min}$	Minimum energy threshold for participation	30 J
T_i	Trust score of client i	$[0,1]$
$T_{\min}$	Minimum trust threshold for secure participation	0.4
$\mathcal{A}_i$	Anomaly score via autoencoder reconstruction	$[0,1]$

(Continued)

Table 2 (continued)

Symbol	Description	Value/Range
π_θ	PPO-based DRL policy	2-layer NN (64 units)
S_t	DRL state: energy, trust, mobility, update quality	Vector of all features
A_t	DRL actions (client selection, epochs, power)	Selection: 0/1; epochs: 1–10
R_t	Reward: accuracy-energy-trust weighted	$\alpha = 0.5,\ \beta = 0.3,\ \gamma = 0.2$
λ_T	Trust update coefficient	0.05–0.1
τ_{sim}	Similarity threshold for detecting anomalies	0.7
θ_{anom}	Autoencoder anomaly threshold	0.15
I	PPO policy update interval	Every 10 rounds
ρ_i	Local computation fraction for client i	0.5–1

3.1 Global Objective and Local Updates

The set of N clients is denoted by $\mathcal{N} = \{1, \ldots, N\}$. Client i has a local dataset D_i of size $|D_i|$, and the total size of the dataset is $|D| = \sum_{i=1}^{N} |D_i|$. The global federation's objective is the weighted average of the local objectives.

$$\min_w F(w) \quad \text{where} \quad F(w) = \sum_{i=1}^{N} \frac{|D_i|}{|D|} F_i(w). \tag{1}$$

The Eq. (1) defines the global loss to be minimized by federated optimization, where $F_i(w)$ is the local loss at client i. The local model update is performed at client i during e_i local epochs with a learning rate η

$$w_i^{t+1} = w^t - \eta \sum_{k=1}^{e_i} \nabla F_i(w_i^{t,k}), \tag{2}$$

where $w_i^{t,k}$ denotes the local model after k local mini-batch updates during round t. The Eq. (2) represents standard gradient-based local training. The edge/cloud aggregates client updates using a confidence-weighted average

$$w^{t+1} = \sum_{i \in \mathcal{S}_t} \frac{\alpha_i T_i^t}{\sum_{j \in \mathcal{S}_t} \alpha_j T_j^t} w_i^{t+1}, \tag{3}$$

where $\mathcal{S}_t \subseteq \mathcal{N}$ is the set of selected clients at round t, $\alpha_i = |D_i|/|D|$ is the data size weight, and $T_i^t \in [0, 1]$ is the trust score for client i at round t.

3.2 Mobility Prediction and Stability Score

Each client i has a mobility state vector containing position and kinematic features

$$M_i(t) = [p_i(t), v_i(t), a_i(t), \theta_i(t)], \tag{4}$$

where p is position, v is velocity, a is acceleration, and θ is direction. A sequence model is used to predict the next step position or waiting time. Suppose the predictor is $f_{\text{mob}}(\cdot)$, then the predicted waiting time within the current RSU coverage is,

$$\hat{\tau}_i(t) = f_{\text{mob}}\big(M_i(t - \Delta t), \ldots, M_i(t)\big). \tag{5}$$

The dwell-time predictor estimates how long client i will remain connected, which is used for scheduling feasibility. Determine the connection stability score $S_i(t) \in [0,1]$ that decreases with lower estimated distance to RSU or dwell time.

$$S_i(t) = \exp\Big(-\frac{\hat{d}_i(t)}{d_{\max}}\Big) = \exp\Big(-\frac{v_i(t)\cdot\hat{\tau}_i(t)}{d_{\max}}\Big), \tag{6}$$

where $\hat{d}_i(t)$ is the predicted displacement during the residence time interval and $d_{\max}$ is a normalization constant. A small $S_i(t)$ indicates a high risk of disconnection.

3.3 Energy Modeling and Feasibility Constraints

The energy consumption per client is modeled as the sum of the computational and communication energy required to participate in one round. The computational (local training) energy for client i executing e_i epochs is,

$$E_i^{\text{comp}}(e_i) = \eta_i \cdot \text{FLOPs}_{\text{per\_epoch}} \cdot e_i, \tag{7}$$

where η_i (Joules per FLOP) is the device-specific energy coefficient and FLOPs_per_epoch depends on the model and batch size. The communication (upload) energy is,

$$E_i^{\text{tx}}(\Delta_i, R_i) = P_i^{\text{tx}} \cdot t_i^{\text{tx}} \quad \text{with} \quad t_i^{\text{tx}} = \frac{\Delta_i}{R_i}, \tag{8}$$

where P_i^{tx} is the effective transmit power (W), Δ_i is the compressed update size (bits), and R_i is the uplink rate (bps). The total expected energy for client i in round t,

$$E_i(t) = E_i^{\text{comp}}(e_i) + E_i^{\text{tx}}(\Delta_i, R_i). \tag{9}$$

The feasibility of computation time and transmission time must comply with the estimated dwell time:

$$T_i^{\text{comp}}(e_i) + T_i^{\text{tx}}(\Delta_i, R_i) \le \hat{\tau}_i(t), \tag{10}$$

where $T_i^{\text{comp}}(e_i) \approx \dfrac{\text{FLOPs}_{\text{per\_epoch}} \cdot e_i}{\text{FLOPS}_i}$ and T_i^{tx} is from (8). The (10) constraint ensures that the client is most likely to complete local training and upload before disconnection. Remaining battery update after participation:

$$E_i^{\text{rem}}(t+1) = E_i^{\text{rem}}(t) - E_i(t) + H_i(t), \tag{11}$$

where $H_i(t)$ models the probability of energy harvesting between rounds. The security constraints are enforced

$$E_i^{\text{rem}}(t+1) \ge E_{\min}, \tag{12}$$

with $E_{\min}$ a predetermined safety threshold (e.g., 10%) battery capacity.

3.4 Anomaly Detection and Trust Update Equations

Device-level anomaly detection uses a local autoencoder to reconstruct the telemetry vector $x_i(t)$ (energy, loss, gradient norm, and telemetry). The mean-squared reconstruction error is calculated as,

$$\mathcal{A}_i(t) = \frac{1}{d}\sum_{j=1}^{d}\left(x_{i,j}(t) - \hat{x}_{i,j}(t)\right)^2 \tag{13}$$

and the normalized anomaly score for trust weighting is computed as,

$$\mathcal{A}_i^{\text{norm}}(t) = \frac{\mathcal{A}_i(t) - \mu_{\mathcal{A}}(t)}{\sigma_{\mathcal{A}}(t)} \tag{14}$$

where: $x_i(t)$ is the local model update vector of client i at round t, $\hat{x}_i(t)$ is the reconstructed vector from the autoencoder, d is the dimension of the update vector, $\mu_{\mathcal{A}}(t)$ and $\sigma_{\mathcal{A}}(t)$ are the mean and std of reconstruction errors across all clients. The Cosine similarity can be given as,

$$\text{sim}_i = \frac{\langle \Delta w_i, \tilde{\Delta w} \rangle}{\|\Delta w_i\| \, \|\tilde{\Delta w}\|}. \tag{15}$$

If $\text{sim}_i < \tau_{\text{sim}}$, the update is suspicious. The confidence score $T_i(t) \in [0,1]$ is updated as a convex combination that penalizes anomalies as,

$$T_i(t+1) = \lambda_T \, T_i(t) + (1-\lambda_T)\left(1 - \frac{\min(\mathcal{A}_i(t), \mathcal{A}_{\max})}{\mathcal{A}_{\max}}\right) \cdot \mathbb{I}[\text{sim}_i \geq \tau_{\text{sim}}], \tag{16}$$

where $\lambda_T \in [0,1]$ is the forgetting factor, $\mathcal{A}_{\max}$ normalizes the anomaly score, and the indicator confirms that low-similarity updates reduce trust. Trust-weighted aggregation uses $T_i(t)$ as in (3).

3.5 Threat Model

The proposed framework used model poisoning and backdoor injection integrated with the trust evaluation mechanism to identify and suppress malicious contributions. The local gradients are manipulated using a subset of adversarial clients before transmission to the RSU. The poisoning is calculated as,

$$\tilde{\mathbf{g}}_i = \mathbf{g}_i + \alpha \, \boldsymbol{\delta}, \tag{17}$$

where $\mathbf{g}_i$ is the benign local gradient of client i, $\boldsymbol{\delta}$ is a randomly sampled perturbation vector (Gaussian noise), and α controls the severity of the attack. A target backdoor is also injected by the malicious clients, including local training labels. Let $\mathbf{x}$ be an input sample and y its correct label, the poisoned labels are generated as

$$y' = \begin{cases} t, & \text{if } \mathbf{x} \in \mathcal{D}_{\text{trigger}}, \\ y, & \text{otherwise}, \end{cases} \tag{18}$$

where t is a predefined target class and $\mathcal{D}_{\text{trigger}}$ denotes samples containing the backdoor trigger. The poisoned updates are detected using gradient direction similarity, energy sufficiency, and mobility stability. Each client's gradient and the median gradient direction compute the cosine similarity as,

$$\mathcal{S}_i = \cos\left(\angle\left(\tilde{\mathbf{g}}_i, \text{median}(\tilde{\mathbf{g}})\right)\right), \tag{19}$$

where lower values indicate potential poisoning. The final trust score is then updated as,

$$T_i^{(t+1)} = \lambda_T \mathcal{S}_i + (1 - \lambda_T) f(E_i, v_i), \tag{20}$$

where $f(E_i, v_i)$ incorporates energy availability E_i and mobility stability v_i of client i, and λ_T controls the weight of anomaly-based trust adjustment.

3.6 MDP Formulation and PPO-Based DRL Optimization

This scheduling problem is transformed into an MDP with states S_t, actions A_t, and rewards R_t. The state vector at round t is,

$$S_t = \left\{E_i^{\text{rem}}(t),\ T_i(t),\ S_i(t),\ R_i(t),\ \Delta F_i(t)\right\}_{i=1}^{N}, \tag{21}$$

where $R_i(t)$ is the uplink rate and $\Delta F_i(t)$ indicates the current local contribution. The action is client selection and per-client configuration.

$$A_t = \left\{a_i(t) = (s_i(t), e_i(t), \rho_i(t))\right\}_{i=1}^{N}, \tag{22}$$

with selection flags $s_i(t) \in \{0,1\}$, epoch $e_i(t) \in \mathbb{Z}^+$, and compression ratio $\rho_i(t) \in (0,1]$. The reward function is multi-objective.

$$R_t = \omega_{\text{acc}} \Delta\text{Acc}_t - \omega_E \overline{E}(t) - \omega_D \text{Drop}_t + \omega_T \overline{T}(t) + \omega_S \overline{S}(t), \tag{23}$$

where:

- ΔAcc_t = global validation accuracy gain at round t,
- $\overline{E}(t) = \frac{1}{|\mathcal{S}_t|} \sum_{i \in \mathcal{S}_t} E_i(t)$ is the average energy consumed,
- Drop_t = fraction of selected clients that fail to upload (disconnect or energy depletion),
- $\overline{T}(t)$ = average trust among selected clients,
- $\overline{S}(t)$ = average stability score among selected clients,
- ω_* is a scalar weight that balances the objectives.

The agent learns a policy $\pi_\theta(A|S)$ parameterized by θ (a neural network) to maximize the expected discounted return is

$$J(\theta) = \mathbb{E}_{\pi_\theta}\left[\sum_{t=0}^{\infty} \gamma^t R_t\right], \tag{24}$$

where $\gamma \in (0,1]$ is discount factors. Using Proximal Policy Optimization (PPO), the surrogate objective for policy updating is,

$$L^{\text{PPO}}(\theta) = \mathbb{E}_t\left[\min\left(r_t(\theta)\hat{A}_t,\ \text{clip}(r_t(\theta), 1-\varepsilon, 1+\varepsilon)\hat{A}_t\right)\right], \tag{25}$$

where $r_t(\theta) = \dfrac{\pi_\theta(A_t|S_t)}{\pi_{\theta_{\text{old}}}(A_t|S_t)}$, $\hat{A}_t$ is the advantage estimate, and ε is the clip parameter.

3.7 Optimization Problem (Mixed-Integer Form) and Relaxation

The combined energy-trust-mobility selection can be posed as a round-by-round constrained optimization problem expressed as,

$$\begin{aligned}
&\min_{s_i,e_i,\rho_i} \sum_{i=1}^{N} s_i\, E_i(e_i,\rho_i) \\
&\text{subject to} \sum_{i=1}^{N} s_i = K, \\
&\quad T_i^{\text{comp}}(e_i) + T_i^{\text{tx}}(\Delta_i(\rho_i), R_i) \le \hat{\tau}_i \quad \forall i : s_i = 1, \\
&\quad E_i^{\text{rem}} - E_i(e_i,\rho_i) \ge E_{\min} \quad \forall i : s_i = 1, \\
&\quad T_i \ge T_{\text{th}} \quad \forall i : s_i = 1, \\
&\quad s_i \in \{0,1\},\ e_i \in \mathbb{Z}^+,\ \rho_i \in (0,1]
\end{aligned} \tag{26}$$

where K is the number of selected target clients. Eq. (26) presents a mixed-integer nonlinear program (MINLP); in practice, the DRL policy approximates the solution online. For analysis, relax $s_i \in [0,1]$ to obtain a solvable convex surrogate for comparison. The total energy consumption is minimized by selecting devices, the number of local computation epochs, and the fraction of local data used. Each device contributes a minimum amount, ensuring devices are selected exactly with respect to latency and energy limits.

3.8 Metrics and Derived Quantities

The average energy per round is defined as,

$$\overline{E}_{\text{round}}(t) = \frac{1}{|\mathcal{S}_t|} \sum_{i \in \mathcal{S}_t} E_i(t). \tag{27}$$

The equation above calculates the mean energy consumption of all selected devices, given the typical energy cost per device. The energy efficiency (EE) as accuracy per unit energy can be calculated as,

$$\text{EE}(t) = \frac{\text{Acc}(t)}{\overline{E}_{\text{round}}(t)}, \tag{28}$$

where $\text{Acc}(t)$ is the global validation accuracy at round t. The model is considered convergent at round $T^\star$ if

$$|F(w^{T^\star}) - F(w^{T^\star-\Delta})| \le \varepsilon_F \tag{29}$$

This equation confirms the training convergence by comparing the global model's loss. Convergence is confirmed if the loss is below the threshold. The detection rate (DR) and false alarm rate (FAR) are calculated from the anomaly detector output as,

$$\text{DR} = \frac{\#\{\text{true positives}\}}{\#\{\text{malicious clients}\}}, \quad \text{FAR} = \frac{\#\{\text{false positives}\}}{\#\{\text{benign clients}\}}. \tag{30}$$

Algorithmic Framework Overview

The proposed framework consists of three integrated Algorithms 1–3 that collaboratively achieve adaptive, secure, and energy-efficient federated learning for intelligent transportation systems (ITS). Each algorithm corresponds to a specific operational stage in the federated learning (FL) cycle: client-side

participation, server-side aggregation, and policy optimization through deep reinforcement learning (DRL). A mobility-aware self-learning system has been developed that optimizes communication, computation, and trust in dynamic IoT-based vehicular environments. A single-tier centralized aggregation has been implemented in the proposed framework. Global model aggregation has been performed at each round using the Roadside Unit (RSU), which acts as the centralized FL coordinator. The local updates from the IoT nodes are sent to the RSU, which aggregates using an averaging rule. The nodes' participation, along with the energy allocation and trust-aware update, is optimized directly by keeping the communication pipeline simple. Table 3 shows the computational complexity of the proposed algorithms.

Algorithm 1: Client selection and local training

Require: global model w_0, DRL policy π_θ, trust T_i^0, thresholds $E_{\min}, T_{\min}, S_{\min}$, total rounds T

Initialize w_0, θ_0, T_i^0 for all clients; set replay buffer $\mathcal{B} \leftarrow \varnothing$
for each round $t = 0$ to $T-1$ **do**
  Telemetry: each client i sends $\{E_i^{rem}(t), M_i(t), R_i(t), |D_i|\}$ to edge server
  Mobility: predict dwell time $\hat{\tau}_i$ and stability S_i for all i
  DRL Selection: $A_t \leftarrow \pi_\theta(\{E_i^{rem}, T_i, S_i, R_i, |D_i|\})$
  for each client i **do**
    if $s_i = 1$ **and** $E_i^{rem} \geq E_{\min}$ **and** $S_i \geq S_{\min}$ **and** $T_i \geq T_{\min}$ **then**
      send (w_t, e_i, ρ_i) to i and start local training on D_i
      if energy use $> E_i^{rem} - E_{\min}$ **then** stop early and send partial Δw_i
      end if
      compute anomaly score $\mathcal{A}_i(t)$ via autoencoder
      label as *suspicious* if $\mathcal{A}_i(t) > \theta_{\text{anom}}$; else *normal*
      send $\{\Delta w_i, \mathcal{A}_i(t), E_i(t)\}$ to server
    end if
  end for
end for

Algorithm 2: Server aggregation and DRL update

Require: updates $\{\Delta w_i, \mathcal{A}_i(t), E_i(t)\}$, trust T_i, similarity threshold τ_{sim}, anomaly threshold θ_{anom}, update interval I

for each communication round t **do**
  Trust Update and Threat Handling
  for each received update Δw_i **do**
    compute similarity $\text{sim}_i \leftarrow \cos(\Delta w_i, \text{median}(\Delta w))$
    if $\text{sim}_i < \tau_{\text{sim}}$ **or** $\mathcal{A}_i(t) > \theta_{\text{anom}}$ **then**
      penalize trust: $T_i(t+1) \leftarrow \lambda_T T_i(t)$
      flag client i as *malicious*
    else
      reward trust: $T_i(t+1) \leftarrow \lambda_T T_i(t) + (1 - \lambda_T)$
    end if
  end for
  if client i dropped out or did not respond **then**
    $T_i(t+1) \leftarrow \lambda_T T_i(t)$ ▷penalize for dropout
    exclude client i from aggregation

(Continued)

Algorithm 2 (continued)

end if
Trust-Weighted Aggregation
normalize trust weights $\tilde{T}_i = \dfrac{T_i}{\sum_{j \in \mathcal{S}_t} T_j}$
update global model $w_{t+1} = \sum_{i \in \mathcal{S}_t} \tilde{T}_i (w_t + \Delta w_i)$
Reward Computation for DRL Agent
compute metrics: accuracy gain ΔAcc_t, mean energy $\bar{E}_t$, drop rate D_t, average trust $\bar{T}_t$
$R_t \leftarrow \text{WeightedReward}(\Delta Acc_t,\ \bar{E}_t,\ D_t,\ \bar{T}_t)$
Normalize reward: $R_t \leftarrow (R_t - \mu_R)/\sigma_R$
store (S_t, A_t, R_t, S_{t+1}) in replay buffer
if t mod $I = 0$ **then**
update DRL policy θ via PPO (Algorithm 3))
end if
if global accuracy converges or energy threshold violated **then**
break ▷ terminate early if criteria met
end if
end for
return final global model w_T and policy π_θ

Algorithm 3: PPO-based DRL policy update

Require: experience buffer $\mathcal{B}$, policy π_θ, value function V_ϕ, clip range ε, learning rates $\alpha_\theta, \alpha_\phi$

initialize advantage estimator $\hat{A}_t$, epochs N_{epo}
for each epoch =1 to N_{epo} **do**
sample minibatch $\{(S_t, A_t, R_t, S_{t+1})\}$ from buffer
compute return $G_t = R_t + \gamma V_\phi(S_{t+1})$
compute advantage $\hat{A}_t = G_t - V_\phi(S_t)$
compute ratio $r_t = \frac{\pi_\theta(A_t|S_t)}{\pi_{\theta_{\text{old}}}(A_t|S_t)}$
if $\hat{A}_t > 0$ **then**
encourage similar actions: $r_t \leftarrow \min(r_t, 1+\varepsilon)$
else
discourage poor actions: $r_t \leftarrow \max(r_t, 1-\varepsilon)$
end if
compute policy loss: $\mathcal{L}_\pi = -r_t \hat{A}_t$
compute value loss: $\mathcal{L}_V = (V_\phi(S_t) - G_t)^2$
compute entropy term: $\mathcal{L}_H = -\beta \mathcal{H}(\pi_\theta(\cdot|S_t))$
total loss: $\mathcal{L} = \mathcal{L}_\pi + c_V \mathcal{L}_V + c_H \mathcal{L}_H$
update $\theta \leftarrow \theta - \alpha_\theta \nabla_\theta \mathcal{L}$
update $\phi \leftarrow \phi - \alpha_\phi \nabla_\phi \mathcal{L}_V$
end for
clear buffer $\mathcal{B}$
update reference policy $\pi_{\theta_{\text{old}}} \leftarrow \pi_\theta$
return updated policy π_θ

Table 3: Computational complexity summary of ESFL framework algorithms

Algorithm	Main operations	Per-round complexity	Per-epoch/Additional complexity
1	Telemetry, mobility prediction, DRL selection, local training, anomaly detection	$O\Big(N + \sum_{i\in\mathcal{S}_t} e_i\lvert D_i\rvert + \lvert D_i\rvert d_{\mathrm{AE}}\Big)$	local epochs e_i included in per-round
2	Trust updates, similarity checks, aggregation, reward computation, buffer update	$O(\lvert\mathcal{S}_t\rvert \cdot d +$ reward/DRL bookkeeping)	DRL update every I rounds; depends on minibatch size and gradient steps
3	Minibatch sampling, advantage computation, policy/value gradient update	–	$O(N_{\mathrm{epo}} \cdot B \cdot (d_\pi + d_V))$, B = minibatch size, d_π, d_V = network sizes

Note: N = total clients, $\lvert\mathcal{S}_t\rvert$ = selected clients per round, e_i = local epochs, $\lvert D_i\rvert$ = dataset size, d_{AE} = autoencoder cost, d = model dimension, N_{epo} = PPO epochs, B = minibatch size, d_π, d_V = policy/value network dimensions.

4 Results and Discussion

This section presents detailed results and a discussion of the proposed framework for intelligent transportation systems. TensorFlow Federated and PyTorch were used to perform the simulations. One hundred IoT vehicle clients operating under mobility and energy constraints derived from the Udacity Autonomous Vehicle Dataset were deployed. The framework integrates a deep reinforcement learning (DRL) based client-selection mechanism and an autoencoder-based threat-detection module. The evaluation focuses on six key metrics: model accuracy, energy consumption, energy efficiency, threat resilience, communication overhead, and DRL reward convergence. A comparative study of several benchmark FL methods was also performed, including FedAvg, FedProx, MOFedAvg, and Energy-Aware FL.

4.1 Simulation and Network Setup

A comprehensive simulation framework was developed using Python with TensorFlow Federated (TFF), PyTorch, and Stable-Baselines. Experiments were run on a workstation equipped with an Intel Core i9 CPU, 32 GB of RAM, and an NVIDIA RTX 4070 GPU. Table 4 shows the detailed simulation parameters and their description. The network of 100 IoT vehicle nodes has been created with a single central aggregator. Telemetry, sensor readings, and steering features have been selected using a random waypoint mobility model. The energy range for the local training epoch has been set to 0.5 to 1.2 J, while the energy per data transmission has been set to 0.2 to 0.8 J. Simulations of over 200 communication rounds were performed.

The experimental setup was simulated for 100 IoT nodes that represent a practical and widely adopted configuration as per MoFeL, ESAFL, and FedProx models. It is consistent with a typical fuzzy-logic-based system, which usually considers 50-200 clients. The increasing number of nodes does not affect the algorithm's trend but increases the computational load. The training analysis covers an average of 200 rounds, which provides sufficient iterations for convergence analysis. Moreover, the model performance saturates after approximately 150–300 rounds. The Udacity Autonomous Vehicle Dataset has been selected, which contains real-world sensor data and driving scenarios. The mobility features, including acceleration, change of direction, and duration, have been considered with the required parameters for energy consumption, power transmission, and latency. The baseline models, including FedAvg, FedProx, MoFeL, and Energy-Aware FL, have been selected for fair comparison.

Table 4: Simulation parameters

Parameter	Value
Number of clients (ICVs)	100
Central aggregator	1
Dataset	Udacity Self-Driving Car Dataset
Local optimizer	Learning rate = 0.01
Batch size	32
Local epochs per round	5
Communication rounds	200
Speed	10–60 km/h
Energy per local training epoch	0.5–1.2 J
Energy per data transmission	0.2–0.8 J
Simulation duration	200 rounds

4.2 Discussion and Analysis

Fig. 2 illustrates the variation in global model accuracy over 50 communication rounds for the conventional FL and the proposed DRL-based FL models. A gradual increase in accuracy for the conventional FL has been observed, reaching 89% after 50 rounds, while the proposed DRL-FL reached 93%. It shows that mobility-aware optimization affects the learning rate and that faster convergence balances local updates and global aggregation.

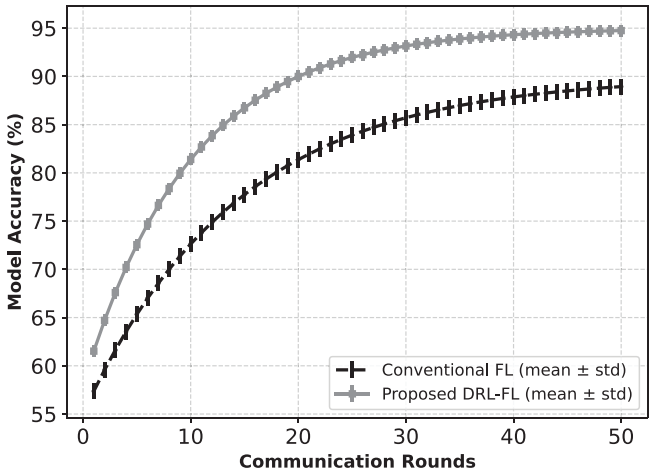

Figure 2: Accuracy vs. communication rounds for FL and DRL-FL. Error bars show standard deviation across rounds

Table 5 shows the comparison results for the proposed DRL-FL with the baseline models. DRL-FL outperforms the other models in terms of the highest accuracy (93.4%), lowest energy consumption (160 J), and highest robustness (86.7%).

The mobility evaluation and the scalability comparison have been shown in Table 6. A gradual decrease in model accuracy and an increase in training latency are observed for both models as the number of IoT nodes increases from 10 to 50. DRL- FL also outperforms FL, maintaining 10% higher accuracy and up to 40% lower latency. The learning performance and energy consumption have been illustrated in the mobility analysis as the vehicle speed increases. The increase in vehicle speed reduces accuracy for both models due

to the lower communication stability and less connection time. DRL-FL comparatively maintains higher accuracy and also consumes less energy.

Table 5: Performance comparison of federated learning algorithms

Algorithm	Accuracy (%)	Energy (J)	Robustness (30% Malicious)
FedAvg	88.4	190	70.5
FedProx	89.7	182	74.3
MOFedAvg	90.9	176	77.1
Energy-Aware FL	91.8	168	81.2
Proposed DRL-FL	93.4	160	86.7

Table 6: Impact of IoT node count on model accuracy and training latency

IoT nodes	Accuracy (%)		Latency (s)	
	FL	DRL-FL	FL	DRL-FL
10	83.5	88.3	2.6	2.0
20	82.0	87.6	3.2	2.3
30	80.6	86.9	3.8	2.6
40	79.2	86.3	4.4	2.9
50	77.9	85.8	5.0	3.2

The energy consumption per communication round has been shown in Fig. 3 for both approaches. The proposed DRL-FL model converges to approximately 150 J, while the FL model converges to approximately 180 J. It shows that the proposed model consumes less energy than the simple FL model up to 16.7%. It is due to the reduced data transmission overhead that ensures the continuous involvement of the IoT nodes.

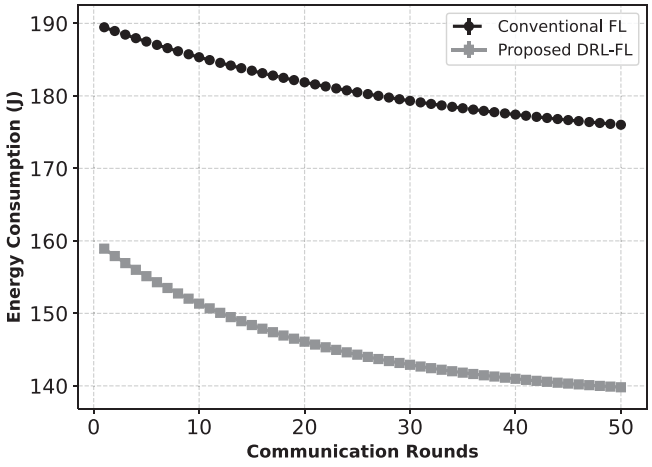

Figure 3: Average energy consumption per communication round. DRL-FL reduces consumption through adaptive selection and power control

The robustness of the proposed framework has been evaluated using parameter sensitivity analysis. Various key design parameters have been assessed, including PPO hyperparameters, trust thresholds, and anomaly-detection parameters. The detailed results are presented in Tables 7–10. Results demonstrated that the proposed model outperforms the baseline models in terms of stability and consistency. A predictable behavior has been observed for PPO in terms of learning rate and discount factor, with aggressive values, while the moderate settings achieved a reasonable balance. The most effective observed parameter is the trust mechanism, with a value range of 0.5–0.7 without increasing the false-positive rate. The anomaly detection parameter demonstrated resilience behavior, avoiding unnecessary exclusion of benign nodes. The overall performance indicates that the framework parameters are not overly sensitive to narrow hyperparameter tuning.

Table 7: PPO hyperparameter sensitivity

Config	Final Acc (%)	Energy (J)	Rounds to Conv.
LR = 1e−4	92.4 ± 0.5	150.2 ± 1.6	142 ± 4
LR = 3e−4	94.1 ± 0.4	148.8 ± 2.0	135 ± 3
LR = 1e−3	90.7 ± 0.9	156.1 ± 2.8	159 ± 6

Table 8: Sensitivity analysis of PPO hyperparameters

Configuration	Accuracy (%)	Energy (J)	Rounds to convergence
LR = 1×10^{-4}	92.4 ± 0.5	150.2 ± 1.6	142 ± 4
LR = 3×10^{-4} (baseline)	94.1 ± 0.4	148.8 ± 2.0	135 ± 3
LR = 5×10^{-4}	93.2 ± 0.6	152.4 ± 2.5	147 ± 5
LR = 1×10^{-3}	90.7 ± 0.9	156.1 ± 2.8	159 ± 6
$\gamma = 0.95$	92.8 ± 0.7	152.1 ± 1.8	146 ± 5
$\gamma = 0.99$ (baseline)	94.7 ± 0.3	148.2 ± 1.9	132 ± 3
$\gamma = 0.995$	93.1 ± 0.4	155.4 ± 3.4	158 ± 7
$\varepsilon = 0.1$	91.9 ± 0.6	151.7 ± 1.5	149 ± 4
$\varepsilon = 0.2$ (baseline)	94.3 ± 0.4	149.1 ± 2.3	134 ± 3
$\varepsilon = 0.3$	93.2 ± 0.5	150.8 ± 2.1	142 ± 6

Table 9: Sensitivity analysis of trust parameters

Setting	Accuracy (%)	Malicious detection rate	False positive rate	Energy (J)
$T_{\min} = 0.3$	89.7 ± 0.8	0.73	0.11	158.9 ± 2.9
$T_{\min} = 0.5$ (baseline)	93.8 ± 0.4	0.87	0.06	148.4 ± 1.7
$T_{\min} = 0.7$	92.1 ± 0.5	0.93	0.17	152.1 ± 2.3
Penalty $\lambda_T = 0.1$	91.2 ± 0.7	0.81	0.09	150.4 ± 2.0
Penalty $\lambda_T = 0.2$ (baseline)	94.4 ± 0.3	0.89	0.05	147.9 ± 1.8
Penalty $\lambda_T = 0.3$	92.9 ± 0.6	0.91	0.14	154.1 ± 2.5

Table 10: Sensitivity analysis of anomaly detection parameters

Threshold/Setting	Accuracy (%)	Detection delay (Rounds)	FPR	Energy (J)
θ_{anom} = 0.4	90.5 ± 0.9	6.3	0.15	160.2 ± 3.1
θ_{anom} = 0.5	92.7 ± 0.6	5.1	0.09	152.7 ± 2.5
θ_{anom} = 0.6 (baseline)	94.9 ± 0.3	3.2	0.05	148.3 ± 2.0
θ_{anom} = 0.7	93.3 ± 0.4	2.8	0.14	155.1 ± 2.8
Window size = 3	92.1 ± 0.5	5.8	0.08	151.9 ± 1.7
Window size = 5 (baseline)	94.5 ± 0.4	3.6	0.06	149.4 ± 2.1
Window size = 10	93.9 ± 0.5	2.9	0.09	154.8 ± 2.3

Fig. 4 exhibits the energy efficiency of various learning configurations. Results indicate that the FL achieves an efficiency of 0.62, while DRL-FL achieves 0.89. It highlights an overall improvement of 43.5%, demonstrating the minimization of redundant computation. Moreover, each training round effectively contributes to the overall learning process without the energy loss.

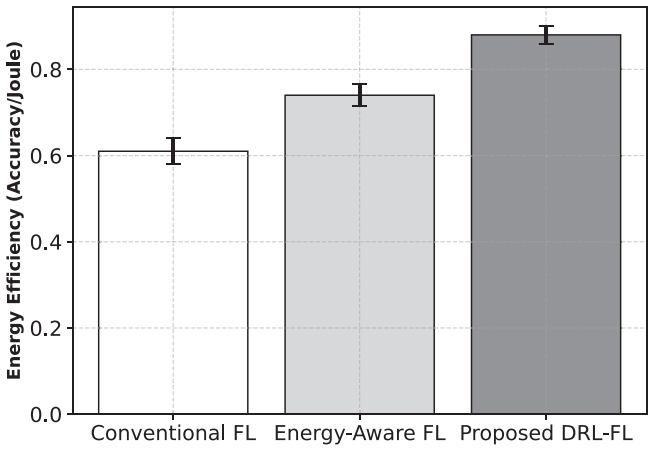

Figure 4: Energy efficiency comparison among baseline algorithms. DRL-FL achieves the highest accuracy-per-joule ratio

The impact of malicious nodes on accuracy has been shown in Fig. 5. FL accuracy decreases gradually from 93% to 69% as the malicious nodes increase from 0%–40%. A minor decrease in the proposed DRL-FL has been observed, thus maintaining the accuracy around 82%. It shows that the proposed model effectively detects suspicious updates, confirming its robust threat detection.

Fig. 6 demonstrates the learning behavior of the DRL agent for 100 training episodes. In the first few episodes, the cumulative reward increased rapidly and stabilized around episode 80. It specifies that the DRL effectively learns to optimize energy and balances competing goals.

The communication efficiency for the FL and DRL-FL models is shown in Fig. 7. The average communication overhead for the FL has been 6.5 MB, while it has been 4.7 MB for the DRL-FL. The nodes are allowed to transmit the model updates using the adaptive share mechanism, which is useful for ITS. It not only reduces the bandwidth but also maintains low-latency communication.

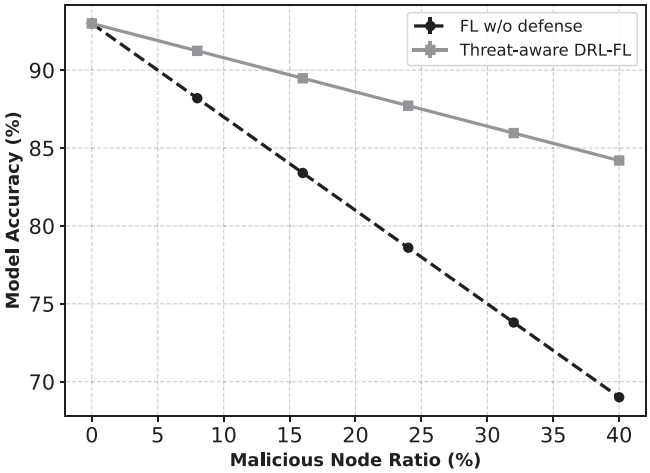

Figure 5: Accuracy degradation under increasing malicious-client ratio. DRL-FL maintains robustness due to trust and anomaly filtering

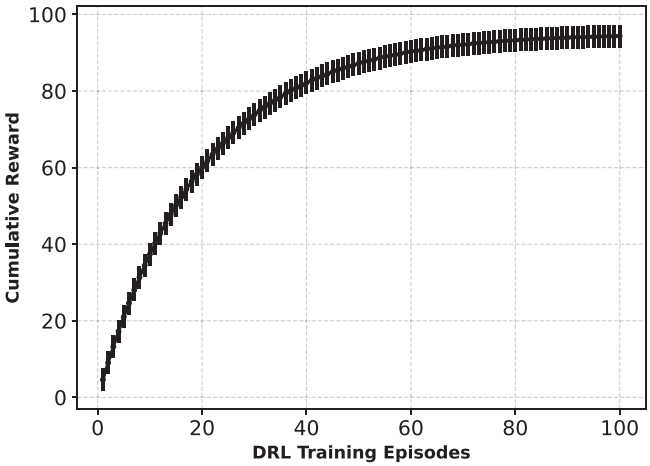

Figure 6: DRL reward convergence across training episodes, showing stable policy learning

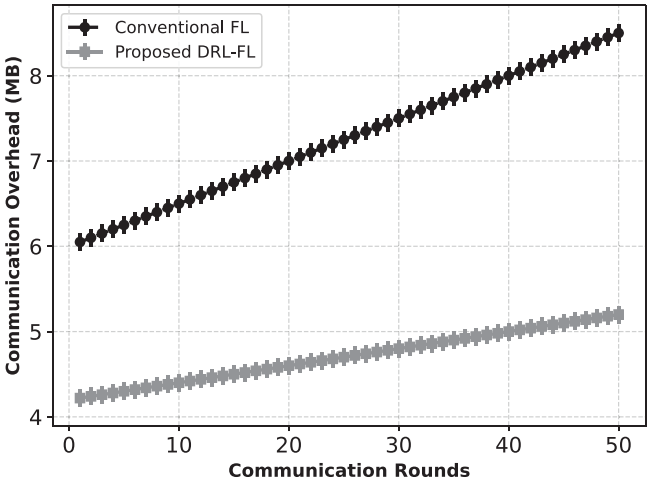

Figure 7: Communication overhead per round. DRL-FL reduces bandwidth usage through optimized participation

The inclusion of core design elements, mobility-aware client selection, dynamic energy management, trust-weighted aggregation, and anomaly detection in the proposed framework leads to superior performance. Algorithms, including FedAvg and FedProx, with static connectivity with rapid change in client availability, may lead to unstable convergence. The proposed DRL-FL model continuously evaluates client stability, avoiding wasting computational resources on potentially disconnected nodes. The issue of overly erratic update quality is avoided by the energy-aware DRL policy, which adjusts the local training workload of battery usage. The trust-weighted aggregation reduces the overall number of malicious model updates based on anomaly scores to prevent malicious influence. The PPO-based policy simultaneously balances accuracy, energy consumption, and robustness, achieving faster convergence and lower energy consumption. The computational cost comparison per communication round for the proposed framework and several models is presented in Table 11.

Table 11: Computational cost comparison per communication round

Method	Training cost (GFLOPs/client)	Aggregation cost (ms/round)	Total runtime (s/round)
FedAvg	1.42	11.3	0.92
FedProx	1.53	12.6	0.98
MOFedAvg	1.61	13.1	1.04
Energy-Aware FL	1.67	14.4	1.09
Proposed DRL-FL	1.83	15.7	1.18

The contribution of each system module for the ablation study has been presented in Table 12. Removing the mobility predictor may increase energy consumption and reduce accuracy by selecting unstable clients. The robustness to adversarial nodes decreases as trust-weighted aggregation is reduced. Convergence can also be delayed by replacing PPO with a heuristic selection strategy, and removing the autoencoder-based anomaly detector increases the exposure to the malicious activities. The analysis confirms that each component in the proposed framework contributes meaningfully to the observed performance metrics.

Table 12: Ablation study: effect of removing modules (mean ± std, $N = 5$ runs)

Configuration	Accuracy (%)	Energy (J)	Rounds to 90%	Robustness (30% mal.) (%)
Full (DRL-FL)	93.4 ± 0.4	160.8 ± 2.1	26 ± 2	86.7 ± 1.2
No mobility model	90.1 ± 0.6	169.4 ± 2.8	38 ± 4	80.2 ± 1.8
No trust mechanism	89.3 ± 0.7	162.0 ± 2.5	41 ± 5	73.5 ± 2.4
No PPO selection	91.0 ± 0.6	166.7 ± 2.6	35 ± 3	78.9 ± 2.0
No anomaly detection	90.5 ± 0.8	164.2 ± 2.9	37 ± 4	75.8 ± 2.6

5 Conclusion

In this study, an integrated Deep Reinforcement Learning (DRL) and Federated Learning (FL) framework has been presented for an intelligent transportation system (ITS). The framework is designed for an IoT environment to address energy efficiency, mobility management, and threat detection issues. The client selection was performed using DRL, while the threat detection mechanism used an autoencoder. Malicious node identification was accomplished for threat identification in a highly dynamic ITS. The proposed model has been trained on factors such as energy availability, trust levels, and mobility dynamics, with a focus on

learning stability and reduced energy consumption. Results reveal that the DRL-FL outperforms the existing model, achieving higher accuracy, lower bandwidth, and less energy loss. The mobility evaluations further endorse the model's effectiveness for different network densities and vehicle speeds. The research can readily be extended to edge computing environments and blockchain-based trust management for dense ITSs.

Acknowledgement: The authors express thanks to Princess Nourah bint Abdulrahman University for supporting this research through the Researchers Supporting Project number (PNURSP2025R510), Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Funding Statement: This research work is supported by Princess Nourah bint Abdulrahman University Researchers Supporting Project No. PNURSP2025R510, Princess Nourah bint Abdulrahman University, Riyadh, Saudi Arabia.

Author Contributions: The authors confirm contribution to the paper as follows: Conceptualization, Hamad Ali Abosaq and Fahad Masood; methodology, Jarallah Alqahtani and Fahad Masood; software, Alanoud Al Mazroa and Muhammad Asad Khan; validation, Muhammad Asad Khan and AKM Bahalul Haque; formal analysis, Hamad Ali Abosaq, Jarallah Alqahtani, and Alanoud Al Mazroa; investigation, Hamad Ali Abosaq, and Alanoud Al Mazroa; resources, AKM Bahalul Haque; data curation, Fahad Masood and Muhammad Asad Khan; writing—original draft preparation, AKM Bahalul Haque and Fahad Masood; writing—review and editing, AKM Bahalul Haque, and Muhammad Asad Khan; visualization, Hamad Ali Abosaq, and Jarallah Alqahtani; supervision, project administration, Hamad Ali Abosaq, and Jarallah Alqahtani. All authors reviewed and approved the final version of the manuscript.

Availability of Data and Materials: The dataset has been obtained from https://public.roboflow.com/object-detection/self-driving-car.

Ethics Approval: Not applicable.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Musa AA, Malami SI, Alanazi F, Ounaies W, Alshammari M, Haruna SI. Sustainable traffic management for smart cities using Internet-of-Things-oriented intelligent transportation systems (ITS): challenges and recommendations. Sustainability. 2023;15(13):9859. doi:10.3390/su15139859.
2. Elassy M, Al-Hattab M, Takruri M, Badawi S. Intelligent transportation systems for sustainable smart cities. Transp Eng. 2024;16(17):100252. doi:10.1016/j.treng.2024.100252.
3. Kairouz P, McMahan HB, Avent B, Bellet A, Bennis M, Bhagoji AN, et al. Advances and open problems in federated learning. Found Trends Mach Learn. 2021;14(1–2):1–210. doi:10.1561/2200000083.
4. Zhang S, Li J, Shi L, Ding M, Nguyen DC, Tan W, et al. Federated learning in intelligent transportation systems: recent applications and open problems. IEEE Trans Intell Transp Syst. 2023;25(5):3259–85. doi:10.1109/tits.2023.3324962.
5. Macedo D, Santos D, Perkusich A, Valadares DC. Mobility-aware federated learning considering multiple networks. Sensors. 2023;23(14):6286. doi:10.3390/s23146286.
6. Alasbali N, Masood F, Alnazzawi N, Ghaban W, Alazeb A, Basurra S, et al. IoT-UAV enabled intelligent resource management in low-carbon smart agriculture using federated reinforcement learning. IEEE Trans Consum Electron. 2025;71(2):6933–41. doi:10.1109/tce.2025.3572552.
7. Masood F, Khan MA, Alshehri MS, Ghaban W, Saeed F, Albarakati HM, et al. AI-based wireless sensor IoT networks for energy-efficient consumer electronics using stochastic optimization. IEEE Trans Consum Electron. 2024;70(4):6855–62. doi:10.1109/tce.2024.3416035.
8. Yang Q, Liu Y, Chen T, Tong Y. Federated machine learning: concept and applications. ACM Trans Intell Syst Technol. 2019;10(2):1–19. doi:10.1145/3298981.

9. Lim WYB, Luong NC, Hoang DT, Jiao Y, Liang YC, Yang Q, et al. Federated learning in mobile edge networks: a comprehensive survey. IEEE Commun Surv Tutorials. 2020;22(3):2031–63. doi:10.1007/978-3-031-07838-5_1.
10. Lamssaggad A, Benamar N, Hafid AS, Msahli M. A survey on the current security landscape of intelligent transportation systems. IEEE Access. 2021;9:9180–208. doi:10.1109/access.2021.3050038.
11. Demertzi V, Demertzis S, Demertzis K. An overview of cyber threats, attacks and countermeasures on the primary domains of smart cities. Appl Sci. 2023;13(2):790. doi:10.3390/app13020790.
12. Asiri F, Malwi WA, Masood F, Alshehri MS, Zhukabayeva T, Shah SA, et al. Privacy preserving federated anomaly detection in IoT edge computing using bayesian game reinforcement learning. Comput Mater Contin. 2025;84(2):3943–60. doi:10.32604/cmc.2025.066498.
13. Xie B, Sun Y, Zhou S, Niu Z, Xu Y, Chen J, et al. MOB-FL: mobility-aware federated learning for intelligent connected vehicles. In: ICC 2023—IEEE International Conference on Communications. Piscataway, NJ, USA: IEEE. p. 3951–7.
14. Jin Z, Yang C, Ye Y, Zhang L, Shen J, Su J. Mobility-aware semi-asynchronous federated learning for vehicular networks. IEEE Trans Veh Technol. 2025. doi:10.1109/TVT.2025.3603978.
15. Chen D, Deng T, Jia J, Feng S, Yuan D. Mobility-aware decentralized federated learning with joint optimization of local iteration and leader selection for vehicular networks. Comput Netw. 2025;263(2):111232. doi:10.1016/j.comnet.2025.111232.
16. Chang X, Obaidat MS, Xue X, Ma J, Duan X. Mobility-aware vehicle selection strategy for federated learning in the internet of vehicles. In: 2024 International Conference on Communications, Computing, Cybersecurity, and Informatics (CCCI). Piscataway, NJ, USA: IEEE; 2024. p. 1–6.
17. Gu X, Wu Q, Fan Q, Fan P. Mobility-aware federated self-supervised learning in vehicular network. Urban Lifeline. 2024;2(1):10. doi:10.1007/s44285-024-00020-5.
18. Saleem M, Arishi A, Farooq MS, Khan MA, Adnan KM. Weighted explainable federated learning for privacy-preserving and scalable energy optimization in autonomous vehicular networks. Egypt Inform J. 2025;31(5):100758. doi:10.1016/j.eij.2025.100758.
19. Almaazmi KIA, Almheiri SJ, Khan MA, Shah AA, Abbas S, Ahmad M. Enhancing smart city sustainability with explainable federated learning for vehicular energy control. Sci Rep. 2025;15(1):23888. doi:10.1038/s41598-025-07844-3.
20. Wang Y, Sui M, Xia T, Liu M, Yang J, Zhao H. Energy-efficient federated learning-driven intelligent traffic monitoring: bayesian prediction and incentive mechanism design. Electronics. 2025;14(9):1891. doi:10.3390/electronics14091891.
21. Firdaus M, Larasati HT. A Blockchain-assisted distributed edge intelligence for privacy-preserving vehicular networks. Comput Mater Contin. 2023;76(3):2959–78. doi:10.32604/cmc.2023.039487.
22. Chen S, Yang L, Shi Y, Wang Q. Blockchain-enabled secure and privacy-preserving data aggregation for fog-based ITS. Comput Mater Contin. 2023;75(2):3781–96. doi:10.32604/cmc.2023.036437.
23. Pacharla N, Srinivasa Reddy K. Trusted certified auditor using cryptography for secure data outsourcing and privacy preservation in fog-enabled VANETs. Comput Mater Contin. 2024;79(2):3089–110. doi:10.32604/cmc.2024.048133.
24. Bakirci M. Internet of Things-enabled unmanned aerial vehicles for real-time traffic mobility analysis in smart cities. Comput Electr Eng. 2025;123(2):110313. doi:10.1016/j.compeleceng.2025.110313.