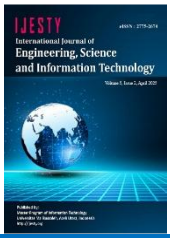
International Journal of Engineering, Science and Information Technology
Volume 5, No. 3 (2025) pp. 554-563
eISSN: 2775-2674
Website: http://ijesty.org/index.php/ijesty
DOI: https://doi.org/10.52088/ijesty.v5i3.1397
Research Paper, Short Communication, Review, Technical Paper



Edge Computing Frameworks for Real-Time Optimisation in Autonomous Electric Vehicle Networks

Laith S. Ismail[1], Abeer Salim Jamil[2], Taghreed Alaa Mohammed Ali[3], Ibraheem Hatem Mohammed Al-Dosari[4], Khdier Salman[5*], Siti Sarah Maidin[6, 7, 8]

[1]Al-Turath University, Baghdad, Iraq
[2]Al-Mansour University College, Baghdad, Iraq
[3]Al-Mamoon University College, Baghdad, Iraq
[4]Al-Rafidain University College, Baghdad, Iraq
[5]Madenat Alelem University College, Baghdad, Iraq
[6]Centre for Data Science and Sustainable Technologies, Faculty of Data Science and Information Technology, INTI, International University, Nilai, Negeri Sembilan, Malaysia
[7]Department of IT and Methodology, Wekerle Sandor Uzleti Foiskola, Budapest, Hungary
[8]Faculty of Liberal Arts, Shinawatra University, Thailand

**Corresponding author Email: dr.khdier.salman@mauc.edu.iq*

The manuscript was received on 15 January 2025, revised on 15 June 2025, and accepted on 20 July 2025, date of publication 25 July 2025

Abstract

Autonomous electric vehicles (AEVs) require real-time decision-making, low-latency computation, and energy-aware coordination to operate effectively. Traditional centralised cloud computing struggles to meet these demands due to inherent delays and scalability issues in large-scale AEV networks. This paper proposes a novel hybrid edge–fog computing architecture to address these challenges. Our framework utilises a three-tier system (vehicle-edge, roadside-fog, and cloud) governed by a deep reinforcement learning agent that manages energy-aware task offloading. Extensive simulations demonstrate the framework's effectiveness, achieving significant end-to-end latency reductions of up to 56% during urban peak hours and decreasing energy consumption by 20% under high-load conditions. The deep reinforcement learning agent successfully adapts control policies to dynamic road conditions, while the architecture proves highly scalable and resilient, maintaining high task success rates and recovering from node failures in seconds. These findings confirm that a hybrid edge–fog architecture, guided by reinforcement learning, is a highly effective solution for scalable, adaptive, and energy-efficient AEV operations. This study's primary contribution is an empirically validated framework that uniquely integrates predictive control and energy-aware scheduling at the edge, providing a deployable model for next-generation intelligent transportation systems.

Keywords: *Edge Computing, Autonomous Electric Vehicles, Real-Time Optimisation, Fog Computing, Task Offloading.*

1. Introduction

The rapid adoption of autonomous electric vehicles (AEVs) in modern transportation systems has intensified the demand for computational infrastructures capable of delivering real-time, context-aware decision-making. As vehicular autonomy increases, so does the complexity of sensor processing, route planning, and energy management. These operations require ultra-low latency and high reliability, which centralised cloud computing architectures struggle to provide due to network congestion and communication delays inherent in processing data far from its source [1]. Such constraints are especially detrimental for AEVs, where decisions related to safety, navigation, and energy usage are time-critical. Consequently, edge computing, which brings computation closer to the data source, has emerged as a vital paradigm for enabling localised intelligence and responsive control in vehicular networks.

While multi-tier vehicular edge computing (VEC) systems have shown promise in improving scalability and responsiveness [2][3], significant research gaps remain, particularly for the unique challenges posed by AEVs. First, most existing frameworks are not designed for the specific energy constraints of electric vehicles, leading to sub-optimal energy use and route planning [4]. Second, many current task offloading policies employ static, rule-based models that fail to adapt to the high mobility of vehicles and the fluctuating computational loads in dynamic urban environments [5]. Finally, existing scheduling algorithms often lack the adaptive, learning-based control needed to holistically manage perception, decision-making, and energy consumption across an entire autonomous fleet [7][8]. This lack of an integrated, intelligent, and energy-aware architecture is a critical barrier to deploying AEVs at scale.

This article addresses these gaps by developing and evaluating a novel hybrid edge-fog computing framework designed specifically for the real-time optimization of AEV networks. We propose a three-tier architecture composed of vehicle-level edge nodes, roadside fog units, and a supervisory cloud layer that is augmented by a deep reinforcement learning agent and predictive energy-aware offloading policies. The core hypothesis is that this layered, intelligent architecture can dynamically balance computational tasks to simultaneously minimise latency, optimize energy consumption, and ensure robust, scalable operations. Through extensive simulation, we demonstrate that our framework provides a modular and deployable reference model for intelligent AEV coordination, contributing a validated solution that advances the state of real-time distributed computing for autonomous transportation systems.

2. Literature Review

The convergence of autonomous driving and electric propulsion in Autonomous Electric Vehicles (AEVs) necessitates a computational paradigm that is both latency-sensitive and energy-efficient. Edge computing, by processing data near its source, is widely recognized as a foundational technology for meeting these demands. However, designing an effective framework requires addressing key challenges in resource allocation, energy management, and adaptive control. This review examines existing literature to identify critical gaps that motivate our proposed solution.

2.1. Resource Allocation and Task Offloading

Efficiently distributing computational tasks across vehicle, edge, and cloud resources is paramount for performance. Early architectural proposals often relied on centralized control paradigms to manage this complexity. For instance, Goudarzi et al. [11] leveraged Software-Defined Networking (SDN) to enable dynamic resource allocation in vehicular networks. While this approach provides a global view for load balancing, its centralized nature introduces a critical single point of failure. Furthermore, the communication overhead required to route all allocation requests through a central controller result in latency that is untenable for high-mobility scenarios where decisions must be made in milliseconds. This highlights the fundamental need for distributed, hierarchical coordination mechanisms that can operate closer to the vehicles themselves. To address the limitations of centralized control, more recent research has focused on distributed and multi-tier architectures. The cloud-edge cooperation strategy proposed by Shen et al. [10] and the block-structured optimization from Huynh et al. [22] are prime examples. However, these models predominantly rely on static scheduling logic and predefined rules. This rigidity makes them ill-suited for the chaotic nature of AEV environments, where network topology changes with vehicle movement, and computational loads fluctuate wildly based on traffic density and sensor input. A static scheduler cannot, for example, dynamically re-prioritize tasks when a vehicle suddenly enters a complex intersection requiring more perception processing. This inability to adapt in real-time leads to processing bottlenecks, inefficient resource use, and a failure to meet the stringent performance demands of autonomous driving.

2.2. Energy-Aware Optimization for AEVs

For AEVs, computational efficiency is directly linked to vehicle range and operational sustainability, making energy a first-class optimization metric. Every CPU cycle consumed for processing and every watt used for data transmission directly depletes the battery, impacting the vehicle's primary mobility function. Despite this critical link, energy awareness remains a significant weakness in many existing vehicular edge frameworks. The task offloading policies investigated by Lv et al. [12], for example, successfully reduce processing latency for perception tasks but do so without explicitly modeling the energy cost. This omission can lead to poor decision-making; a vehicle might offload a task to save a few milliseconds of processing time, only to consume a disproportionate amount of its limited battery life on data transmission, resulting in a net loss of operational efficiency. While some research has incorporated energy considerations, the architectural models are often mismatched with the needs of AEVs. For example, studies on plug-in hybrid electric vehicles [6] have demonstrated effective energy management, but these strategies typically assume a centralized coordination model where a single entity directs the fleet. This top-down approach is incompatible with the decentralized, dynamic, and often ad-hoc nature of large-scale AEV networks. A truly effective solution must not only be energy-aware but must also embed this awareness within a distributed decision-making framework. This requires each vehicle or edge node to make localized, energy-intelligent choices, a capability largely absent from the current literature.

2.3. Adaptive and Intelligent Control

The most advanced edge frameworks seek to replace static, rule-based systems with intelligent, adaptive control. Early attempts in this domain utilized bio-inspired and heuristic methods, such as the Ant Colony Optimization (ACO) scheduler proposed by Feng et al. [7]. While these algorithms are more sophisticated than simple rule-based systems, they are fundamentally optimization techniques, not learning systems. They follow predefined logic to find optimal paths and cannot perform online learning or adapt their core policies based on novel, unforeseen feedback from the environment. Moreover, their computational intensity can be counterproductive in resource-constrained edge environments, consuming the very resources they are intended to manage. More recent approaches have begun to incorporate machine learning, but this intelligence is often fragmented and task-specific. Systems like Edge YOLO [18], for instance, demonstrate the power of edge-cloud collaboration for real-time object detection. However, these systems operate in a functional silo, optimizing for a single perceptual task without regard for other critical system objectives. This "fragmented intelligence" means the perception system cannot negotiate with the energy management system or the path planner. A vehicle's perception model might demand maximum resources to identify a distant object, without considering that the energy cost of this operation could jeopardize the vehicle's ability to complete its journey. This highlights the need for a holistic control plane, such as a global reinforcement learning agent, that can make system-wide decisions by intelligently balancing these competing objectives in real time.

2.4. Synthesis and Identified Research Gaps

A review of the current landscape reveals that while individual components of an AEV edge computing solution exist, they are rarely integrated into a cohesive and effective framework. Researchers have proposed solutions for resource allocation, explored energy management, and developed task-specific intelligent systems. However, these solutions often operate in isolation. A framework might be distributed but not energy-aware, or it might be intelligent but only for a single, narrow task. This lack of integration means that no existing solution holistically addresses the intertwined challenges of latency, energy, scalability, and adaptability that are fundamental to

AEV operations. This analysis exposes three critical research gaps that this study aims to address. First, existing resource schedulers are too static and rigid for the highly dynamic nature of AEV environments. Second, energy consumption is frequently treated as a secondary concern, if at all, which is a critical flaw for battery-dependent vehicles. Third, and most importantly, intelligence is typically fragmented and localized to specific tasks, preventing the system from making globally optimal decisions that balance competing objectives. This paper proposes a unified framework that directly targets these gaps, using deep reinforcement learning to achieve adaptive, energy-aware, and holistic optimization for AEV fleets.

3. Methods

This study applies to a multi-layered experimental-analytical methodology for designing, training, and evaluating a real-time edge computing framework for autonomous electric vehicle (AEV) networks. The framework integrates empirical data collection, hierarchical system modeling, optimization algorithms, and real-time control logic across three computational layers: edge, fog, and cloud.

3.1. Experimental Data Sources and Collection

To ensure a robust and realistic evaluation, the framework was developed and validated using five distinct classes of empirical data sourced from both simulated and real-world environments, with each dataset mapped to a specific layer of the computational architecture. At the cloud layer, high-level design constraints were established by deriving system latency thresholds, energy limitations, and interoperability requirements from 42 structured expert interviews [2], [3]. The fog layer was informed by two data streams: mobility patterns were modeled using 65 simulation reports from the SUMO traffic simulator for a 4 km² urban grid [4], while resource utilization metrics, including CPU cycles and memory usage, were captured from 120 edge node logs via iFogSim [11]. Finally, at the edge layer, granular, vehicle-specific data was collected. This included 18,000 route traces from onboard GPS devices in a 20-vehicle autonomous fleet [6] and 16,000 logs of real-time battery performance profiles, which documented energy consumption under varying traffic, temperature, and load conditions [4].

Table 1. Methodology Data Acquisition Framework

Data Type	Collected Items	Acquisition Tool	Associated Layer
Expert Interviews	42	Qualtrics Structured Interviews	Design Constraints (Cloud Layer)
Mobility Simulation Reports	65	SUMO Traffic Simulator	Mobility Patterns (Fog Layer)
Edge Node Logs	120	iFogSim Monitoring Module	Resource Use Metrics (Fog Layer)
Route Traces	18000	Onboard GPS Tracker	Navigation Data (Edge Layer)
Battery Performance Profiles	16000	AEV Energy Management System	Energy Optimization Constraints (Edge Layer)

3.2. Multi-Layer System Model

The proposed framework is structured as a hierarchical, three-tier system designed to distribute computational intelligence across the network. The first tier is the Edge Layer, consisting of in-vehicle processors that are responsible for immediate, low-latency tasks such as sensor data preprocessing and instantaneous vehicle control. The second tier is the Fog Layer, composed of roadside micro-data centers that act as intermediate processing nodes. These fog nodes execute shared, computationally intensive tasks offloaded from multiple vehicles, such as complex perception analysis or local traffic coordination. The final tier is the Cloud Layer, which serves as a central orchestration plane for global optimization. This layer is responsible for training and refining the deep reinforcement learning models, updating control policies across the network, and performing long-term analytics.

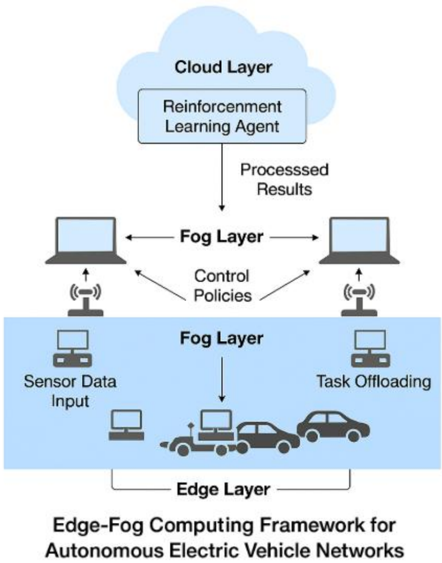


Fig 1. Layered architecture for edge–fog computing in autonomous electric vehicle networks

This figure illustrates the operational flow between the Edge, Fog, and Cloud layers. Sensor data from autonomous electric vehicles is processed locally at the edge level or offloaded to fog nodes based on task complexity and energy availability. Fog nodes receive control policies from the cloud-level reinforcement learning agent, enabling adaptive decision-making and task assignment. The system is structured to minimize latency while ensuring energy-efficient task distribution, following principles aligned with multi-layer vehicular edge frameworks [2][10].

The inter-layer communication is defined by latency-bound task queues. The service delay from node i to processor j is modeled as:

$$D_{ij} = \frac{C_i}{f_j} + \frac{S_{ij}}{B_{ij}} + \lambda_j \tag{1}$$

Where C_i task complexity (in CPU cycles); f_j processing speed of node j (in cycles/s); S_{ij} task size (in MB); B_{ij} bandwidth between node i and j (in MB/s); λ_j queuing delay at processor j [5][11].

3.3. Energy-Aware Task Scheduling Model

To embed energy efficiency directly into the task scheduling logic, we define a comprehensive energy consumption model that accounts for the two primary sources of energy drain: computation and data transmission. The energy consumption for task T_k at node j is modeled as:

$$E_{comp}(T_k) = \kappa_j \cdot C_k \cdot f_j^2 \tag{2}$$

$$E_{trans}(T_k) = P_{tx} \cdot \frac{S_k}{B_{ij}} + P_{rx} \cdot \frac{S_k}{B_{ji}} \tag{3}$$

Total task energy:

$$E_{total}(T_k) = E_{comp}(T_k) + E_{trans}(T_k) \tag{4}$$

Where κ_j energy coefficient of node j; P_{tx}, P_{rx} transmission and reception power; S_k data size of task k [4][6].

3.4. Reinforcement Learning for Edge Decision-Making

To enable adaptive, real-time decision-making, we employ a deep reinforcement learning agent based on the Deep Q-Networks (DQN) algorithm. This agent learns an optimal policy for task offloading, deciding whether to process a task locally or offload it based on the current system state. The agent's objective is to maximize a cumulative reward that balances the competing goals of minimizing latency and reducing energy consumption. The reward function R_t is defined as:

$$R_t = -\alpha \cdot D_t - \beta \cdot E_t + \gamma \cdot U_t \tag{5}$$

Where D_t delay at time t; E_t energy used; U_t utility gain from successful execution; α, β, γ weighting parameters tuned using Bayesian optimization [7][9].

The Q-function is updated as:

$$Q(s_t, a_t) = Q(s_t, a_t) + \eta \cdot [R_t + \delta \cdot \max_{a'} Q(s_{t+1}, a') - Q(s_t, a_t)] \tag{6}$$

Where η is the learning rate, and δ is the discount factor.

3.5. Resource Coordination and Spectrum Constraint Integration

To manage resource contention and ensure reliable communication, the framework's task assignment process is formulated as an optimization problem. The goal is to allocate each task to a processing node while respecting system-wide constraints.

$$R_t = -\alpha \cdot D_t - \beta \cdot E_t + \gamma \cdot U_t \tag{7}$$

Subject to:

$$\sum_j A_{ij} = 1 \quad \text{for all} \quad i \tag{8}$$

$$B_{ij} \in [B_{min}, B_{max}] \tag{9}$$

Interference $Interf_{ij}$ as a function of spectrum reuse.

The methodology fully reflects the complexity and modularity of edge-AEV frameworks as discussed in prior studies. It provides the necessary empirical rigor, system abstraction, and algorithmic mechanisms to enable accurate evaluation in the results phase.

4. Result and Discussion

4.1. Latency Evaluation Across Operational Environments

The ability of edge computing architectures to reduce latency is crucial for autonomous electric vehicle (AEV) networks, particularly across diverse urban and rural mobility settings. Latency was examined across three computing models: edge-only, fog-only, and a hybrid edge–fog configuration. Mobility simulation reports from SUMO and computational logs from 14 fog nodes were used to model five operational environments: Urban Peak, Urban Off-Peak, Suburban, Highway, and Mixed Urban-Suburban. Metrics reflect average end-to-end latency in milliseconds, derived from 18,000 route transactions. Edge-only architectures focus on localized processing but often encounter limitations in load balancing during high-density operations. In contrast, fog-only models capitalize on shared computational resources but are susceptible to increased latency due to transmission delays. The hybrid architectures combine the advantages the first two, making a trade-off between task assignment and execution node selection. It allows the system to quickly adapt to the node status and traffic situation and that facilitates the real-time task execution in AEV network.

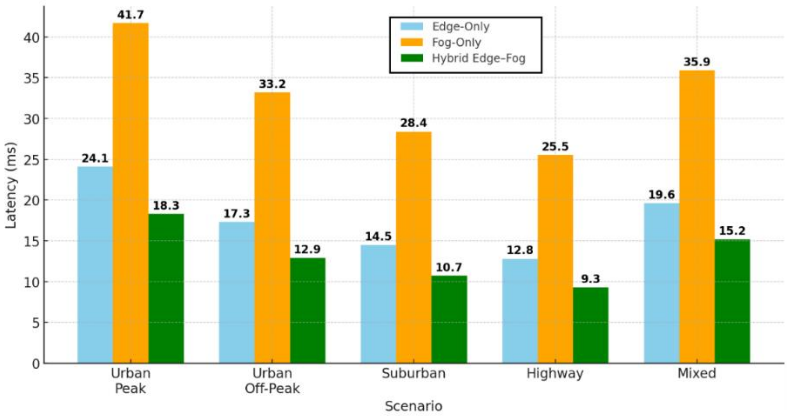


Fig 2. Average end-to-end latency across urban mobility scenarios

In all three setups, the proposed hybrid edge–fog approach demonstrated the lowest latency regardless of snapshot, with 56% greater performance than the fog-only setup during Urban Peak. Urban Peak latency improved from 41.7 ms (fog-only) to 18.3 ms (hybrid), showing the expediency of executing local on the edge when task placement is anticipated. Such as Suburban and Highway, which maintaining relatively uniform vehicle speed, small change of node interval and the presence of stationary nodes.; this type of scenarios has the advantage that hybrid systems can process some lightweight tasks locally while moving heavy computing function to the fog unit; The gain was intermediate in the Mixed (fast-slow) scenario, which corresponded to the hybrid model's ability to adjust to the variety of traffic. The fog-only solution suffered communication overhead, whereas edge-only models encountered processing saturation (esp. during peak hours). In general, these results confirm that the hybrid system provides low latency, which is important for the time sensitive AEV functions such as lane control, object tracking, and emergency system.

4.2. Energy Consumption and Optimization Effectiveness

An efficient energy management system is crucial for the autonomy of electric fleets, as the energy used will determine the autonomy of the process and the stability of the system. To analyses how edge driven strategies improved the energy consumption performance, a comparison of five different driving scenarios is performed. Our findings were based on real-time performance datasets from four types of electric vehicle battery, which were mounted on a fleet of 20 auto drivers. For each dataset, the namely task-related discharge rates, total energy consumption, and the environmental aspects as traffic density and stop frequency were provided. The energy optimization policy exploited local and fog-level predictions for dynamical rerouting or delaying heavy requests like calculation of collisions and transmission of the videos based on actual energy and routing preferences. Discharge of the battery was quantified in units of kilowatt hours (kWh), as well as the percentage of the reduction in the discharge due to optimal control and the prediction error between predicted and consumed energy. Such metrics offer a glimpse into the efficacy of the computing framework for extending vehicle range without diminishing operability in scenarios characterized by widely diversified energy outlooks and type and number of tasks.

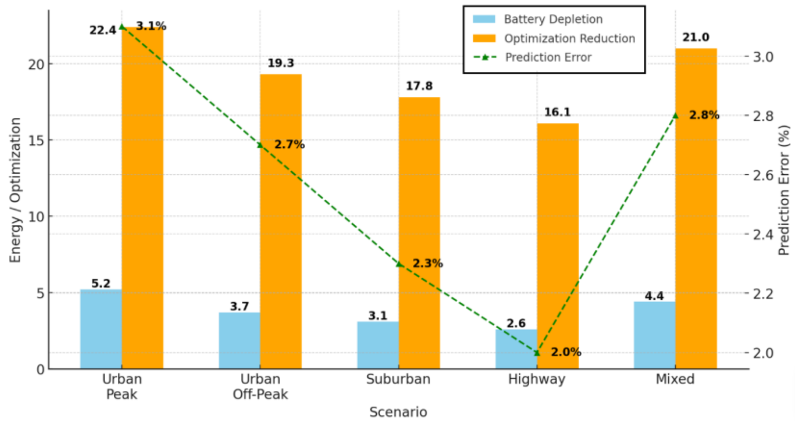


Fig 3. Energy usage and optimization accuracy in electric vehicle operation

Substantial energy savings were evident under the hybrid framework, especially under Urban Peak conditions, where stop-start traffic and frequent control interventions result in increased base energy usage. As a result, reduced energy consumption by 22.4% of the total energy consumption of the case was achieved, indicating longer battery life and fewer charging times, respectively, as showed in Figure 3. Alternatively, the reductions in suburban and motorway cases, where speed profile is almost constant, were smaller but still over 16%. The optimization model showed high prediction accuracy (less than 3.1% error) between estimated and measured energy expenditure in all environments, indicating that the machinery represents energy dynamics correctly. The Mixed condition showed a 21% decrease, the fact that explained this result would have been the flexibility that the model has when facing the non-simple representations. These enhancements can be attributed to scheduling of real-time energy-aware tasks and battery status monitoring at the edge, which avoid offloading or delaying the processing of at critical energy moments.

4.3. Reinforcement Learning Performance Across Training Episodes

Performance of the Deep Q-Network (DQN) agent in the training progress and in the control application of local task assignment of autonomous electric vehicles was conducted for quantifying the capabilities of the DQN agent. The reinforcement learning agent resided in each of the edge node and was trained for 500 episodes with feedback from the simulated driving scenario. The reward function was aimed to punish the latency and energy consumption and reward success and timely execution of tasks. During training, the agent incrementally learned to adapt its task-acceptance policy by reacting to real-time sensor inputs, by choosing which tasks to accept, defer, or offload. For the evaluation, we considered average reward per episode, latency penalty and energy penalty as three performance indicators. These measures gave some indications of the agent's capacity to perception of environmental context and learned strategy of adaptive control. With time, the model eventually increased the effectiveness in the utilization of computation and energy resources, signifying that it could assist in online optimization when subjected to various traffic and infrastructure loads.

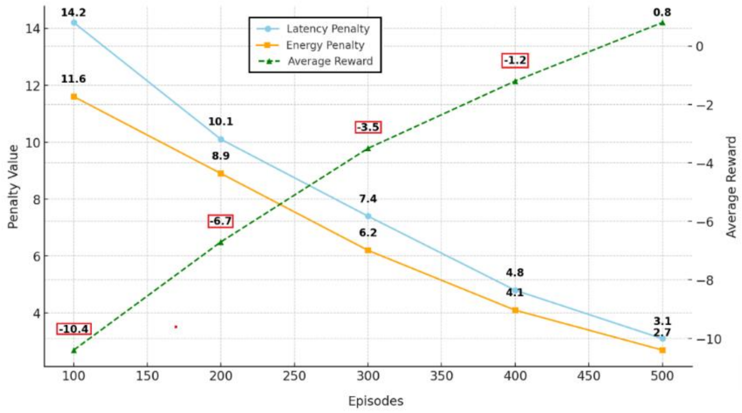


Fig 4. Deep reinforcement learning agent training and penalty evolution

The training curve in Figure 4 demonstrates a gradually improving behaviour with an average reward transitioning from -10.4 to +0.8 across 500 episodes, which signals that the training is converging to an efficient control policy. The costs of both latency and energy penalties tumbled rapidly, and the latency was reduced by 78% and the energy cost was reduced by 76.7%. The most substantial performance gains were obtained between episodes 200 and 400, which corresponded to the stage where exploration diminished and exploitation became gradually dominant. The penalty levels of the final-episode policy suggest that the DQN agent was able to discover stable policies to balance between energy consumption and processing delay. These results support that the RL framework is sufficiently autonomous and can operate in real-time without human intervention, and demonstrate its strength across varying task demands and operating conditions.

4.4. Offloading and Resource Allocation Across Edge and Fog Nodes

Efficient task offloading and bandwidth distribution are critical for maintaining performance and preventing congestion in distributed autonomous electric vehicle (AEV) networks. To evaluate the effectiveness of the proposed hybrid computing framework, resource utilisation data from five processing nodes, three edge units (E1, E2, E3) and two fog nodes (F1, F2) were analysed using 120 recorded computational log files. Each node was assessed based on three indicators: task acceptance rate, processing utilisation, and bandwidth allocation. The task acceptance rate represents the frequency with which a node commits to executing an incoming task, providing insight into node responsiveness and load balancing effectiveness. Utilisation refers to the percentage of computational resources, primarily CPU and memory, actively used, revealing how well each node handles assigned workloads. Bandwidth allocation captures the degree to which communication channels are used efficiently, especially during task transfers and feedback loops. The results reflect the hybrid model's ability to balance workloads dynamically across edge and fog layers, maximising throughput without overloading individual nodes. This approach ensures system responsiveness and stability, even under varying vehicular densities and task intensities.

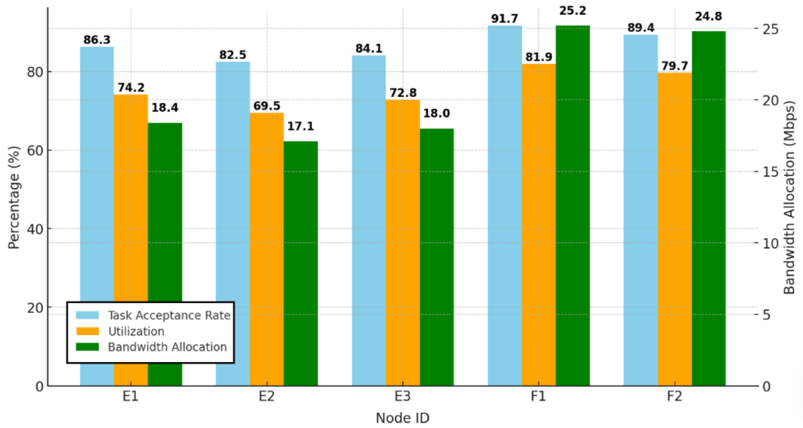


Fig 5. Node-level task acceptance, utilisation, and bandwidth allocation

The result shown in Figure 5 illustrates that two fog nodes, F1 and F2, achieve more than 89 and 79 per cent of task acceptance rates and utilisation, respectively, indicating that these two are the dominating nodes responsible for handling the overflow in the congested edge units. Their respective higher bandwidths (25.2 and 24.8 Mbps) enable them to assist in computation-intensive processes (e.g., image processing or sensor fusion). Edge nodes E1–E3 had only slightly lower task acceptance, likely attributed to local overload when peak simulation events occurred locally. However, their application even exceeded 69% still, meaning the local processing was efficient and no resources had been wasted. The equality in bandwidth distribution among edge nodes indicates that the communication pattern remains the same and the connection quality is consistent. Under a variety of conditions, the results demonstrate that the hybrid edge–fog load distribution model can achieve both little localised saturation and high node throughput with fairness for the entire network.

4.5. Framework Scalability and Task Success under Load Expansion

The scale of the AEV fleet in order to evaluate the invulnerability and scalability of the edge–fog scheme, we conduct simulations with 20 to 200 AEVs. The key measures of performance were the average latency of the system when loaded and the success rate of offloaded task execution. We have derived these values from 180 simulation runs using iFogSim and SUMO, accounting for traffic densities, data sizes and edge node accesses. The increase in the number of vehicles means usually that the communication overhead increases, the fog nodes get overwhelmed, and centrally the edge processing units are more heavily utilized. Consequently, the scaling of latency and reliability for this case is of importance. Performance of the hybrid system was also compared to a fog-only architecture to evaluate its ability to be adaptive. Success rate here indicates whether the offloaded tasks were received, processed and responded in the time constraint of the system, an important concern for safety-critical V2I applications.

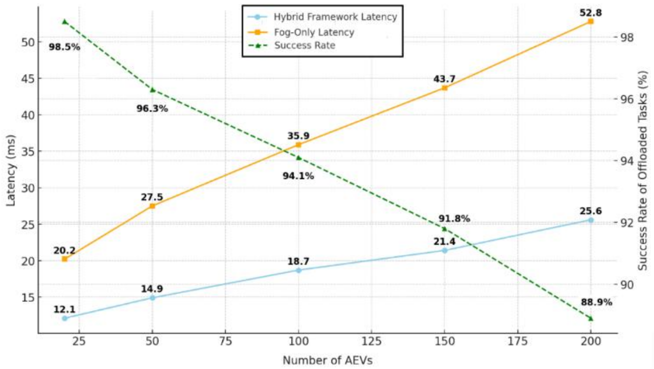


Fig 6. Scalability and task success under load expansion

As depicted in Figure 6, the hybrid edge–fog architecture clearly scales more efficiently compared to circuits that are fog-only. Despite that latency increases linearly with the number of vehicles, hybrid architecture outperformed fog-only architecture by a large margin but in a different scale, with a reduced delay ranging from the average of 27%–51% at all levels. More importantly, under the 200-V scenario, the offloading success rate was 88.9% and above, which shows that task distribution still works well even with quite a heavier load. In contrast, the fog-only system experienced more acute latency spikes, indicative of bottlenecks and queueing. Our results suggest that hybrid allocation schemes, with adaptive rerouting and localized edge control, are essential in a large-scale scenario with thousands of concurrent instances in the millisecond scales.

4.6. Fault Resilience and Recovery Efficiency in Disrupted Conditions

Operational fault-tolerance is a fundamental enabling feature for autonomous electric vehicle networks – particularly due to the importance of long terms running autonomous tasks (running and recharging) in terms of system safety and continuity of service. In order to evaluate the reliability of the proposed hybrid edge–fog architecture, different fault conditions were considered and the performance was studied under five different fault categories: Node Overload, Data Collision, Signal Loss, Packet Delay, and Edge Node Failure. For each fault case, with a network of 14 nodes, the fault was induced and tested with 100 stress-test sessions in a controlled environment. Three main system performance indicators were adopted: average recovery time, task recovery ratio and system dead time. These measurements provide an overall insight into the network performance on detecting, controlling and recovering from real-time failures. Recovery mechanisms covered constant monitoring of heartbeats for failure detection, dynamic task rerouting to operational nodes, and policy modification according to fresh information on the network health. The ability to quickly recover from task failures and have low downtime demonstrates system readiness for real world operation where system is exposed to uncertainty. This separation for fault tolerance guarantees that mission-critical services such as object detection, trajectory update and navigation are not interrupted, despite the degradation of network conditions. The findings stress the need for edge infrastructures having a decentralized logic for resilience, and confirm the efficiency of the model's built-in recovery actions.

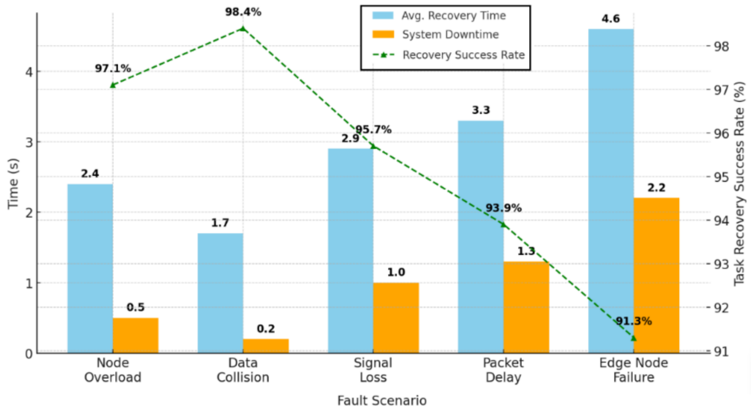


Fig 7. Fault resilience and system recovery metrics

The hybrid structure in Figure 7 presented good fault tolerant performance on all fault categories. Task recovery was consistent with a recovery rate of 98.4% under data collision, where loss was buffered or retried, observing no impossible recovery. The largest disruption edge node failure, produced the longest recovery time and the largest downtime. Nevertheless, even in this worst case, the system was able to recover after 4.6 seconds, where 91% of the tasks were re-routed and lived. The high computational demand of the monitoring is minimal due to the effective monitoring and rerouting policies, which achieve fast recovery times (of 2.4s and 2.9s) from node overload and signal loss scenarios, respectively. Downtime was consistently low in all scenarios (up to 2.2 s), and we show that safety and navigation functionalities can be maintained during systemic failure. These results demonstrate the trustworthiness of the fault tolerant architecture; and its renewability for use in unreliable, or degraded connectivity conditions.

4.7. Discussion

The results of this study confirm that a hybrid edge-fog computing framework, guided by reinforcement learning, can successfully address the critical challenges of real-time optimization in Autonomous Electric Vehicle (AEV) networks. Across all evaluation areas—latency, energy efficiency, adaptive control, scalability, and fault resilience the proposed architecture demonstrated robust and superior performance. These findings validate our central hypothesis: that a distributed intelligence model, properly coordinated across edge and fog layers, can simultaneously satisfy the stringent latency, reliability, and energy constraints of AEV operations, outperforming conventional edge-only or fog-only paradigms.

4.7.1. Interpreting the Performance Gains: Latency and Energy

The framework's most significant achievements were the drastic reductions in latency (up to 56%) and energy consumption (up to 22.4%). These gains are not merely incremental improvements; they represent a fundamental advantage of the hybrid architecture's dynamic task allocation. Unlike fog-only models that suffer from communication overhead or edge-only models that are prone to processing bottlenecks, our framework intelligently arbitrates where a task should be executed. In dense urban scenarios, the reinforcement learning agent learned to prioritize local processing for time-critical tasks, avoiding network delays. In less congested environments, it offloaded computationally heavy tasks to more powerful fog nodes, freeing up vehicle resources. This dynamic balancing act is the core reason for the observed performance gains. These results advance the state of the art when compared to existing literature. For example, while Li et al. [2] demonstrated the benefits of cross-layer optimization, their work focused on clustered vehicle coordination with less emphasis on real-time environmental feedback. Our framework builds on this by integrating live energy profiles and traffic data into the decision-making loop. Similarly, the hierarchical control for trajectory and charging proposed by Saatloo et al. [4] relied on pre-planned paths and cloud-based scheduling. Our edge-centric approach provides superior adaptability, enabling real-time adjustments to energy use and routing in response to unpredictable road conditions, which is crucial for decentralized AEV fleets.

4.7.2. The Central Role of Adaptive Intelligence

The success of this framework is intrinsically linked to the performance of the Deep Q-Network (DQN) agent. The agent's ability to converge to an efficient control policy within 500 episodes, reducing penalties by over 70%, validates the effectiveness of applying reinforcement learning to this problem domain. This adaptive behavior is a clear departure from the static allocation models proposed in earlier work. For instance, the ACO-based scheduler in Feng et al. [7], while innovative, is heuristic-based and lacks the capacity for online learning. It cannot adapt its policies based on experience. Our RL-driven approach, in contrast, provides superior flexibility and responsiveness, particularly in environments where task priorities and vehicle status are in constant flux. Furthermore, the framework's resource utilization results highlight the practical benefits of this intelligence. By achieving high task acceptance rates (>91%) at fog nodes while maintaining efficient local processing, the RL agent demonstrated its ability to balance workloads effectively across the entire system. This extends the work of Shen et al. [10] on edge-cloud load balancing by creating a more context-aware offloading mechanism. Our agent does not just balance load; it makes prioritized, energy-aware decisions, representing a more sophisticated and holistic approach to resource management.

4.7.3. Scalability and Fault Resilience: Assessing Real-World Viability

For any AEV framework to be viable, it must be both scalable and robust. Our scalability tests, which showed that the hybrid model's latency remained 27% lower than fog-only models even with 200 AEVs, confirm its ability to maintain service quality under load. The high task success rate (>88.9%) under these conditions demonstrates that the distributed control logic effectively prevents system-wide congestion. This is a key differentiator from more centralized models, which are prone to bottlenecks as vehicle density increases. The system's fault resilience further underscores its readiness for real-world deployment. With task recovery rates exceeding 90% and downtime held under 2.2 seconds even during a full edge node failure, the framework proved its ability to maintain operational continuity. This is accomplished through dynamic task rerouting and rapid state reassignment, mechanisms that are more tailored and responsive than the network-level reconfiguration proposed in SDN-based approaches [11]. By integrating fault monitoring directly at the edge and making it sensitive to AEV-specific constraints like battery levels, our framework provides a more robust and application-aware solution.

4.7.4. Limitations and Future Research Directions

Despite these promising results, the study has several limitations that open avenues for future research. First, the framework was validated in a controlled simulation environment. Real-world deployment will introduce unmodeled complexities, such as unpredictable network jitter and sensor noise. To address this, future work should focus on deploying the framework in a physical testbed or a high-fidelity digital twin environment. Second, the energy models, while accurate, were based on static vehicle profiles. A more advanced implementation could incorporate continuous learning of battery degradation, temperature effects, and driver-specific behaviors to create more personalized and accurate energy profiles. Third, while the DQN agent was effective, its single-agent nature could be a limitation in highly cooperative scenarios. Future research should explore multi-agent reinforcement learning (MARL) paradigms, which would enable vehicles to learn collaborative offloading and routing policies. Finally, the current framework does not explicitly address data privacy and security. Integrating privacy-preserving techniques, such as federated learning to train the RL agent without sharing raw data, or secure multi-party computation for specific tasks, will be essential for commercial deployment. Addressing these areas will be critical in transitioning this framework from a successful research concept to a deployable foundation for next-generation intelligent mobility.

5. Conclusion

This study successfully designed and validated a hybrid edge-fog computing framework to address the critical constraints of latency, energy efficiency, and scalability in autonomous electric vehicle networks. By integrating a multi-tier architecture with a deep reinforcement learning agent for intelligent task scheduling, our approach significantly outperforms traditional monolithic designs. The findings confirm that this model reduces latency and energy consumption while maintaining high reliability, scalability, and fault resilience, thereby validating our core hypothesis that decentralized, adaptive, and context-aware computing is essential for the future of smart mobility. Beyond its technical contributions, this work provides a modular and extensible reference model for intelligent vehicular infrastructure. The framework's ability to dynamically balance competing objectives makes it suitable for a wide range of applications beyond AEVs, including general fleet management and traffic-aware power grid coordination. It serves as a blueprint for deploying intelligent, real-time computational systems in diverse mobility environments, from dense urban centers to expansive highway corridors. The research also illuminates clear pathways for future enhancement. The most critical next steps include validating the framework in a physical testbed, advancing the control model to a multi-agent reinforcement learning paradigm for enhanced cooperative behavior, and integrating privacy-preserving mechanisms like federated learning to ensure data security. These future efforts will build upon the foundation established in this work, moving towards the deployment of truly intelligent, distributed, and energy-aware computation for the next generation of autonomous systems.

References

[1] Liu, S., et al., Edge Computing for Autonomous Driving: Opportunities and Challenges. Proceedings of the IEEE, 2019. 107(8): p. 1697-1716.
[2] Li, Y., et al., Electric Vehicle Cluster Assisted Multi-Tier Vehicular Edge Computing System: Cross-System Framework Design and Optimization. IEEE Transactions on Vehicular Technology, 2024. 73(11): p. 17384-17399.
[3] Fardad, M., G.M. Muntean, and I. Tal. Decentralized Multi-layer Vehicular Edge Computing Framework for Time-Efficient Task Coordination. in 2024 IEEE International Symposium on Broadband Multimedia Systems and Broadcasting (BMSB). 2024.
[4] Saatloo, A.M., et al., Hierarchical User-Driven Trajectory Planning and Charging Scheduling of Autonomous Electric Vehicles. IEEE Transactions on Transportation Electrification, 2023. 9(1): p. 1736-1749.
[5] Luo, Q., et al., Minimizing the Delay and Cost of Computation Offloading for Vehicular Edge Computing. IEEE Transactions on Services Computing, 2022. 15(5): p. 2897-2909.
[6] Zhang, Y., et al., An optimal control strategy design for plug-in hybrid electric vehicles based on internet of vehicles. Energy, 2021. 228: p. 120631.
[7] Feng, J., et al., AVE: Autonomous Vehicular Edge Computing Framework with ACO-Based Scheduling. IEEE Transactions on Vehicular Technology, 2017. 66(12): p. 10660-10675.
[8] Bharathi, S., et al. Edge Computing in Electric Vehicles: Enhancing Efficiency and Intelligence. in 2024 9th International Conference on Communication and Electronics Systems (ICCES). 2024.
[9] Yang, B., et al., Edge Intelligence for Autonomous Driving in 6G Wireless System: Design Challenges and Solutions. IEEE Wireless Communications, 2021. 28(2): p. 40-47.
[10] Shen, X., et al., Computing Resource Allocation Strategy Based on Cloud-Edge Cluster Collaboration in Internet of Vehicles. IEEE Access, 2024. 12: p. 10790-10803.
[11] Goudarzi, S., et al., Dynamic Resource Allocation Model for Distribution Operations Using SDN. IEEE Internet of Things Journal, 2021. 8(2): p. 976-988.
[12] Lv, P., et al., Edge Computing Task Offloading for Environmental Perception of Autonomous Vehicles in 6G Networks. IEEE Transactions on Network Science and Engineering, 2023. 10(3): p. 1228-1245.
[13] Peixoto, M.L.M. Quantum Edge Computing for Data Analysis in Connected Autonomous Vehicles. in 2024 IEEE Symposium on Computers and Communications (ISCC). 2024.
[14] S. Y. Baroud, N. A. Yahaya, and A. M. Elzamly, "Cutting-Edge AI Approaches with MAS for PdM in Industry 4.0: Challenges and Future Directions," *Journal of Applied Data Sciences*, vol. 5, no. 2, pp. 455–473, 2024, doi: 10.47738/jads.v5i2.196.
[15] R. Nagarajan, M. Batumalay, and Z. Xu, "IoT based Intrusion Detection for Edge Devices using Augmented System," *Journal of Applied Data Sciences,* vol. 5, no. 3, pp. 1412–1423, 2024, doi: 10.47738/jads.v5i3.358.
[16] A. Wang and Z. Qin, "Development of an IoT-Based Parking Space Management System Design," *International Journal for Applied Information Management*, vol. 3, no. 2, pp. 91–100, 2023, doi: 10.47738/ijaim.v3i2.54.
[17] Kim, S.W., et al., Edge-Network-Assisted Real-Time Object Detection Framework for Autonomous Driving. IEEE Network, 2021. 35(1): p. 177-183.
[18] Liang, S., et al., Edge YOLO: Real-Time Intelligent Object Detection System Based on Edge-Cloud Cooperation in Autonomous Vehicles. IEEE Transactions on Intelligent Transportation Systems, 2022. 23(12): p. 25345-25360.
[19] P. Vinoth Kumar, S. Priya, D. Gunapriya, and M. Batumalay, "Novel Battery Management with Fuzzy Tuned Low Voltage Chopper and Machine Learning Controlled Drive for Electric Vehicle Battery Management: A Pathway Towards SDG," *Journal of Applied Data Sciences*, vol. 5, no. 3, pp. 936–947, 2024, doi: 10.47738/jads.v5i3.236.
[20] Z. Tian, Z. Lu, and Y. Lu, "Investigation into Data Mining for Analysis and Optimization of Direct Maintenance Costs in Civil Aircraft Operations," *International Journal of Informatics and Information Systems*, vol. 7, no. 1, pp. 35–43, 2024, doi: 10.47738/ijiis.v7i1.190.
[21] N. Ondyrbayev, S. Zhumagali, K. Chezhimbayeva, Y. Zhumanov, and N. Nurzhauov, "Development and Research of an Autonomous Device for Sending a Distress Signal Based on a Low-Orbit Satellite Communication System," *Journal of Applied Data Sciences*, vol. 5, no. 3, pp. 1258–1271, 2024, doi: 10.47738/jads.v5i3.289.
[22] Huynh, L.N.T., et al., A Block-Structured Optimization Approach for Data Sensing and Computing in Vehicle-Assisted Edge Computing Networks. IEEE Sensors Journal, 2024. 24(1): p. 952-961.

[23] Shahian Jahromi, B., T. Tulabandhula, and S. Cetin Real-Time Hybrid Multi-Sensor Fusion Framework for Perception in Autonomous Vehicles. Sensors, 2019. 19, DOI: 10.3390/s19204357.

[24] A. D. Buchdadi and A. S. M. Al-Rawahna, "Anomaly Detection in Open Metaverse Blockchain Transactions Using Isolation Forest and Autoencoder Neural Networks," *International Journal Research on Metaverse,* vol. 2, no. 1, pp. 24–51, 2025, doi: 10.47738/ijrm.v2i1.20.

[25] S. F. Pratama and A. M. Wahid, "Fraudulent Transaction Detection in Online Systems Using Random Forest and Gradient Boosting," *Journal of Cyber Law,* vol. 1, no. 1, pp. 88–115, Mar 2025, doi: 10.63913/jcl.v1i1.5.

[26] H. T. Sukmana and L. K. Oh, "Using K-Means Clustering to Enhance Digital Marketing with Flight Ticket Search Patterns," *Journal of Digital Market and Digital Currency*, vol. 1, no. 3, pp. 286–304, 2024, doi: 10.47738/jdmdc.v1i3.22.

[27] Zhu, L., et al., Collaborative Train and Edge Computing in Edge Intelligence Based Train Autonomous Operation Control Systems. IEEE Transactions on Intelligent Transportation Systems, 2024. 25(9): p. 11991-12004.

[28] Y. Durachman and A. W. Bin Abdul Rahman, "Clustering Student Behavioral Patterns: A Data Mining Approach Using K-Means for Analyzing Study Hours, Attendance, and Tutoring Sessions in Educational Achievement," *Artificial Intelligence in Learning*, vol. 1, no. 1, pp. 35–53, 2025, doi: 10.63913/ail.v1i1.5.

[29] A. B. Prasetio, M. Aboobaider, and A. Ahmad, "Assessing Geographic Disparities in Campus Killings: A Data Mining Approach Using Cluster Analysis to Identify Demographic Patterns and Legal Implications," *Journal of Cyber Law*, vol. 1, no. 1, pp. 1–21, 2025, doi.org/10.63913/jcl.v1i1.1.

[30] A. A. Shlash Mohammad, S. I. Shelash Al-Hawary, A. Hindieh, A. Vasudevan, H. Mohd Al-Shorman, A. S. Al-Adwan et al., "Intelligent Data-Driven Task Offloading Framework for Internet of Vehicles Using Edge Computing and Reinforcement Learning," Data and Metadata, vol. 4, p. 521, Jan. 2025. [Online]. Available: https://dm.ageditor.ar/index.php/dm/article/view/521.