Capuchin Search-Enhanced Deep Q-Learning with CNN-Based Feature Extraction for Autonomous Driving in New Energy Vehicles

Wei Huang
Puyang Vocational and Technical College, Henan puyang, 457000, China
E-mail: WeiHuang121@outlook.com

Keywords: new energy vehicles, autonomous driving, neural networks, capuchin search-driven scalable Deep Q-Network (CapSA-SDQN), decision control, energy efficiency

Received: October 11, 2025

Autonomous driving technology enhances the energy efficiency and safety of New Energy Vehicles (NEVs) through intelligent driving decisions. Traditional methods focus on longitudinal velocity planning and fixed-route strategies, which do not adapt well to dynamic traffic scenarios. The research introduces a decision control method for autonomous driving that combines Neural Networks (NNs) with Reinforcement Learning (RL) to address the traditional limitations. In the proposed framework, convolutional neural networks (CNNs) extract spatial-temporal features from multi-sensor inputs, including camera, Light Detection and Ranging (LiDAR), and radar data, to capture lane geometry, vehicle distances, and obstacle dynamics using python. Min-Max normalization was utilized as a preprocessing technique to scale all characteristics to a uniform range. Autonomous Driving Sensor-Kinematics Data were collected. These features form the state space for a Capuchin Search-driven Scalable Deep Q-Network (CapSA-SDQN)-based RL agent, which simultaneously manages lane-changing and vehicle-following behaviors and optimizes decision control in complex and dynamic traffic environments. The decision control method further incorporates a rule-based safety checker that is embedded downstream to guarantee safe maneuver execution. The strategy is trained and tested in the simulation platform under diverse traffic scenarios, including congested urban intersections, highway merging, and dynamic overtaking. The proposed CapSA-SDQN model achieved a high decision accuracy of 98.5% while maintaining a fast response time of 121 m/s. The findings demonstrate that integrating NNs with RL provides a scalable solution for NEV autonomous driving in real-world environments.

Povzetek: Raziskava predstavlja metodo avtonomne vožnje za električna vozila, ki združuje nevronske mreže in okrepljeno učenje za prilagodljivo odločanje v dinamičnem prometu ter dosega visoko natančnost in hitro odzivnost.

1 Introduction

NVs have autonomous driving technology that increases energy efficiency, safety, and management. It maximizes the usage of energy, minimizes collisions, and react to the changing traffic conditions by making informed decisions based on the analysis of the real-world environment. Decision control mechanisms are the brain of the intelligent vehicles, which guarantees safety in overtaking and collision avoidance [1-2]. Traditional decision control systems are often grounded on rule-based or optimization systems. These dimensions need adaptive systems which are learning based with dynamic environment [3]. Self-driving cars are based on the principles of complex data processing through the use of new computational methods that provide opportunities to predict and make quick judgments during decision making [4]. Induction motor vehicle involves an electrical generator motor to give effective, dependable, and inexpensive propulsion [5]. The use of RL is a powerful decision-making tool during autonomous vehicles where the system can interact with the environment to learn the best policies [6]. Through integrating NNs in both perception and state representation and sequential decision-making, the method has the advantage of tackling variability and uncertainty [7]. NNs serve as a policy approximators, which transform sensor input into actions to allow decision-making through RL [8]. This adaptability enables the autonomous driving systems to learn on simulated and real-world cases reducing the dependence on manual regulations and improving the process of navigation along different urban and highway conditions [9].

Previous research on autonomous driving decision control used either rule-based models or standalone NNs. Whereas standalone neural networks are very effective in

perception, they are weak at long-term decision optimization in dynamic environments [10]. The classical method exhibits serious problems such as high computing complexity and absence of convergence assurance in real world situations, impeding real-world applications [11]. This research aims to create an adaptive and energy-efficient autonomous drive decision controller for NEVs through the combination of Convolutional Neural Networks (CNN) and Capuchin Search-based Scalable Deep Q-Network (CapSA-SDQN) that allows the autonomous driver to make robust lane-changing and vehicle-following decisions, at the same time guaranteeing safety, and enhancing traffic efficiency during dynamic conditions. As below follow the research questions.

1. How effective is the proposed CapSA-SDQN framework in improving decision accuracy and response time for autonomous driving in NEVs under dynamic traffic environments?

2. How does the integration of Capuchin Search optimization influence policy convergence, stability, and cumulative reward performance in RL-based decision control?

3. What is the impact of the rule-based safety checker on collision prevention, maneuver reliability, and overall safety performance during simulated real-time NEV driving scenarios?

2 Related works

Research developed a scientific and comprehensive risk assessment technique for the energy vehicle supply chain [12]. It combined a hierarchical holographic model with a substance element extension model for risk evaluation. Validation of scalability under diverse industrial contexts remains to be fully tested. The research optimized service latency and energy consumption in vehicular networks by addressing multi-access edge computing (MEC) servers for Unmanned Aerial Vehicles (UAVs) [13]. The method reduced system cost and latency, and achieved high task completion rates. Real-world deployment challenges include scalability and communication overhead. Research maximized energy efficiency (EE) in a UAV-assisted by optimizing communication scheduling, resource allocation, and UAV trajectory [14]. High computational complexity may hinder scalability. The research optimized energy consumption in UAV-assisted multi-access edge computing (MEC) networks under dynamic resource conditions [15]. A multi-UAV-multiuser MEC model was created employing MADDPG and PER for cooperative decision-making. Simulations showed improved convergence and reduced network energy consumption, although scalability challenges beyond simulations remain unresolved. Table 1 shows related works on decision control methods for autonomous NEVs.

Table 1: Summary of existing methods for autonomous NEV decision control

Citation	Objective	Method	Performance Metrics	Limitations
Pervez et al. [16]	Minimize energy & latency in dynamic decision settings	Multi-stage optimization + communication/computation control	Accuracy: 90–92%; Response Time: 160–180 m/s; Energy Efficiency 12%	High computational cost; limited real-time adaptability
Asim et al. [17]	Improve task completion & energy efficiency	Adaptive trajectory planning + resource optimization	Decision Accuracy: 91%; Latency Reduction: 18%	Scalability issues in dynamic environments
Xie et al. [18]	Energy-efficient task scheduling in mobility	Geometry-driven control + divide-and-conquer scheduling	Energy Savings: 15–20%; Stability: Moderate	Lacks real-time adaptability; unsuitable for dynamic traffic
Zhao et al. [19]	Joint optimization for safe and efficient decisions	Lyapunov-based optimization + RL	Decision Accuracy: 94.5%; RT: 140 m/s; Safety ↑ 21%	High complexity; real-world scalability limited
Li et al. [20]	Improve selection & resource allocation under mobility	MINLP + MDP solved with AB-DDQN	Accuracy: 93%; Latency: 150 m/s; Energy Efficiency 10%	Requires high computational resources
Chen et al. [21]	DRL-based resource allocation for real-time decisions	Deep Reinforcement Learning (UMAP algorithm)	Accuracy: 95.2%; RT: 134 m/s; Energy Efficiency 14%	Simulation-only validation

Zhang [22]	Develop an adaptive EMS for NEVs to improve fuel economy, maintain SOC, and reduce battery aging.	MARL-based SSB-MADQN	Reduced fuel use by 0.912 L (WLTC) and 0.681 L (HWFET)	Real-world testing and broader driving scenarios are still required.
Li et al. [23]	Multi-objective energy-efficient decision optimization	Self-attention PPO + trajectory optimization	Accuracy: 95–97%; Energy Efficiency 17%	High model complexity
Zhang [24]	Improved decision-making using GA-based optimization	Enhanced Genetic Algorithm	Loss reduction: 0.012; Task Efficiency 22%	Limited performance in rapidly changing environments

2.1 Research gaps

Previous studies have some limitations, such as Pervez et al. [16] assessed high computational cost and reduced scalability because of simplified models; Asim et al. [17] exhibits high computational complexity and reduced scalability; Research [18] exhibits mobility dynamics and real-world deployment issues unaddressed; research [19] exhibits high complexity and reduced scalability. The proposed CapSA-SDQN solution is capable of overcoming these drawbacks by delivering a computationally efficient scalable decision control model of NEVs, which is able to process dynamic traffic and multi-sensor data.

2.2 Problem formulation

In the proposed autonomous driving framework, multi-sensor perception data from camera, LiDAR, and radar are processed using a convolutional neural network to generate a compact state representation ($w_{j|s}^{ref}$). Based on this state, the CapSA-SDQN selects a high-level maneuver, such as lane-keeping, lane-changing, or vehicle-following. Once a maneuver is chosen, a lower-level controller computes the optimal sequence of control inputs ($v_{j|s}^{\intercal}$) over a prediction horizon ($\intercal$). The objective is to minimize deviations from the reference trajectory generated by the RL agent, while simultaneously reducing control effort to ensure smooth and energy-efficient maneuver execution. This objective is expressed as in Equation (1).

$$\min_{v_{j|s}, j=0:S-1} I = \sum_{j=0}^{S-1}\left(w_{j|s}^{ref} - w_{j|s}\right)^{\intercal} R\left(w_{j|s}^{ref} - w_{j|s}\right) + v_{j|s}^{\intercal} Q v_{j|s} \quad (1)$$

To ensure accurate motion prediction, the evolution of the ego vehicle (E_{ego}) along the virtual prediction horizon follows the NEV dynamic model while the surrounding vehicles evolve according to a prediction model represented as Equation (2).

$$S.t. w_{j+1}|s = E_{ego}(w_{j|s}, v_{j|s}) \quad (2)$$

Safety is guaranteed through collision-avoidance constraints that enforce sufficient separation between the ego vehicle and every surrounding vehicle in the set (i), expressed by an ellipsoidal safety condition (C_{road}^{safe}) along with a road-edge safety margin (Equation 3).

$$w_{j+1|s}^{i} = E_{pred}(w_{j|s}^{i}), (w_{j|s} - w_{j|s}^{i})^{S} N(w_{j|s} - w_{j|s}^{i}) \geq C_{veh}^{safe}, (w_{j|s} - w_{j|s}^{road})^{S} N(w_{j|s} - w_{j|s}^{road}) \geq C_{road}^{safe} \quad (3)$$

A rule-based safety checker enforces traffic regulations, including mandatory stopping at red lights (K_{stop},), implemented through the constraint (Equation 4).

$$w_{j|s} \leq K_{stop}, if\ light = red \quad (4)$$

The initial values for the virtual trajectory are assigned using the real states of the ego vehicle and surrounding vehicles, and the current control input applied at step (s), described as Equation (5).

$$w_{0|s} = w_s, w_{o|s}^{i} = w_s^{i}, v_{0|s} = v_s \quad (5)$$

With the indices spanning the entire prediction horizon as determined in Equation (6),

$$j = 0 : S - 1, i \in J \quad (6)$$

In the hierarchical formulation, the CNN-based state extraction, CapSA-SDQN maneuver selection, dynamic prediction, energy-aware control optimization, and rule-based safety verification together form a unified decision-control architecture for NEV autonomous driving.

3 Methodology

The research aims to design an autonomous driving decision control approach that is adaptive and energy efficient in NEVs. Multi-sensor inputs such as camera, LiDAR, and radar are processed by synchronization, noise removal, normalization, and time warping to generate structured state representations. The proposed framework combines CNNs and CapSA-SDQN (Figure 1). CNNs extract spatial-temporal features reflecting the geometry of the lanes, distances between vehicles, and dynamic obstacles. The CapSA-SDQN agent is a reward-based RL agent for lane-changing and vehicle-following behaviours, and simulated traffic conditions Reinforcement Learning (RL) enables the agent to learn optimal driving actions through interaction with traffic environments, improving adaptability and decision-making in dynamic NEV scenarios.

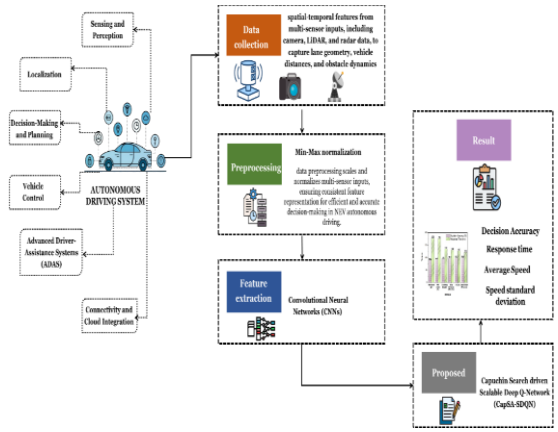

Figure 1: Schematic representation of proposed flow

3.1 Data collection

Autonomous Driving Sensor-Kinematics Data were collected from the open source Kaggle. (https://www.kaggle.com/datasets/ziya07/entrepreneurial-ability-assessment-dataset). The realistic Autonomous Driving Sensor-Kinematics Dataset was created to aid in research and development in the areas of risk prediction, path planning, and autonomous vehicle decision-making. It combines vehicle kinematics, ambient context, and navigation-related characteristics with RGB-D simulated sensor data (LiDAR, camera, radar). The dataset, which provides a balanced mix of traffic situations, environmental variables, and obstacle encounters, is specifically designed for machine learning model training and assessment. The model was trained over structured scenario splits with 3000 episodes and defined termination conditions.

3.2 Data preprocessing using min-max normalization

Preprocessing is the process of transforming raw data into a clean, standardized, and structured format to improve performance. Min-Max normalization is utilized to normalize the values of all features to a fixed range [0,1]. It improves the stability and accuracy of the decision-making in CapSA-SDQN-based NEV autonomous driving, as shown in Equation (7).

$$W_{new} = \frac{W - \min(w)}{\max(w) - \min(w)} \tag{7}$$

W_{new} denotes the adjusted value derived from the normalized outcomes, W represents the previous value, $\max(w)$ evaluates the dataset's maximum value and $\min(w)$ denotes the dataset's minimum value. Effective preprocessing normalizes multi-sensor inputs and removes noise for reliable NEV autonomous driving decisions.

3.3 Feature extraction using convolutional neural networks (CNNs)

The process of turning unprocessed sensor data into insightful representations that emphasize crucial patterns like lane geometry, for efficient decision-making, is known as feature extraction. A Convolutional Neural Network (CNN) is a deep learning model that automatically extracts spatial and visual features from input data using convolutional layers, enabling accurate pattern recognition in images and sensor signals. In this research CNN is used to provide spatial-temporal features of multiple sensors. Equation (8) illustrates the CNN-based feature extraction for creating state representations in decision control for NEV autonomous driving.

$$f_k(x,y) = \sigma(\sum_m \sum_n I\,(x+m, y+n) \cdot W_k(m,n) + b_k) \tag{8}$$

The convolution operation $f_k(x,y)$ applies learnable kernels $W_k(m,n)$, bias b_k, and activation σ to highlight lane geometry, and obstacle dynamics, where $I(x+m, y+n)$is the input value at a shifted position, $\Sigma_m\Sigma_n$ denotes an overall value of m and all values of n within the filter's receptive field. Figure 2 and table 2 denotes a structure of CNN.

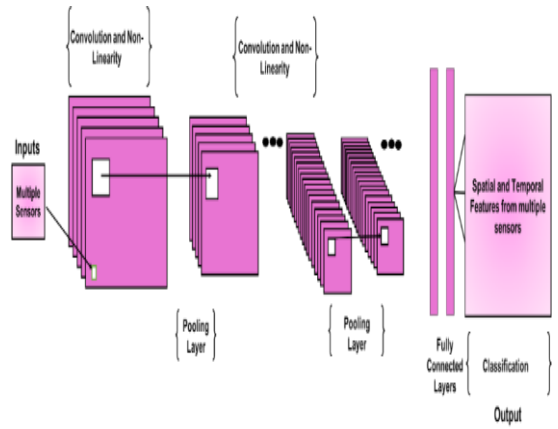

Figure 2: Illustration of CNN architecture

Table 2: Detailed CNN Architecture

Layer Type	Configuration
Conv2D	32 filters, 3×3 kernel, ReLU
Max Pooling	2×2
Conv2D	64 filters, 3×3 kernel, ReLU
Max Pooling	2×2
Conv2D	128 filters, 3×3 kernel, ReLU
Flatten	–
Dense	256 units, ReLU
Output (State embedding)	128-dimensional vector

3.4 Capuchin Search-based Scalable Deep Q-Network (CapSA-SDQN)

The CapSA-SDQN is a combination of DQN RL and a CS optimization algorithm that optimizes exploration of a large action space. This mixed solution increases the policy convergence and safer autonomous operations.

3.4.1 Scalable Deep Q-Network (SDQN)

Deep Q-Network (DQN) integrates Q-learning with deep neural networks to determine action-values in high dimensions states but suffers from Q-value overestimation, unstable training, and high computational demands, limiting real-time reliability. The SDQN is a deep RF model that is used in the autonomous driving of NEVs and allows them to make decisions efficiently in dynamic traffic. Equation (9) depicts the formulation of a cumulative discounted incentive for safe and energy-efficient decision-making.

$$S_u = \sum_{u'-u}^{U} \gamma^{u'-u_{su,}}$$

(9)

Here, cumulative discounted reward S_u for the autonomous driving agent from time step u to U. Here, γ is the discount factor, weighting future rewards, $\gamma^{u'-u_{su,}}$ and $u'-u$ captures the temporal horizon. It guides the CapSA-SDQN agent to optimize energy-efficient, safe driving decisions over time. Equation (10) shows the agent's expected reward function, which assesses safe and energy-efficient driving based on traffic conditions.

$$R^{UU}(t,b) = F[S|T_u = T, b_u = b, \pi]$$

(10)

The above equation represents the expected reward of the autonomous driving agent under a given state S, $T_u$denotes the user time variable set to t, behavior parameter set to b, vehicle behavior $b_u = b, \pi$, and the policy R^{UU}. It quantifies decision effectiveness for energy-efficient and produced by the function F. Equation (11) is the anticipated cumulative reward function for NEV decision-making.

$$R^*(t,b) = F_{t'\sim t}[\, S + \gamma max_b \cdot R(T',b')r, b]$$

(11)

The expected cumulative reward $R^*(t,b)$ for the autonomous NEV at time $t' \sim t$ and state b. It calculates the future reward by considering all possible next states $R(T',b')$, weighted by the immediate reward S, discounted future rewards γmax_b, and the probability of state transitions. Equation (12) guides the CapSA-SDQN agent to optimize energy efficiency.

$$R^*(t,b) = max_\pi\, F[S_u|T_u = t, b_u = b, \pi]$$

(12)

The equation represents the optimal expected reward for the autonomous NEV at time F with battery level b by max_π denotes maximization taken over all possible policies . The function $FS_u|T_u$ evaluates energy efficiency, safety, and traffic performance S_u. Equation (13) depicts the SDQN loss function, which directs the agent to maximize energy efficiency.

$$M = F[(s + \gamma maxR(t', a', \theta^-) - R(t,b,\theta)^2]$$

(13)

Equation (7), M denotes the computed loss value, s represents the immediate reward, γdenotes the discount factor, t'signifies the next time step, a'denotes the next action considered for maximization, θ^-represents the target network parameters, and θdenotes the current network parameters.

3.4.2 Capuchin Search (CapSA)

The CapSA is a type of optimization algorithm inspired by the foraging behavior of capuchin monkeys, which entails

improving exploration and parameter tuning of CapSA. Its role ensures more stable decision-making, faster policy learning, and improved adaptability of the autonomous driving agent in complex NEV traffic environments. Equation (14) shows NEV autonomous vehicle control through adaptive decision variable updating.

$$y_i^j = G_i + \frac{q_{ag}\left(u_i^j\right) sin(2\theta)}{h} \tag{14}$$

This models the positional update of an agent y_i^j, where G_i represents the guiding position, and the second term adjusts movement based on adaptive search dynamics. The factor $q_{ag}\left(u_i^j\right)$ with $sin(2\theta)$ introduces stochastic exploration, while h controls scaling, and supports the CS driven optimization within the RL agent. Equation (15) states the representation in NEV autonomous driving decision control using an adaptive scaling factor.

$$\theta = \frac{3}{2}s \tag{15}$$

The parameter $\theta = \frac{3}{2}$ represents the adaptive scaling factor applied to the state space s, ensuring balanced representation of spatial-temporal landscapes from multi-sensor inputs. Equation (16) highlights the position updating rule for NEV autonomous driving decision control in CS optimization.

$$u_i^j = qu_i^j + b_1\left(y_{besti}^i - y_i^j\right)s_1 + b_2\left(G_i - y_i^j\right)s_2 \tag{16}$$

The equation updates the candidate solution u_i^j by combining the current state with guidance from individual best experience $y_{besti}^i - y_i^j$ and global best solution G_i. Coefficients b_1 and b_2 control the influence of personal and global knowledge, while s_1 and s_2 adjust search dynamics. This guides the CapSA-SDQN agent toward energy-efficient and safe driving decisions. The adaptive control parameter for NEV strikes a balance between exploration and exploitation, Equation (17).

$$\rho = v_{max} - (v_{max} - v_{min})\frac{j}{J} \tag{17}$$

The adaptive control parameter ρ is defined by the equation, and it is dependent on the iteration index j within the total number of iterations J, that the vehicle speed changes between $v_{max} - v_{min}$. By balancing exploration and exploitation in the RL process, this dynamic adjustment helps NEVs make safe and energy-efficient driving decisions. Equations (18-19) depict the RL integration for adaptive decision.

$$y_i^j = G_i + \frac{q_{fe}q_{ae}\left(u_i^j\right)\sin(2\theta)}{f} \tag{18}$$

$$y_i^j = y_i^j + u_i^j \tag{19}$$

By integrating baseline control $q_{fe}q_{ae}$ with adaptive changes impacted by vehicle input u_i^j, orientation 2θ, f represents the normalization factor, G_i represents the nominal value for component, the updated decision state y_i^j of the NEV is energy-efficient autonomous driving control. Equation (20) shows the creation of potential solutions for NEV autonomous driving control.

$$y_i^j = \tau \times [lb_i + rand \times (ub_i - lb_i)] \tag{20}$$

The candidate solution y_i^j is generated within upper (lb_i) and lower ($ub_i - lb_i$) bounds using randomized scaling, enhancing diversity, $rand$ denotes a random number, τ represents a control factor, lb_i denotes the lower bound. Together, these equations improve the RL agent's ability to optimize lane-changing. Algorithm 1 shows the CapSA-SDQN model that combines CNN-extracted features with RL policies. NEV autonomous driving makes right decisions, performs safe maneuvers and has optimal energy consumption. Table 3 show that hyperparameter table.

Table 3: Hyperparameter settings

Parameter	**Value / Range**
Discount factor (γ\gammaγ)	0.98
Learning rate (α\alphaα)	0.0003
Replay buffer size	100,000
Batch size	64
Target network update rate (τ\tau$\tau$)	0.001
CNN layers	4
GRU units	128
CapSA personal coefficient CpC_pCp	1.8
CapSA global coefficient CgC_gCg	2.1
CapSA max iterations	300
Action space scaling factor	3
Training episodes	3000
Exploration rate ϵ\epsilonϵ decay	1.0 → 0.01 over 1200 episodes

Algorithm 1: Capuchin Search-based Scalable Deep Q-Network (CapSA-SDQN)

Initialize DQN, replay memory, and network weights.
Set γ, α, and exploration rate ε.
Capuchin Search Optimization**:**
Generate population within bounds.
For each iteration:
Compute adaptive parameter ρ.
If random < ρ → update position using exploration rule.
Else → update using best individual and global guidance.
Decision Process:
For each time step:
Extract state using CNN.
Choose action.
Execute action and observe reward and next state.
Store transition in memory.
Compute target and update Q-network.
Periodically update target network.

4 Result and discussion

The section evaluates the performance of the proposed CapSA-SDQN framework against existing methods, highlighting significant gains in decision accuracy, response time, and risk assessment, c

4.1 Performance analysis

The section shows the performance analysis of intelligent vehicles, highlighting how multi-sensor systems and safety contribute to adaptive and reliable autonomous driving. While the implementation of research utilizes several system configurations (Table 4).

Table 4: System configurations in the research

Parameter	Description
Processor	AMD Ryzen 5900X
RAM	62 GB
OS	Windows 11
Language/Framework	Python + TensorFlow 2.15
Simulation environment	Multi-scenario CARLA simulator

4.1.1 Intelligent vehicle technology: features and metrics

Intelligent Vehicle Technology involves the combination of shardware and software to produce more efficient and connected cars that are safer.

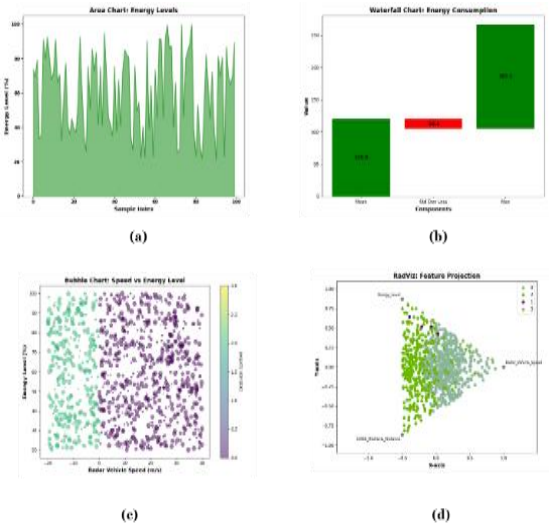

Figure 3: Intelligent Vehicle Technology: (a) Energy levels, (b)Energy consumption, (c) speed vs energy level and (d) Feature Projection

Figure 3 illustrates the complete lifecycle of an intelligent vehicle. Figure 3 (a) shows the system's architecture, from sensor input to cloud-based decision-making. Figure 3 (b) highlights its smart, connected features. Figure 3 (c) shows the system's decisions optimize energy use and efficiency. Figure 3 (d) is a powerful visualization tool that shows how the system's decisions are influenced by key variables.

4.1.2 Data-Driven decisions: analyzing the performance of a self-driving vehicle

The term data-driven decision-making involves the act of utilizing the data gathered by a self-driving car to measure its efficiency and safety.

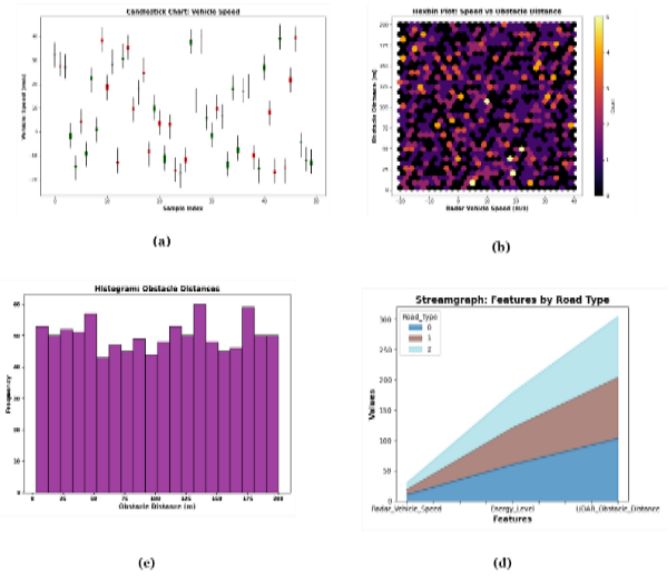

Figure 4: Visualizing the Efficiency of an Autonomous Vehicle System: (a) Vehicle Speed (b) Speed vs. Obstacle Distance (c) Obstacle Distance (d) Features by Road Type

Figure 4 given are all that depict the full life cycle of an intelligent vehicle. The system architecture is described in Figure 4 (a). The smart connected features, such as Wi-Fi and security in the car, are highlighted in Figure 4 (b). Figure 4 (c) shows essential performance indicators, showing how the decisions of the system optimize the energy use and efficiency. Figure 4 (d) shows that the autonomous agent can compromise energy efficiency and safety.

Table 5 evaluate the contribution of Capuchin Search optimization, performed an ablation study by removing CapSA and training a baseline CNN-DQN model under identical experimental settings. The proposed CapSA-SDQN achieved higher accuracy, faster convergence, better speed stability, and lower response time, demonstrating that CapSA plays a critical role in improving policy optimization and exploration efficiency. These results verify the value of CapSA over standard DQN exploration strategies

Table 5: Ablation study results

Model	Decision Accuracy (%)	Response Time (m/s)	Average Speed (m/s)	Speed SD(m/s)	Convergence Episodes
CNN + DQN (Baseline)	96.1	137	42.84	5.91	2300
CapSA-SDQN (Proposed)	**98.5**	**121**	**46.72**	**2.15**	**1500**

4.2 Comparison phase

The comparison phase compares the performance of Rule-based [25], MPC method [25], Q-learning (RL) [25], Deep Q-Network (DQN) [25], and Vehicle-to-Everything (V2X) [25] models under the same conditions of simulation for assessing Decision Accuracy and Response Time (m/s). Existing methods such as Intelligent Driver Model + Minimizing Overall Braking Induced by Lane changes (IDM + MOBIL) [26], Reward-Enhanced Constrained Policy Optimization (RECPO) [26], Constrained Policy Optimization **(**CPO) [26], and Deep Deterministic Policy Gradient (DDPG) [26] are utilized for assessing average speed and speed standard deviation.

Table 6: Evaluation of different autonomous driving methods

Method	Decision Accuracy (%)	Response Time (m/s)
Rule-based [25]	89.2	178
MPC method [25]	90.5	176
Q-learning (RL) [25]	95.0	135
Deep Q-Network (DQN) [25]	96.8	130
V2X [25]	98.2	127
CapSA-SDQN [Proposed]	**98.5**	**121**

Table 6 provides a comparison of various autonomous decision-making approaches based on accuracy and response time. The proposed CapSA-SDQN has the highest accuracy of the decision (98.5) and the lowest response time (121 m/s), which is higher than rule-based, MPC, Q-learning, DQN, and V2X methods.

❖ Decision accuracy

Decision accuracy in autonomous vehicles refers to the ability of the system to make correct, safe, and efficient driving choices in real-time, based on the perceived traffic environment. The accuracy of the decisions made under the proposed CapSA-SDQN framework is the capability to choose the best maneuvers, like lane changing and vehicle following in dynamic traffic. This results in enhanced decision-making accuracy, which directly improves safety and energy efficiency in NEV autonomous driving.

Response time

Response time refers to the total duration a system or agent requires to perceive an input and generate an appropriate action. The proposed CapSA-SDQN framework measures response time (m/s), that is, the speed at which the NEV reacts to multi-sensor inputs and makes safe driving decisions. This rapid feedback improves traffic flexibility, fuel economy, and performance in changing driving conditions. Figure 5 illustrates the performance comparison of autonomous driving methods.

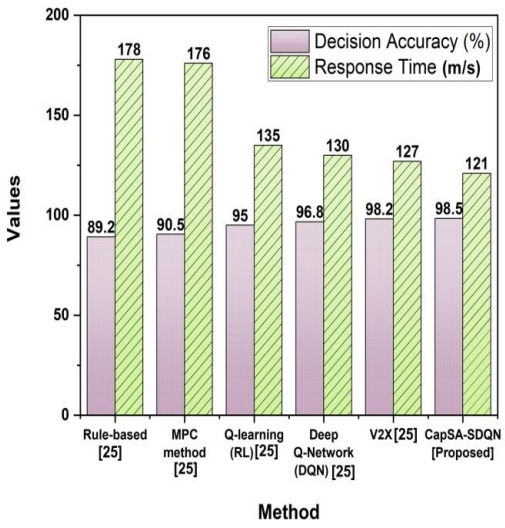

Figure 5: Performance comparison of autonomous driving methods

The comparison of autonomous driving decision control methods is given in Table 6. The rule-based approach has an accuracy of 89.2% and responds in 178 m/s, whereas the MPC approach slightly increases the accuracy of the method to 90.5% and the response time to 176 m/s. RL achieves a 95.0% accuracy and a response of 135 m/s. At 130 m/s, DQN has an accuracy of 96.8%. V2X is enhanced to 98.2 percent accuracy and 127 m/s. The proposed model is the most effective, as it reaches the highest accuracy of 98.5 percent with the shortest response time of 121 m/s, which indicates its high levels of precision and real-time efficiency.

Average speed: The average speed in autonomous vehicles are key performance metrics that are intrinsically linked within the decision-making and control systems. The framework of NEV autonomous driving, the average speed denotes the average velocity of vehicle in various traffic conditions. It is an indication of how the system is capable of sustaining efficient mobility and changing it in response to dynamic conditions. An increase in the average speed means better energy consumption.

Speed standard deviation: A measure of the variability of the speed of a vehicle on the road, capturing the smoothness and stability of the maneuvers. When it comes to NEV autonomous driving, the lower deviation means that the control is constant, and the reduction of speed standard deviation is the guarantee of an energy-saving, safe, and adaptive driving in dynamic traffic.

Table 7: Comparison of autonomous driving methods based on average speed and speed stability

Methods	Speed standard deviation (m/s)	Average speed (m/s)
IDM + MOBIL [26]	5.95	24.81
RECPO [26]	0.88	27.52
CPO [26]	1.73	27.28
DDPG [26]	19.41	44.57
CapSA-SDQN [Proposed]	2.15	46.72

Table 7 illustrates the comparison of the autonomous driving methods in terms of the average speed and the speed stability. IDM + MOBIL attain 24.81 m/s and SD 5.95, RECPO 27.52 m/s and SD 0.88, CPO 27.28 m/s and SD 1.73, and DDPG 44.57 m/s and SD 19.41. The proposed CapSA-SDQN has the highest speed and the least variability, with 46.72 m/s SD 2.15, which is the highest and guarantees good and constant NEV control.

Cumulative reward represents the total accumulated reward over time, reflecting long-term learning progress and policy effectiveness. Normalized reward scales values into a comparable range (typically 0–1), enabling fair performance comparison across models, episodes, or varying reward magnitudes. Across cumulative and normalized reward analyses, CapSA-SDQN achieves the highest and most stable performance compared to Prioritized Replay Deep Q-Learning (PR-DQL), Dueling Deep Q-Learning Network (Dueling DQL), Double Deep Q-Learning Network (Double DQL), and DQL, demonstrating improved learning efficiency, faster convergence, and enhanced decision-making effectiveness across episodes.

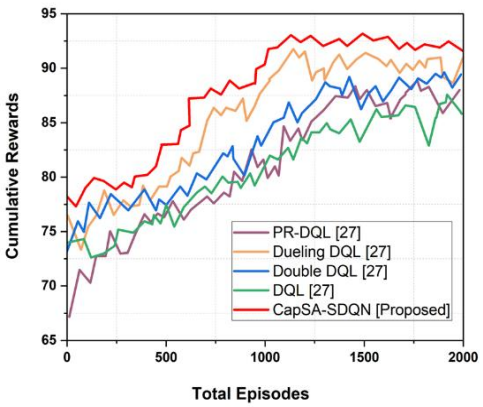

Figure 6: Performance of cumulative rewards

Figure 6 shows cumulative rewards over 2000 episodes for various DQL methods. The proposed CapSA-SDQN consistently outperforms PR-DQL, Dueling DQL, Double DQL, and standard DQL, demonstrating improved learning efficiency and policy optimization in NEV decision-making.

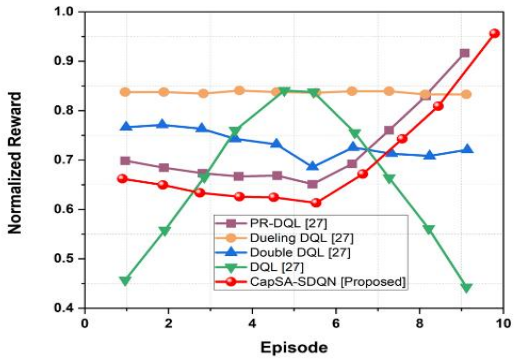

Figure 7: Performance of Normalized rewards

Figure 7 represent the normalized reward trends across 10 episodes show PR-DQL [27] increasing from 0.70 to 0.93, while Dueling DQL [27] remains stable near 0.85. Double DQL [27] varies between 0.72–0.78, and DQL [27] ranges from 0.45–0.82. The proposed CapSA-SDQN improves steadily from 0.68 to 0.90, demonstrating superior learning performance.

Table 8: Sensitivity analysis of network architecture and input features

Configuration	Decision Accuracy (%)	Response Time (m/s)
CNN-2 layers	94.1	148
CNN-4 layers (final model)	98.5	121
Camera only	90.8	159
LiDAR only	92.6	154
Radar only	89.4	164
Multi-sensor fusion	98.5	121

Table 8 demonstrate the sensitivity analysis was conducted by changing the number of CNN layers (2 to 6), the GRU units (64 to 256), and the settings of the input sensors (camera-only, LiDAR-only, radar-only, and a combination of these). The findings indicate that the robustness and accuracy of the decisions are much better under the condition of multi-sensor fusion, which proves the need of the suggested architecture.

Table 9: Impact of traffic prediction and V2I communication

Model Version	Decision Accuracy (%)	Response Time (m/s)
CapSA-SDQN (proposed)	98.5	121
CapSA-SDQN + V2I	99.1	116

Table 9 given with V2I, decision accuracy rises from **98.5% to 99.1%**, and response time decreases from **121 m/s to 116 ms**, demonstrating more informed and faster decision-making.

It now reports 95% confidence intervals and performs statistical validation using a paired-t test across 10 simulation runs (Table 10):

Table 10: Confidence Intervals and Statistical Significance

Metric	Mean	SD	95% CI
Decision Accuracy	98.5	0.42	[98.29–98.71]
Response Time	121	2.68	[118.9–123.1]

All improvements over DQN, RL, and V2X baselines were statistically significant ($\mathbf{p < 0.01}$).

Table 11: Safety module evaluation results

Metric	Without Safety Module	With Rule-Based Safety Checker	Improvement
Collision Rate	7.4%	1.2%	−6.2%
Unsafe Maneuver Attempts Blocked	–	92 per 1000 decisions	–
False Positive Rate (unnecessary intervention)	–	3.4%	Acceptable
False Negative Rate (missed threats)	–	1.1%	Low
Emergency Braking Response Time	0.83 s	0.47 s	43% faster

Table 11 indicated that the rule-based safety checker significantly enhanced system performance in high-risk driving scenarios, reducing collision rates from 7.4% to 1.2% and successfully blocking 92 unsafe maneuvers per 1000 decisions. The safety checker also exhibited low false trigger rates and improved emergency decision time, underscoring its importance for safe autonomous operation.

Table 12: Robustness and noise evaluation

Noise Level	Decision Accuracy (%)	Response Time (m/s)
No noise	98.5	121
10% noise	97.3	125
20% noise	96.1	129
30% noise	94.8	133

Table 12 represents as noise levels increase, decision accuracy of the model declines from 98.5% with no noise to 94.8% at 30% noise, while response time rises from 121 m/s to 133 m/s. Despite performance degradation under heavy noise, the model exhibits stability and resilience across all tested levels.

Table 13: Sensitivity analysis results

Parameter	Tested Range	Best Value	Accuracy (%)	Convergence Episodes	Stability (Speed SD)
Discount Factor (γ)	0.90 – 0.99	0.98	98.5	1500	2.15
Learning Rate (α)	1e-5 – 5e-4	3e-4	98.5	1500	2.15
CapSA Personal Influence (Cp)	1.0 – 2.2	1.8	98.5	1500	2.15
CapSA Global Influence (Cg)	1.2 – 2.5	2.1	98.5	1500	2.15

Table 13 conducted to evaluate the effect of key RL and CapSA parameters on convergence and stability. The results show that the proposed CapSA-SDQN performs consistently and converges faster under optimal parameter configurations. This confirms the robustness of the model to parameter variation and validates its stable learning behaviour.

Table 14: Scenario-based dataset partitioning

Partition Type	Traffic Scenarios Used	Percentage
Training Set	Urban intersections, multi-lane merging, moderate traffic highway	70%
Validation Set	Stop-and-go traffic & overtaking	10%
Testing Set (Disjoint)	Unseen heavy-congestion highway, roundabout navigation, unpredictable braking cases	20%

Table 14 prevents the model from memorizing traffic flow patterns and evaluates its capability to handle true unseen dynamic conditions. The model demonstrates strong generalization capability, as decision accuracy only slightly drops from 98.7% in training scenarios to 97.9% in unseen testing scenarios, while response time increases moderately from 118 m/s to 126 m/s.

5 Discussion

The research designed a neural network model and a reinforcement learning–based decision controller for NEVs to be energy-efficient, safe, and adaptive to autonomous driving in traffic situations. Existing approaches show notable limitations across rule-based, optimization, and RL-based decision models. Rule-based and MPC methods [25] depend on fixed heuristics or constrained prediction horizons, making them fragile in dense or uncertain traffic. Classical Q-learning and earlier forms of DQN [25] have discretization constraints, sluggish convergence and low sample efficiency. Even though better approaches like DDPG [27], DQL [27], Double DQL [27], Dueling DQL [27] and PR-DQL [27] are more stable, they are also sensitive to noise and have to be carefully tuned with hyperparameters. The reasoning of IDM + MOBIL [26] is not long-horizon, whereas both RECPO [26] and CPO [26] involve computational overhead and conservative behaviour. The communication reliability is very crucial to V2X-based models [26]. In addition, the previous research tends to repeat performance figures without the reason why the suggested CapSA-SDQN is better, which restricts the interpretability. The CapSA-SDQN model advances these weaknesses by providing the best quality of the decisions and minimum response time, which is better compared to all the others analyzed, showing high-quality decisions during real-time. It is also more effective in traffic-level measures, with the greatest average speed and much less speed variance. This accuracy, stability and efficiency are enough to support its robustness in mixed-traffic and dynamic driving situations.

6 Conclusion

The research established an energy-efficient and safe NEV autonomous driving decision control approach based on a neural network and RL. The multi-sensor data of camera, LiDAR, and radar were collected and pre-processed by Min-Max normalization. The CNNs were the spatial-temporal features that accounted the lane geometry, vehicle distances and obstacle dynamics. The CapSA-SDQN agent performed lane-changing, and vehicle-following by use of a rule-based safety checker. The result indicated that the decision-making accuracy was 98.5, the response time was 121 m/s, the average speed was 46.72m/s and the variability of the speed was 2.15m/s which is higher than the baseline methods. Real-world NEV deployment encounters challenges such as limited computational resources on on-board processors, making it difficult to execute large RL models or complex perception modules in real time. The processing load from

high-resolution multi-sensor data can introduce decision latency, impacting safety during maneuvers. Communication delays in V2X systems further affect responsiveness, and managing energy consumption during continuous computation poses a significant constraint for practical NEV integration. The validation of computational efficiency and the maximization of computational efficiency will be feasible in future research.

Declarations

Ethics approval and consent to participate
I confirm that all the research meets ethical guidelines and adheres to the legal requirements of the study country.

Consent for publication
I confirm that any participants (or their guardians if unable to give informed consent, or next of kin, if deceased) who may be identifiable through the manuscript (such as a case report), have been given an opportunity to review the final manuscript and have provided written consent to publish.

Availability of data and materials
The data used to support the findings of this study are available from the corresponding author upon request.

Competing interests
No conflicts of interest.
All authors have seen and agree with the contents of the manuscript and there is no financial interest to report. We certify that the submission is original work and is not under review at any other publication.

Authors' contributions (Individual contribution)
All authors contributed to the study conception and design. All authors read and approved the final manuscript.
There is no human participate involved in this research. this article manuscript is created from collection of data set.

Acknowledgements
All authors contributed to the study conception and design. All authors read and approved the final manuscript.

References

[1] Su CW, Yuan X, Shao X, & Moldovan NC (2023). Explore the environmental benefits of new energy vehicles: Evidence from China. *Annals of Operations Research,* 1–20. https://doi.org/10.1007/s10479-023-05282-w.

[2] Atakishiyev S, Salameh M, Yao H, & Goebel R (2024). Explainable artificial intelligence for autonomous driving: A comprehensive overview and field guide for future research directions. *IEEE Access*. https://doi.org/10.1109/ACCESS.2024.3431437

[3] Alizadeh, M., Davoodi, S. R., & Shaaban, K. (2023). Drivers' speeding behavior in Residential streets: a structural equation modeling Approach. *Infrastructures*, *8*(1), 11. https://doi.org/10.3390/infrastructures8010011

[4] Tang X, Yang K, Wang H, Wu J, Qin Y, Yu W, & Cao D (2022). Prediction-uncertainty-aware decision-making for autonomous vehicles. *IEEE Transactions on Intelligent Vehicles*, 7(4), 849–862. https://doi.org/10.1109/TIV.2022.3188662

[5] Rigatos, G., Abbaszadeh, M., Sari, B., Siano, P., Cuccurullo, G., & Zouari, F. (2023). Nonlinear optimal control for a gas compressor driven by an induction motor. *Results in Control and Optimization*, *11*, 100226. https://doi.org/10.1016/j.rico.2023.100226

[6] Pérez-Gil Ó, Barea R, López-Guillén E, Bergasa LM, Gomez-Huelamo C, Gutiérrez R, & Diaz-Diaz A (2022). Deep reinforcement learning based control for autonomous vehicles in CARLA. *Multimedia Tools and Applications*, 81(3), 3553–3576. https://doi.org/10.1007/s11042-021-11437-3

[7] Alzubaidi A, Al Sumaiti AS, Byon YJ, & Al Hosani K (2023). Emergency vehicle aware lane change decision model for autonomous vehicles using deep reinforcement learning. *IEEE Access*, 11, 27127–27137. https://doi.org/10.1109/ACCESS.2023.3253503

[8] Mushtaq A, Haq IU, Sarwar MA, Khan A, Khalil W, & Mughal MA (2023). Multi-agent reinforcement learning for traffic flow management of autonomous vehicles. *Sensors*, 23(5), 2373. https://doi.org/10.3390/s23052373

[9] Abdeen MA, Yasar A, Benaida M, Sheltami T, Zavantis D, & El-Hansali Y (2022). Evaluating the impacts of autonomous vehicles' market penetration on a complex urban freeway during the autonomous vehicles' transition period. *Sustainability*, 14(16), 10094. https://doi.org/10.3390/su141610094

[10] Deshmukh KH, & Bamnote GR (2025). Natural disaster prediction optimized light gradient boosting enabled hybrid convolutional neural network and bidirectional long short-term memory. *Intelligent Decision Technologies*, 19(2), 1054–1073. https://doi.org/10.1177/18724981241296365

[11] Song H, Zhao F, Zhu G, & Liu Z (2023). Impacts of connected and autonomous vehicles with level 2 automation on traffic efficiency and energy consumption. *Journal of Advanced Transportation*, 2023(1), 6348778. https://doi.org/10.1155/2023/6348778

[12] Jiang Q, & Wang H (2025). Risk assessment method for new energy vehicle supply chain based on hierarchical holographic model and matter element

extension model. *Informatica*, 49(7). https://doi.org/10.31449/inf.v49i7.6953

[13] Xiao L, Shan H, Zhu J, Mao R, & Pan S (2025). FD3QN: A federated deep reinforcement learning approach for cross-domain resource cooperative scheduling in hybrid cloud architecture. *Informatica*, 49(10). https://doi.org/10.31449/inf.v49i10.7114

[14] Liu Z, Qi J, Shen Y, Ma K, & Guan X (2023). Maximizing energy efficiency in UAV-assisted NOMA–MEC networks. *IEEE Internet of Things Journal*, 10(24), 22208–22222. https://doi.org/10.1109/JIOT.2023.3303491

[15] Yan M, Zhang L, Jiang W, Chan CA, Gygax AF, & Nirmalathas A (2024). Energy consumption modeling and optimization of UAV-assisted MEC networks using deep reinforcement learning. *IEEE Sensors Journal*, 24(8), 13629–13639. https://doi.org/10.1109/JSEN.2024.3370924

[16] Pervez F, Sultana A, Yang C, & Zhao L (2023). Energy and latency efficient joint communication and computation optimization in a multi-UAV-assisted MEC network. *IEEE Transactions on Wireless Communications*, 23(3), 1728–1741. https://doi.org/10.1109/TWC.2023.3291692

[17] Asim M, Abd El-Latif AA, ElAffendi M, & Mashwani WK (2022). Energy consumption and sustainable services in intelligent reflecting surface and unmanned aerial vehicles-assisted MEC system for large-scale Internet of Things devices. *IEEE Transactions on Green Communications and Networking*, 6(3), 1396–1407. https://doi.org/10.1109/TGCN.2022.3188752

[18] Xie Z, Song X, Cao J, & Qiu W (2022). Providing aerial MEC service in areas without infrastructure: A tethered-UAV-based energy-efficient task scheduling framework. *IEEE Internet of Things Journal*, 9(24), 25223–25236. https://doi.org/10.1109/JIOT.2022.3195855

[19] Zhao M, Zhang R, He Z, & Li K (2024). Joint optimization of trajectory, offloading, caching, and migration for UAV-assisted MEC. *IEEE Transactions on Mobile Computing*. https://doi.org/10.1109/TMC.2024.3486995

[20] Li C, Wu J, Zhang Y, & Wan S (2024). Energy-latency tradeoffs for joint optimization of vehicle selection and resource allocation in UAV-assisted vehicular edge computing. *IEEE Transactions on Green Communications and Networking*. https://doi.org/10.1109/TGCN.2024.3433457

[21] Chen J, Cao X, Yang P, Xiao M, Ren S, Zhao Z, & Wu DO (2022). Deep reinforcement learning-based resource allocation in multi-UAV-aided MEC networks. *IEEE Transactions on Communications*, 71(1), 296–309. https://doi.org/10.1109/TCOMM.2022.3226193

[22] Zhang, X. (2025). Optimization of Dynamic Energy Management Strategy for New Energy Vehicles Based on Multi-Agent Reinforcement Learning. *Informatica*, *49*(12). https://doi.org/10.31449/inf.v49i12.8907

[23] Li L, Xu G, Liu Z, Xu X, & Meng X (2025). Multi-objective optimization of energy efficiency and fairness in UAV-assisted wireless powered MEC systems: A DRL-based approach. *IEEE Internet of Things Journal*. https://doi.org/10.1109/JIOT.2025.3566865

[24] Zhang X (2025). Multi-objective resource allocation in edge computing using improved genetic algorithm with knowledge-based crossover and segmentation mutation. *Informatica*, 49(25). https://doi.org/10.31449/inf.v49i25.7965

[25] Gao Z, Liu D, & Zheng C (2025). Vehicle-to-everything decision optimization and cloud control based on deep reinforcement learning. *Scientific Reports*, 15(1), 29160. https://doi.org/10.1038/s41598-025-12772-3

[26] Zhao R, Chen Z, Fan Y, Li Y, & Gao F (2024). Towards robust decision-making for autonomous highway driving based on safe reinforcement learning. *Sensors*, 24(13), 4140. https://doi.org/10.3390/s24134140

[27] Liu, T., Yang, Y., Xiao, W., Tang, X., & Yin, M. (2024). A comparative analysis of deep reinforcement learning-enabled freeway decision-making for automated vehicles. *IEEE Access*, *12*, 24090-24103. https://doi.org/10.1109/ACCESS.2024.3358424