Available online at www.ijournalse.org

Emerging Science Journal

(ISSN: 2610-9182)

Vol. 10, No. 3, June, 2026

Dual-Agent Q-Learning for Cross-Layer IEEE 802.11bd Optimization in Dense VANETs

Galih Nugraha Nurkahfi [1, 2], Suyoto [2], Agus Subekti [2], Budi Prawara [2], Ratna Mayasari [1, 3*], Andy Triwinarko [4], Nasrullah Armi [2, 5], Eueung Mulyana [1], Nana Rachmana Syambas [1]

[1] *School of Electrical Engineering and Informatics, Bandung Institute of Technology (ITB), Bandung 40132, Indonesia.*

[2] *Research Organization of Electronics and Informatics, National Research and Innovation Agency (BRIN), Bandung 40135, Indonesia.*

[3] *Center of Excellence for Intelligent Sensing-IoT, Research Institute of Sustainable Society, Telkom University, Bandung 40257, Indonesia.*

[4] *Network and Multimedia Study Program, State Polytechnique of Batam (Polibatam), Batam 29431, Indonesia.*

[5] *Center of Excellence for Advanced Intelligent Communications (AICOMS), Telkom University, Bandung 40257, Indonesia.*

Abstract

Dense vehicular ad hoc networks face critical challenges in reliably delivering safety messages due to channel congestion, packet collisions, and interference. This study develops a dual-agent Q-learning framework for cross-layer IEEE 802.11bd optimization to improve latency and power efficiency while maintaining acceptable packet delivery ratios in dense traffic. We propose a decomposed architecture separating PHY-layer power control and MAC-layer beacon rate adaptation, with deterministic SINR-based MCS selection ensuring IEEE 802.11bd compliance. The framework is evaluated using a Python-based VANET simulator implementing the IEEE 802.11bd PHY/MAC stack with realistic SUMO mobility, multi-class background traffic, and omnidirectional/sectoral antennas across 20-90 vehicles/km densities. Results show dual-agent Q-learning reduces average latency by 44.6% (31.1ms to 17.2ms) and transmission power by 55% (15-20dBm to 9dBm) compared to static baselines, with acceptable 5-11% PDR reduction (94.2% to 88.6%). The approach converges within 8,500 episodes, significantly faster than single-agent Q-learning (12,500) and dual-agent DQN (14,000-35,000). This work introduces the first dual-agent tabular Q-learning for joint power-rate-MCS optimization in IEEE 802.11bd VANETs, demonstrating that agent decomposition reduces state-action complexity while enabling interpretable, fast-converging control suitable for sub-100ms vehicular applications.

Keywords:

Vehicle-to-Vehicle Communication; IEEE 802.11bd; Reinforcement Learning; Cross-Layer Optimization; Congestion Control; Power Efficiency; Connected Vehicles; Dense Traffic; Sectoral Antenna; Omnidirectional Antenna.

Article History:

Received:	09	December	2025
Revised:	09	May	2026
Accepted:	13	May	2026
Published:	01	June	2026

1- Introduction

Traffic accidents remain a critical global challenge, where the WHO Global Status Report on Road Safety 2023 notes 1.19 million annual fatalities from road traffic crashes, representing more than 3,200 deaths per day worldwide [1]. Vehicle-to-Everything (V2X) communication is emerging as a technology to address these challenges by enabling real-time information exchange among vehicles, infrastructure, digital infrastructure, and other road users [2].

V2X systems extend vehicle perception capabilities beyond the physical limitations of onboard sensors through collaborative networking, creating a distributed safety ecosystem that improves situational awareness and allows proactive hazard mitigation [3] as illustrated in Figure 1.

* **CONTACT**: ratnamayasari@telkomuniversity.ac.id

DOI: https://doi.org/10.28991/ESJ-2026-010-03-019

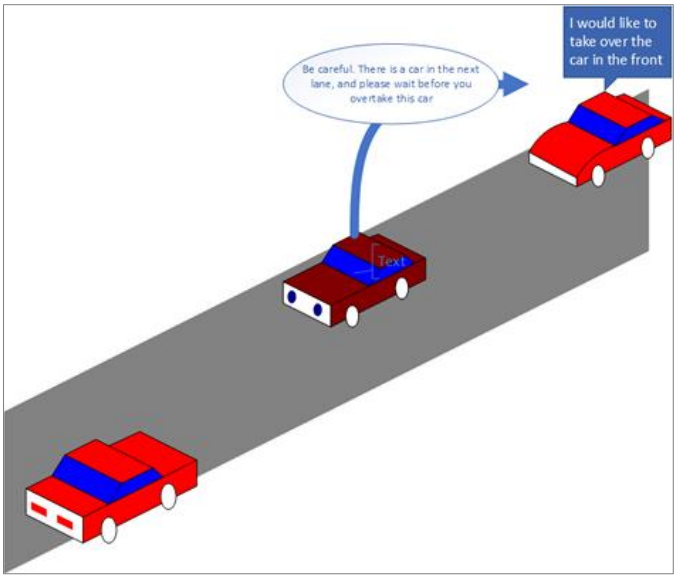

Figure 1. Safety Communication Between Cars

Vehicle-to-vehicle (V2V) communication, as a critical component of Vehicle-to-Everything (V2X), is based on the continuous broadcast of safety messages following standards such as SAE J2735 Basic Safety Message (BSM), which includes essential kinematic data, including position, velocity, acceleration, and heading information [4].

Current vehicular communication utilizes two primary standards: IEEE protocols (including IEEE 802.11p for DSRC and the upcoming IEEE 802.11bd) and 3GPP Cellular V2X (C-V2X). In contrast, European deployments use the C-ITS framework based on ITS-G5 with CSMA/CA channel access [5-8]. IEEE 802.11bd introduces enhancements, such as advanced LDPC encoding, midamble-based channel estimation, and 256-QAM modulation, which provide up to 2x improvements in minimum throughput and enhanced reliability [6]. IEEE 802.11bd also introduces significant enhancements over IEEE 802.11p, including midambles for channel tracking in high-mobility scenarios, LDPC coding, higher-order modulation (up to 256-QAM), MIMO support, and adaptive packet repetition, achieving data rates up to 180 Mbps compared to IEEE 802.11p's 27 Mbps [9].

However, dense vehicular scenarios still face serious challenges for reliable safety message delivery due to wireless channel congestion from simultaneous transmissions, causing increased packet collision rates, higher channel busy ratios (CBR), and poor packet delivery reliability that degrades network performance [10, 11]. The high-speed mobility and rapid topology changes make these problems worse, as CSMA/CA with the distributed coordination function (DCF) prevents simultaneous transmissions to avoid interference, leading to receiver blocking where vehicles cannot communicate at the same time. At the PHY layer, dense VANETs cause interference saturation from high transmission activity, and combined MAC and PHY layer limitations reduce network performance by making more vehicles wait to use fewer transmitters, even though proper parameter optimization could allow better coexistence [12].activity, and combining MAC and PHY layer limitations reduce network performance by making more vehicles wait to use fewer transmitters, even though proper parameter optimization could allow better coexistence.

Omnidirectional antennas operating in the 5.9 GHz DSRC band can transmit signals 360 degrees around, with typical gains of 2-10 dBi. However, they encounter problems with increased co-channel interference, limited spatial reuse capabilities, and a reduced effective communication range when many vehicles attempt to access the spectrum simultaneously in high-density deployments [13, 14]. Directional antenna systems solve these problems through enhanced spatial selectivity and interference mitigation. Research shows that directional antennas significantly improve multicast capacity in VANETs by optimizing beamwidth configuration to enable better spatial reuse of wireless channels [15], create substantial range improvements through enhanced signal-to-noise ratio performance in vehicle-to-vehicle communications [16], and enable sophisticated routing protocols and neighbor discovery mechanisms that improve overall network throughput [17]. However, comparative analyses that test the practical implementation trade-offs between sectoral and omnidirectional antenna configurations in various densities of VANET scenarios are still limited, specifically in terms of integration with optimization algorithms and real-world deployment considerations.

The complex and dynamic characteristics of VANET require an optimization approach that can intelligently respond to rapidly changing conditions. This approach proposes overcoming the limitations of static configurations [18], which create suboptimal communication performance, as well as the limitations of previous single-agent systems such as [18-22]. This paper proposes a dual-agent Q-learning framework that jointly optimizes transmission power, Modulation and Coding Scheme selection, and beacon rate adaptation in IEEE 802.11bd-compliant vehicular networks, as illustrated in Figure 2.

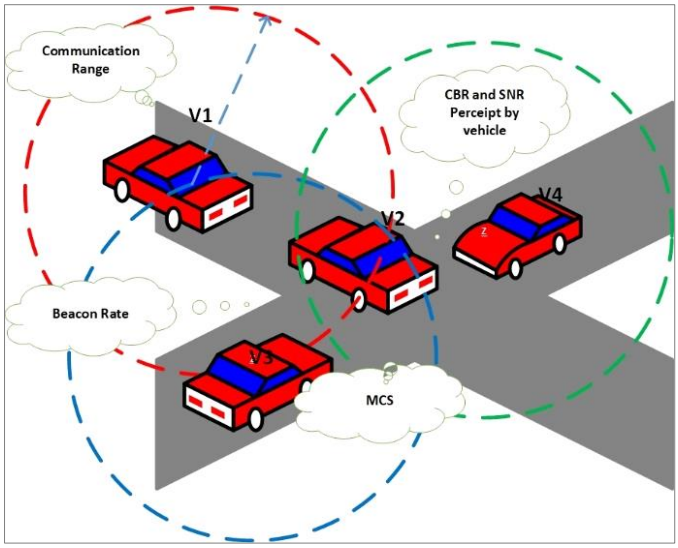

Figure 2. Dense-Vanet Performance Optimization

Previous studies have applied deep reinforcement learning, such as DQN [23], PPO [24-27], and SAC [27] demonstrating adaptability in high-dimensional state spaces. Multi-agent approaches such as federated MADDPG [28] multi-agent DDPG with attention, and multi-agent Q-learning with LSTM [29]. However, these approaches require complex neural network architectures, extensive training, and often lack convergence guarantees, which limit their practicality for safety-critical vehicular communication [30], besides limiting practicality for commercial OBUs with 1-2 GB RAM [31, 32]. Furthermore, existing multi-agent strategies use per-vehicle decomposition, causing state-action space explosion, rather than functional layer-based decomposition. In contrast, our framework uses tabular Q-learning [33], which is decomposed into two agents for PHY and MAC control. This design provides faster inference through simple table lookups, avoids the state–action space explosion of single-agent control, and offers interpretability essential for safety validation. This design provides faster inference via the table lookups, reduces the state-action space from 710 million to 6.5 million, requires only 25 MB of memory for OBU deployment, and offers interpretability for safety validation. By aligning with IEEE 802.11bd constraints [9], our approach achieves a practical balance between performance and deployability in real-time VANET scenarios.

Main Contributions: A Q-learning framework with separate agents for PHY-layer optimization (power and MCS adaptation targeting SINR performance) and MAC-layer beacon rate control (Channel Busy Ratio management). This proposed framework enables specific optimization for the PHY and MAC layers while maintaining coordinated cross-layer parameter adaptation in each vehicle. This allows for independent adaptation to sectoral versus omnidirectional antenna characteristics across varying vehicle densities, without interference from other optimizations. The research conducts an optimization experiment and performance evaluation comparing sectoral and omnidirectional antenna configurations across multiple vehicle densities, from sparse (20 vehicles) to dense (90 vehicles). And then measures performance characteristics under varying configurations and provides deployment guidelines for practical implementations. In addition, we design five reward configurations (throughput, conservative, reliability, balanced, and density-adaptive) to investigate practical deployment trade-offs across application scenarios.

Implementation: a Python-based VANET simulator that integrates a UBX-V2X-based [34] IEEE 802.11bd stack is developed in this research. This simulator includes channel models, antenna pattern implementations, RL-based optimization, and vehicular mobility characteristics, enabling the control of experimental variables. For the Optimization experiment, in addition to the dual-agent Q-learning algorithm, we also developed and tested the optimization using conventional single-agent Q-learning. Each experiment uses a zero-gain configuration to isolate the algorithm's effectiveness from the specific benefits of each antenna gain configuration, ensuring an objective evaluation of the sectoral and omnidirectional antenna systems' characteristics for various vehicle densities and channel conditions. The proposed framework operates in a non-cooperative and distributed manner, where each vehicle independently adapts its communication parameters based on local CBR (MAC layer agent target parameter) and SINR (PHY layer agent target parameter) observations, without centralized coordination. This approach enables scalable deployment with low-latency requirements for real-time optimization applications.

2- Related Works

2-1- Traditional Parameter Optimization Methods

Vehicular networks use periodic broadcast messages, Cooperative Awareness Messages (CAM) in Europe and Basic Safety Messages (BSM) in the United States, to share position, speed, and acceleration data as part of the Cooperative Awareness Service (CAS) defined by ETSI [35-38]. These safety messages need proper parameter configuration: higher

data rates reduce packet duration but need stronger SINR, while lower rates increase transmission time and channel occupancy [21]. Previous research used conventional algorithms with decentralized distributed transmission power control [22], reliability scoring [39], and centralized solutions [19]. However, research also shows that static parameter configurations often fail to meet performance requirements standards under dynamic conditions [18, 40].

2-2-Advanced Optimization Methods Evolution

The limitations of the conventional framework have led to the development of more effective methods, including fuzzy logic frameworks, such as FLRM [41], and multi-parameter optimization [42]. Additionally, game theory has been explored in several research studies, including those that proposed the BFPC protocol [43] and coalition formation [44]. Machine learning techniques, particularly reinforcement learning, emerged as potentially better solutions for dynamic adaptation. While conventional methods, such as CACC [45] and JATB [46], provide rule-based solutions, hybrid approaches, like the context-aware ECPR system [47], propose comprehensive solutions with PHY/MAC cross-layer design [48]. Q-learning frameworks, such as MDPRP [11], have demonstrated the effectiveness of jointly optimizing beacon rate and transmission power. Meanwhile, Deep Q-Networks have demonstrated the capability to control transmission power and frequency spectrum [49]. Furthermore, advanced policy methods using PPO and SAC have achieved simultaneous adaptation of data rate and transmission power [27].

Recent deep RL approaches for VANET resource allocation, such as AAFSA-DDQNet [30], achieve high performance (92.8% PDR, 12.5 ms latency) but require GPU acceleration with 24 GB VRAM and 5.2 GFLOPs computational overhead. Our dual-agent tabular Q-learning achieves practical OBU deployability with only 25 MB memory by separating PHY and MAC optimization into independent agents, trading neural network approximation for exact state-action mappings suitable for resource-constrained vehicular hardware.

Meanwhile, recent advances in edge intelligence and digital twin networks (DTN) have enabled proactive resource management in vehicular networks. Elloumi et al. [29] proposed PRISM, a DTN-empowered framework that proactively predicts vehicle positions using LSTM with transfer learning and multi-path channel gains using ray tracing, achieving up to 33% capacity improvement over non-proactive schemes. Their multi-agent Q-learning approach for joint vehicle clustering and power allocation requires iterative training (65-70 seconds for 30 vehicles) and maintains separate Q-tables for each vehicle, with storage requirements scaling linearly with vehicle count. Our dual-agent tabular Q-learning approach achieves practical deployability by decomposing the problem into PHY and MAC agents rather than per-vehicle agents, requiring only 25 MB memory total and enabling sub-100ms control loops suitable for real-time OBU execution without GPU acceleration or cloud connectivity.

Latest research has explored various optimization approaches. Conventional methods use static parameters such as fixed power levels and beacon rates [18], which offer limited adaptability to dynamic traffic conditions. Single-parameter approaches focus on power control [50-52] or rate control [53-55] separately, missing potential benefits from cross-layer interactions. Deep RL methods like DQN [23] and PPO [24, 27] show promise but require substantial computational resources and long training times, unsuitable for real-time vehicular control [56-58]. Recent hybrid architectures have attempted to balance computational efficiency with performance. Kai & Liang [59] proposed a CNN-LSTM framework with reinforcement learning for emergency maneuver control in VANETs, demonstrating that hybrid deep learning approaches can achieve collision probability reduction (29%) and low decision latency (<225ms) while maintaining real-time responsiveness. However, their approach still relies on deep neural networks for spatial-temporal feature extraction, which introduces greater training complexity and computational overhead than tabular methods.

Recent work has used federated multi-agent reinforcement learning for V2X resource allocation. Liu & Ma [28] proposed AFL-MADDPG, which combines asynchronous federated learning with multi-agent deep reinforcement learning to optimize spectrum, power, and bandwidth allocation in V2I/V2V networks, achieving 19.1% spectral efficiency improvement. However, their approach has limitations for IEEE 802.11bd: (1) it focuses on coarse-grained resource block selection rather than fine-grained PHY-MAC parameters (MCS, beacon rate), (2) it requires federated coordination between vehicles, and (3) it uses deep neural networks with millions of parameters, making it computationally expensive compared to tabular methods.

Recent multi-agent deep RL approaches have demonstrated effectiveness in VANET resource allocation. Liu & Deng [60] proposed AMADRL, combining multi-agent DDPG with attention mechanisms to jointly optimize spectrum and power allocation, achieving superior convergence and meeting heterogeneous V2V/V2I QoS requirements. However, such approaches require substantial computational resources (1.03s average computation time) and complex network architectures with multiple critic networks (1024-512-256 neurons). Our dual-agent tabular Q-learning achieves practical deployability with only 25 MB of memory on commercial OBUs by decomposing the problem into PHY and MAC agents, avoiding the neural network overhead required by deep MADRL methods.

Several gaps remain unaddressed properly. First, in the past, most studies optimized single parameters rather than joint cross-layer control [54, 55, 61-63]. Second, IEEE 802.11bd-specific features, like doubled MCS levels and

improved channel coding, need specialized approaches [64-66]. Third, existing multi-parameter methods use monolithic agents suffering from state-action space explosion and conflicting objectives [14]. Fourth, the integration of antenna configuration with RL optimization remains underexplored [16, 17].

Unlike federated multi-agent DRL systems that use deep neural networks for high-level resource block coordination [28] or deep learning-based VANET control systems that use monolithic CNN-LSTM architectures for decision-making [59], our dual-agent approach solves these problems by splitting PHY and MAC control into separate agents. This reduces the state-action space from 710 million to 6.5 million entries, maintains IEEE 802.11bd standard compliance, and provides fast inference suitable for real-time vehicular control. All these frameworks' evolution can be seen in Figure 3.

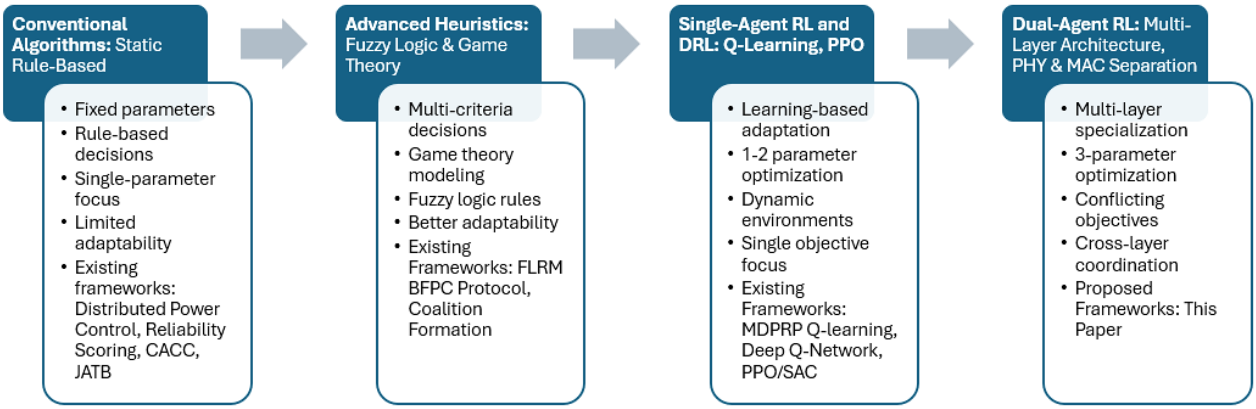

Figure 3. Frameworks Evolution

2-3-Algorithm Selection, Parameter Selection, and Optimization Scopes

Why Q-learning instead of other RL methods: Tabular Q-learning is chosen over deep RL methods like DQN, PPO, and SAC for several reasons: 1. Vehicular applications need fast responses (under 100ms), and looking up values in Q-tables may be faster than running neural networks. 2. Tabular Q-learning has stronger theoretical guarantees about finding reasonable solutions compared to deep RL methods. 3. The problem uses discrete values that are suitable for Q-tables. 4. Q-tables are easier to understand and debug, which is important for safety applications. While deep RL methods, such as PPO or SAC [27], have shown promising results in previous VANET work , DQN [30, 67], and multi-agent deep RL approaches [28, 60], they may introduce training complexity and potentially higher computational issues that could be problematic for real-time vehicle communication systems.

Single-parameter approaches focused on beaconing rate adjustment [27, 54], transmission power control [12, 51, 61, [68-71], queue and bandwidth allocation [72, 73] or data rate adjustments [55], but may not adequately consider parameter interdependencies. As shown in Table 1 RL-based Vanet Optimization in the previous works, existing RL-based approaches primarily optimized two parameters: beacon rate and transmission power, targeting CBR, PDR, and throughput, transmission power and frequency spectrum targeting latency [49], or data rate and transmission power targeting CBR and throughput [11].

Table 1. RL-based VANET Optimization in The Previous Works

Controlled Parameters	Optimized Parameter	Control	Methodologies	Cooperative/non-cooperative	RATs	Paper
Beaconing Rate & Transmission Power	CBR: PDR, Throughput	Decentralized	Q-learning	Non-Cooperative	IEEE 802.11p	Aznar-Poveda et al. [11]
Transmission Power & Frequency Spectrum	Latency	Decentralized	Deep Q-Network	Non-Cooperative	LTE-Vehicular	Ye et al. [49]
Data Rate & Transmission Power	CBR: Throughput	Centralized	PPO, SAC	Cooperative	IEEE 802.11p	Aznar-Poveda et al. [27]
Spectrum, Power & Bandwidth	Spectral Efficiency	Federated/Distributed	AFL-MADDPG	Semi-Cooperative	Cellular	Liu & Ma [28]
Spectrum & Power	QoS (V2V/V2I)	Distributed	Multi-agent DDPG with Attention	Cooperative	Cellular	Liu & Deng [60]
Vehicle Clustering & Power	Capacity, Channel Prediction	Centralized	Multi-agent Q-learning + LSTM	Cooperative	Cellular Beyond 5G	Elloumi et al. [29]
Resource Block Selection	PDR, Latency	Decentralized	AAFSA-DDQNet	Non-Cooperative	Cellular 5G NR	Cui [30]
MCS, Beacon Rate, Transmission Power	CBR, SNR: Latency, PDR	Centralized	Cross-Layer Multi-Agent Q-Learning	Semi-Cooperative	IEEE 802.11bd	Present Study

Parameter Selection Rationale: The choice of transmission power, MCS, and beacon rate for joint optimization is motivated by IEEE 802.11bd-specific characteristics. MCS selection becomes critical with the introduction of advanced modulation schemes, such as 256-QAM, by 802.11bd, which directly affects both data rate and error resilience [6]. Transmission power affects signal coverage and interference patterns, while the beacon rate controls the frequency of channel access.

Target Metrics Justification: CBR is selected as the MAC-layer target because it serves as the standard congestion metric in DSRC/C-ITS standards [36]. At the same time, SINR serves as the PHY-layer target because it directly reflects signal quality under interference conditions, particularly relevant for dense vehicular scenarios where multiple transmissions compete for spectrum access.

Gap Identified: Limited research addresses this three-parameter combination, potentially missing optimization opportunities in which MCS adaptation can simultaneously affect PHY-layer performance and MAC-layer channel utilization efficiency.

2-4- Antenna Configuration Integration

According to Eckhoff et al. [74] antenna configuration research reveals significant performance variations across selected patterns. Realistic antenna patterns can affect crash avoidance outcomes compared to ideal antennas. In contrast, directional antennas may enable increased spatial reuse and higher gains, potentially leading to better throughput in dense environments [14]. Studies have analysed multicast capacity under directional antenna configuration and measured real-world performance characteristics [16]. However, as shown in Table 2, few comparative studies examine practical trade-offs in the implementation of sectoral and omnidirectional antenna configurations when integrated with PHY/MAC parameter optimization algorithms for VANETs.

Table 2. Antenna Configuration Research in VANET

Antenna Configuration	Performance Metrics	Methodology	Key Findings	Environment/Scenario	Standard	Paper
Realistic vs. Isotropic	Collision Avoidance Success	Simulation	Antenna patterns affect crash outcomes	City-wide & Intersection	DSRC	Eckhoff et al. [74]
Directional with Local Beam Tables	Throughput, Channel Utilization	Simulation	Higher throughput than multi-channel protocols	Dense VANET	IEEE 802.11	Wu et al. [14]
Directional with Delay Constraints	Multicast Capacity	Analytical & Simulation	Capacity analysis under mobility models	Random Walk Mobility	Generic VANET	Ren et al. [15]
Steerable Beam Directional	Communication Range, SNR	Real-world Measurement	Range improvements with proper configuration	Inter-vehicular Scenarios	V2V Communication	Subramanian et al. [16]
Sectoral & Omnidirectional	PDR, Throughput, Latency, Power	Simulation	Context-dependent performance trade-offs	Dense VANET (20-90 vehicles)	IEEE 802.11bd	Present Study

3- Methodology and Experiment Framework

3-1- Simulation Platform

Experimental Framework**:** The experimental framework integrates several functionalities into a VANET evaluation environment developed in Python, including custom features such as mobility simulation, IEEE 802.11bd stack, VANET simulation, and Q-Learning communication performance optimization agent. Realistic mobility is achieved using car-following and lane-changing models, with position updates every 100ms. The integrated communication stack provides modelling of the IEEE 802.11bd PHY/MAC layers, including OFDM modulation and LDPC coding. The channel model transitions from AWGN to complex NLOS multipath fading as vehicle density increases. Antenna patterns for both omnidirectional and sectoral configurations are supported in the simulation.

A dual-agent Q-learning framework is integrated within the simulator. The agents operate in sync with the communication stack, receiving state updates and conducting adjustment actions (such as transmission power and beacon rate) every 100ms. Experience replays buffers, and comprehensive logging facilitates efficient learning and analysis.

Multi-Class Background Traffic Modelling: The simulation generates realistic background traffic comprising four classes to model comprehensive network load conditions beyond safety beacons: Periodic Beacons (CAM/BSM): 1-20 Hz, 350 bytes average Infotainment Traffic: Background data applications (video, audio, web). Management Traffic: Control signalling and topology updates. Protocol Signalling: Network coordination messages. The background traffic load is density-dependent. The Channel Busy Ratio (CBR) is sampled from a normal distribution based on vehicle density:

$$CBR_{bg} \sim N(\mu_{density}, \sigma^2_{density}) \quad (1)$$

The corresponding packet generation rate is calculated as:

$$\lambda_{bg} = CBR_{bg} / PacketDuration_{bg} \quad (2)$$

where, L_{bg} is the background packet size in bytes, and R_{eff} is the effective data rate in bits per second.

Additionally, event-driven safety messages are generated probabilistically based on traffic density and receive priority scheduling in the MAC layer for latency and reliability analysis. The total effective CBR experienced by vehicles combines beacon and background traffic:

$$CBR_{eff} = CBR_{beacon} + \alpha \cdot CBR_{bg} \quad (3)$$

where, α is the background load scaling factor, this integrated approach creates realistic channel occupancy conditions for accurate VANET performance, typically 0.5-1.0, depending on traffic class priority and application mix.

Real-Time Agent-Environment Interaction*:* A coordination protocol ensures synchronized interaction between the learning agents and the simulation environment, as depicted in Figure 4. State information is propagated hierarchically: The PHY agent receives signal quality data (SINR, interference, path loss). The MAC agent receives network congestion metrics (CBR, collision rate, queue occupancy). Actions are applied in sequence: PHY actions (power control) first, followed by MAC actions (rate adaptation), with compliance checks for the IEEE 802.11bd standard. Performance feedback is collected in real-time for learning and logged for post-simulation analysis of both individual and coordinated agent performance.

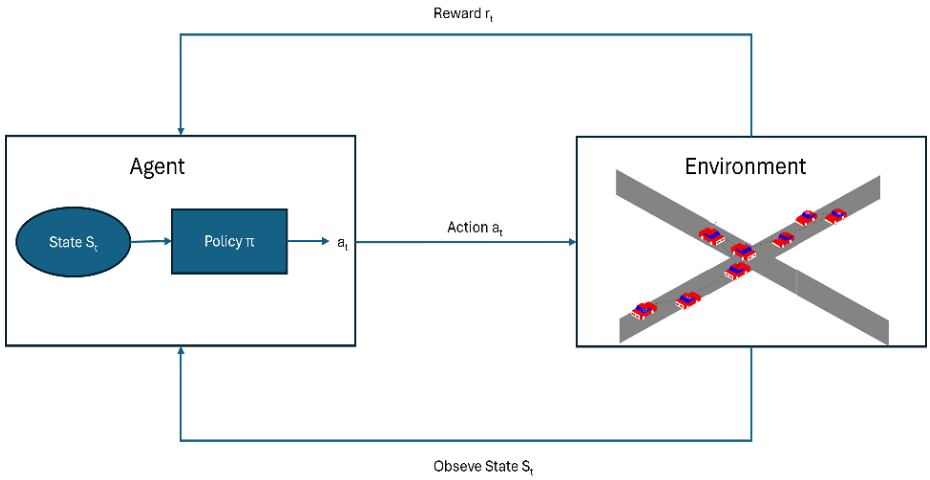

Figure 4. Agent-Environment Interaction

3-2- Simulation Workflow

The simulation workflow, illustrated in Figure 5, consists of the following steps:

- Road and Vehicle Mobility Data Loading: SUMO-generated mobility data is loaded. Vehicle trajectories are sanitized, standardized, and time-aligned to ensure consistency across all agents and timesteps.
- Communication Range and Neighbor Detection: For each timestep, pairwise distances are computed to determine communication ranges and identify neighbors. Path loss, interference, and SINR are calculated using established channel propagation models.
- VANET PHY/MAC Layer Communication: The simulation models background traffic and beacon transmissions. It calculates key performance metrics, including effective SINR, CBR, PDR, and latency. Background traffic and noise power are derived from real-world channel models.
- Data Collection and RL Integration: Vehicles collect local parameter configuration performance metrics (Power transmit, MCS, beacon rate, CBR, SINR) at every interval. These data are transmitted via a socket interface to a Q-Learning agent.
- RL Optimization: The RL agent processes the current observation tuple (CBR, SINR). It outputs a continuous action vector for the three adjustable parameters, constrained within IEEE 802.11bd compliant bounds. The action is sent back via TCP socket. During training, the agent's policy is updated using a reward function based on PDR and SINR, with experience tuples stored in a replay buffer.
- Metrics Logging and Analysis: All communication events and metrics are logged. Post-simulation, averages are computed per vehicle and for the entire network. Results are stored and visualized for performance comparison.

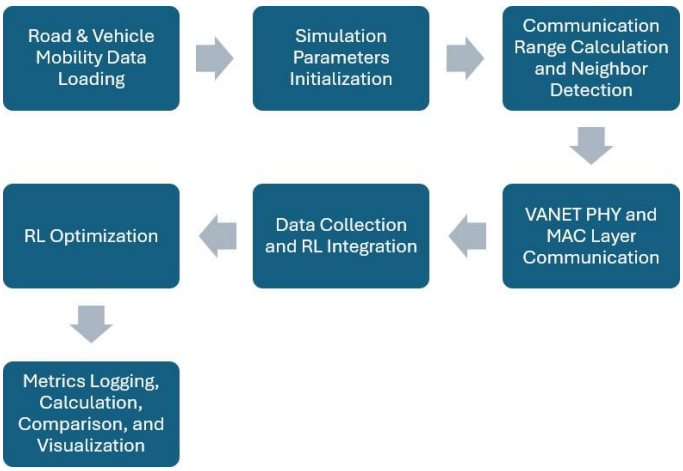

Figure 5. Experiment Workflow

The experiment uses a 1km road segment with six lanes (three lanes in each direction), each 3.7 meters wide, following standard highway design specifications. Vehicle dimensions follow standard specifications, with an average length of 5 m and a minimum distance between vehicles of 2m, creating a 7m front-to-front spacing under dense conditions [75]. The road segment supports vehicle density variations from 20 to 90 vehicles/km/ direction, enabling comprehensive evaluation across sparse to severely congested conditions. These density standards are created based on federal transportation standards, where free-flow conditions typically happen below 7.5 vehicles/km/lane (22.5 vehicles/km/direction for three-lane facilities) and capacity is reached at approximately 28 vehicles/km/lane (84 vehicles/km/direction) [75].

Speed profiles are dynamically adjusted based on density according to the flow-density mapping [75]: 80-120 km/h for sparse conditions (20-30 vehicles/km/direction), 50-80 km/h for moderate density (45-60 vehicles/km/direction), and 30-50 km/h for high density (75-90 vehicles/km/direction). Vehicle density directly influences wireless channel characteristics and network topology in vehicular ad-hoc networks.

3-3- IEEE 802.11bd Stack Modelling.

Communication Stack Integration: The PHY communication stack in this simulation framework is modelled using the UBX-V2X [34], a modular IEEE 802.11bd-compliant PHY stack developed in Python, and a V2X MAC stack also developed in Python, based on a study conducted in Dharsandiya & Patel [76]. This stack is designed to emulate VANET PHY and MAC-layer behaviour, including transmission, channel propagation, and receiver under realistic conditions.

Figure 6 illustrates the detailed operation of the communication stack, which is structured into three main sections: sender, channel, and receiver. The emulation process begins at the sender, where the MAC layer passes frames to the PHY layer for conversion into a bitstream. These bits are going through a scrambling process to improve statistical properties before being encoded using LDPC encoding for forward error correction [77]. The encoded bits are then mapped to modulation symbols according to the selected MCS. The data is modulated using dual-carrier OFDM, enabling high spectral efficiency and robustness against multipath fading. Once modulated, the signal enters the emulated wireless channel [77].

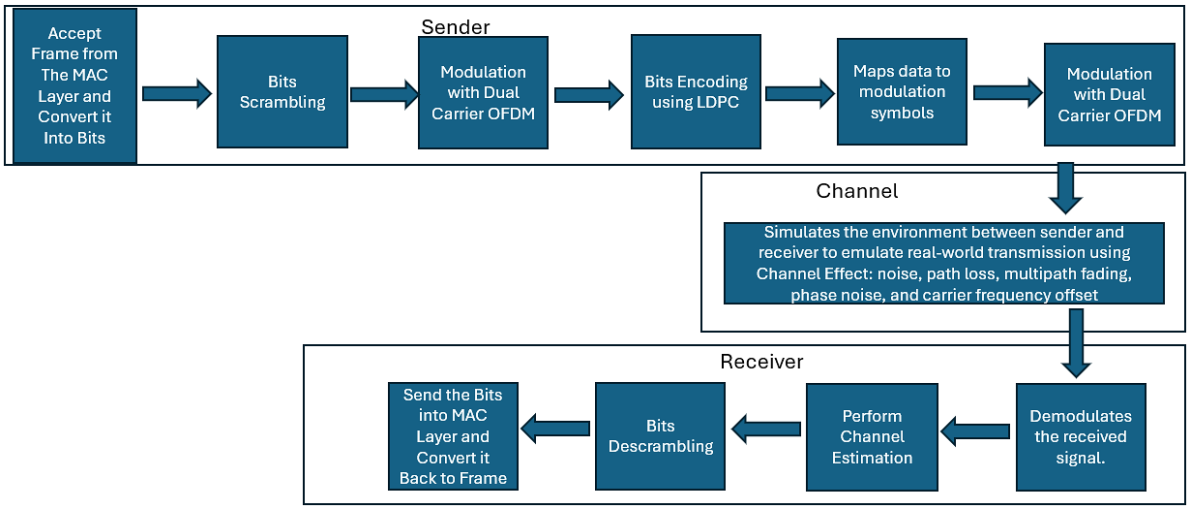

Figure 6. IEEE 802.11bd PHY and MAC emulation flow using the UBX-V2X communication stack

The Channel module simulates radio propagation effects using a Nakagami-*m* fading model combined with additive white Gaussian noise (AWGN), path loss, phase noise, and carrier frequency offset. This ensures that signal degradation does not occur during transmission. Upon reaching the receiver, the signal is demodulated, followed by channel estimation to reconstruct the transmitted symbols. The demodulated symbols are decoded using LDPC and descrambled to retrieve the original bitstream. Finally, the bits are returned to the MAC layer for reassembly into a complete frame.

The MAC layer is fully implemented in Python and adheres to the IEEE 802.11 standard [76, 77]. It controls medium access using CSMA/CA (Carrier Sense Multiple Access with Collision Avoidance), enforces interframe spacing (SIFS, DIFS), and manages retransmissions and contention window adjustments. Frame queuing and prioritization are applied to support heterogeneous traffic classes, beacons, background infotainment, and safety alerts.

Physical Layer Formulation: The experimental framework simulates LOS/NLOS propagation, multipath fading, and Doppler effects. The communication range R is calculated using the modified Friis transmission equation:

$$R = (P\text{t}G\text{t}G\text{r}\lambda 2 / (4\pi)2P\text{s,min})1/\alpha \quad (4)$$

where, Pt is transmission power, Gt and Gr are antenna gains, $\lambda = c/f$ is wavelength at 5.9 GHz, Ps,min is receiver sensitivity (-95 dBm), and α is the path loss exponent (2.5 for highway LOS, 3.0-4.0 for NLOS). Path loss calculation follows the log-distance model:

$$\text{PL}(d) = \text{PL}0 + 10\alpha \log 10(d/d0) + X\sigma \quad (5)$$

where, $PL0$ is the reference path loss at distance $d0$ = 1 m, and $X\sigma$ represents log-normal shadowing with standard deviation $\sigma = 3 - 8$ dB depending on environment density.

MAC Layer Formulation: The IEEE 802.11bd MAC layer implements CSMA/CA with vehicular-specific characteristics [76]. Unlike the generic WLANs, VANETs remove RTS/CTS handshakes to minimize overhead and latency for safety-critical messaging. Retransmission is available for non-safety communication. The backoff mechanism using dynamic contention window adjustment:

$$\text{CW} = \min(\text{CWmax}, 2\text{BE} \cdot \text{CWmin}) \quad (6)$$

where, BE is the backoff exponent, CWmin = 15, and CWmax = 1023 for AC_BE (Best Effort).

Enhanced Distributed Channel Access (EDCA) provides QoS with four access categories:

$$T\text{backoff} = \text{Random}[0, \text{CW}] \times T\text{slot} + \text{AIFSN} \times T\text{slot} \quad (7)$$

where, Tslot = 13 μs is the slot duration for IEEE 802.11bd, and AIFSN (Arbitration Inter-Frame Space Number) specifies the number of slot times before transmission attempt, with AC-specific values: AC_VO (voice) = 2, AC_VI (video) = 3, AC_BE (best effort) = 6, AC_BK (background) = 9.

Transmission Opportunity (TXOP) limits channel occupancy:

$$T\text{TXOP} = \min(T\text{limit}, T\text{packet} + T\text{SIFS} + T\text{ACK}) \quad (8)$$

where, TSIFS = 32 μs is the Short Inter-Frame Space, TACK is the acknowledgment frame transmission time (approximately 44 μs for a 10 MHz channel bandwidth), and Tlimit is the AC-specific TXOP limit that prioritizes safety applications over background traffic. Beacon transmission uses deterministic scheduling with jitter control:

$$T\text{beacon} = T\text{period} + \text{Random}[-T\text{jitter}/2, T\text{jitter}/2] \quad (9)$$

where, Tperiod = 100 ms for basic safety messages and Tjitter = 5 ms prevents synchronized collisions in dense networks.

SINR Calculation and Interference Modeling: SINR calculation includes realistic interference from neighboring vehicles [77]:

$$\text{SINR} = P\text{r} / (\sigma\text{n}2 + \Sigma\text{j}\neq\text{i}P\text{I,j}) \quad (10)$$

where, Pr is received signal power, $\sigma 2 = kTB \cdot NF$ is thermal noise power with k = Boltzmann constant, T = 290K, B = 10 MHz, and NF = 9 dB noise figure, and PI,j represents interference from transmitter j.

PER is calculated using SINR-dependent models for different MCS levels:

$$\text{PER}MCS = 1 - (1 - \text{BER}MCS(\text{SINR}))L\text{packet} \quad (11)$$

where, Lpacket is packet length in bits, and BERMCS follows IEEE 802.11bd specifications with LDPC coding gains.

3-4- Antenna Configuration and Adaptation

The experimental framework evaluates both omnidirectional and sectoral antenna configurations to validate the dual-agent approach across different spatial interference patterns. For sectoral antennas with beamwidth θbeam, the effective neighborhood is modified as:

$$N\text{sectoraleffective} = N\text{total} \cap A(\theta\text{beam}, \varphi\text{pointing}) \quad (12)$$

The state representation incorporates antenna-specific propagation characteristics through density adaptive neighbor counting and enhanced coordination mechanisms.

Omnidirectional Configuration:

Antenna gain: Five and zero dBi uniform pattern with 360° coverage. The gains of five dBi and zero dBi are compared for the static/no-optimization experiment. Neighbor density: Full surrounding vehicle aware- ness Power range: Density-adaptive from 1-30 dBm (rural) to 1-3 dBm (extreme density). Control strategy: Conservative optimization for interference minimization as depicted in Table 3.

Sectoral Configuration: Two-sector system implementing selective RL control. Front/Rear sectors: Five and zero dBi gain, RL controlled power allocation. Five dBi gain and zero dBi gain are compared for the static/no-optimization experiment, and zero dBi gain is achieved for the optimization schema. Effective neighbor reduction: Neff = Ntotal × 0.7 (30% interference reduction). Density-adaptive power ranges: Enhanced upper limits due to directional benefits, SINR enhancement: +3-5 dB typical improvement over omnidirectional. The sectoral antenna system implements gain patterns with smooth transitions:

Table 3. Sectoral Antenna Gain Pattern

Angular Range (θ)	Gain Expression
$\lvert\theta - \theta c\rvert \le 45°$ *Main beam*	Gmax
$45° < \lvert\theta - \theta c\rvert \le 67.5°$ *First sidelobe*	Gmax (1 - 0.15α)
$67.5° < \lvert\theta - \theta c\rvert \le 112.5°$ *Extended coverage*	Gmax (0.85 - 0.35β)
$\lvert\theta - \theta c\rvert > 112.5°$ *Back lobe*	0.2 Gmax

where, θc is the sector center angle, and α and β are normalized angular deviation factors ensuring continuous gain variation.

The key architectural advantage of sectoral antennas lies in the selective RL control strategy: front/rear sectors optimize for dynamic vehicular communication patterns while left/right sectors maintain static power (zero for this experiment) to reduce system complexity.

3-5- Dual-Agent Q-Learning Framework and Implementation.

(1) Motivation and Problem Formulation: Dense IEEE 802.11bd VANETs suffer from congestion, collisions, and interference that degrade safety-message delivery. Static parameters (fixed power, fixed beacon rate) cannot adapt to rapid topology and load changes, leading to high CBR, increased latency, and reduced reliability. Cross-layer adaptation is therefore required.

Why Q-learning: We select tabular Q-learning over deep RL (DQN/PPO/SAC) because: (i) inference is fast (table lookup) and suitable for sub-100 ms control loops; (ii) actions are naturally discrete (dBm steps, Hz steps, MCS levels); (iii) policies are interpretable and easy to debug in safety contexts; and (iv) convergence guarantees exist under standard conditions. Prior work shows Q-learning's effectiveness for VANET congestion/power control [9], while deep methods can be competitive but heavier to tune and deploy in real time [27, 49].

Problem setting: Consider N vehicles operating under IEEE 802.11bd. Each vehicle i chooses a control vector

$$\text{xi} = [P\text{i}, B\text{i}, M\text{i}]\text{T} \quad (13)$$

where, Pi ∈ [1, 30] dBm is transmit power, Bi ∈ [1, 20] Hz is beacon rate, and Mi ∈ {0, . . 9} is the MCS index (10 MHz profile). The vehicle observes local congestion and link quality

$$s\text{i} = (\text{CBRi}, \text{SINRi}, N\text{i}, \ldots) \quad (14)$$

where Ni is the neighbor count. Objectives are: (i) maintain channel utilization near target (CBR≈ 0.6), (ii) maximize reliability (PDR), (iii) minimize latency, and (iv) reduce power. To ensure standards compliance and predictable reliability, MCS is selected deterministically from a SINR-based lookup:

$$Mi \leftarrow flookup(SINRi - margin) \quad (15)$$

following IEEE 802.11 thresholds [6, 77].

Limitations of joint single-agent control: A single-agent that jointly optimizes (P, B, M) faces: (1) *temporal mismatch* (PHY reacts on millisecond scales, MAC must be more stable);

(2) *competing objectives* (increasing P improves SINR but raises interference/CBR; increasing B improves awareness but raises congestion); and (3) *state-action space explosion* (naive discretization leads to ~7.1×108 joint states/actions), slowing learning and harming convergence [11, 27].

Formulation rationale for dual-agent design: We therefore decompose control into two specialized agents: a PHY agent that adjusts P to meet SINR/power-efficiency goals, and a MAC agent that adjusts B to meet CBR/throughput goals. MCS remains a deterministic function of SINR for IEEE 802.11bd compliance. This separation aligns with layer timescales, reduces interference between objectives, and shrinks effective complexity while retaining cross-layer awareness via shared state and coordination rewards. Later sections detail the architecture, rewards, algorithm workflow (including our single-agent baselines and dual-agent DQN comparison), and theoretical complexity analysis .[11, 27, 49]

Architecture Design*:* We design a dual-agent architecture that separates PHY and MAC control, illustrated in Figure 7. The PHY agent adapts transmit power for SINR and efficiency, while the MAC agent adapts beacon rate for congestion management. MCS is not learned; it is deterministically chosen from the IEEE 802.11bd SINR-based lookup (Table 5), ensuring standards compliance [6].

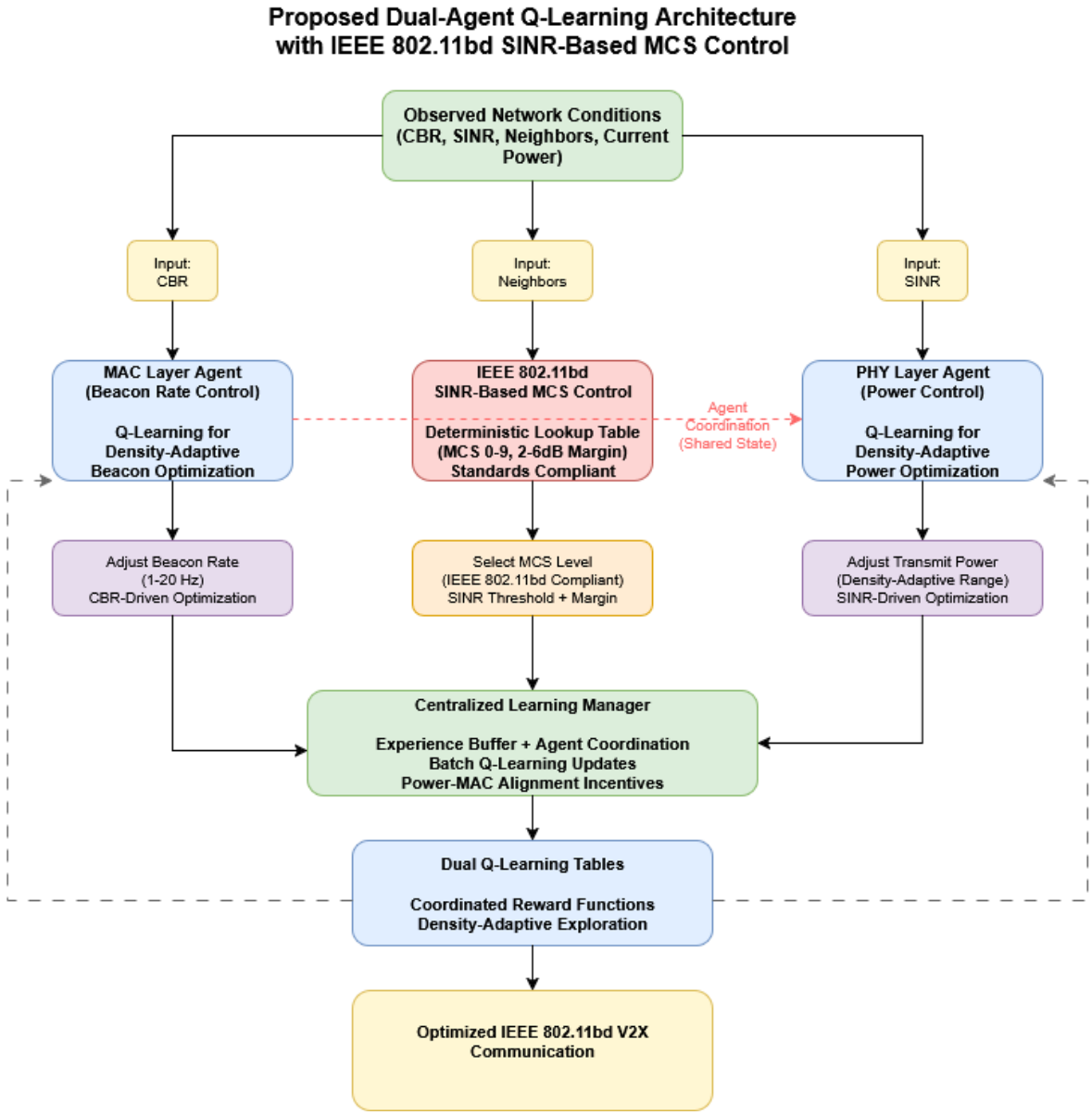

Figure 7. Proposed dual-agent Q-learning Framework Showing Hierarchical Coordination Between PHY and MAC Agents

State design: Each agent observes local congestion and link quality.

$$s_{PHY}^{t} = (CBR^{t}, SINR^{t}, P^{t}, N^{t}) \quad (16)$$

$$s_{MAC}^{t} = (CBR^{t}, SINR^{t}, B^{t}, N^{t}, a_{PHY}^{t}) \quad (17)$$

Where, N t is the neighbor count. Discretization reduces complexity: |SPHY| = 60,000 and |SMAC| = 520,000, versus ∼7.1×108 for joint single-agent control.

Action design: PHY actions adjust power in dBm steps; MAC actions adjust beacon rates in Hz steps

$$\mathcal{A}\text{PHY} = \{0, \pm1, \pm2, \pm3, \pm5, \pm10, \pm15\} \quad (18)$$

$$\mathcal{A}\text{MAC} = \{0, \pm1, \pm2, \pm3, \pm5, \pm10\} \quad (19)$$

Power and rate are clipped to IEEE 802.11bd-compliant bounds. Density-adaptive constraints further narrow the feasible space in Table 4.

Table 4. Density-Adaptive Power Range Constraints

Neighbors	Category	Omnidirectional (dBm)	Sectoral (dBm)
≤ 4	Very Low	[1,30]	[1,30]
5–8	Low	[1,20]	[1,22]
9–12	Medium	[1,15]	[1,17]
13–17	High	[1,10]	[1,12]
18–25	Very High	[1,6]	[1,8]
> 25	Extreme	[1,3]	[1,4]

MCS selection (deterministic): Given measured SINR, the transmitter chooses

$$\text{MCSt} \leftarrow f\text{lookup}(\text{SINRt} - \text{margin}) \quad (20)$$

with thresholds in Table 5, sectoral antennas use a smaller margin due to improved interference rejection.

Table 5. IEEE 802.11bd MCS-SINR Lookup (10 MHz Channel)

MCS	Mod.	Rate	SINR Thresh. (dB)	Data Rate (Mbps)
0	BPSK	1/2	< 2.0	2.2
1	BPSK	3/4	2.0–3.9	3.3
2	QPSK	1/2	4.0–6.9	6.5
3	QPSK	3/4	7.0–9.9	9.8
4	16-QAM	1/2	10.0–12.9	13.0
5	16-QAM	3/4	13.0–15.9	19.5
6	64-QAM	2/3	16.0–19.9	26.0
7	64-QAM	3/4	20.0–22.9	29.3
8	256-QAM	3/4	23.0–25.9	39.0
9	256-QAM	5/6	≥ 26.0	43.3

Hierarchical control: At each 100 ms interval, the PHY agent acts first to stabilize SINR, then the MAC agent adapts beacon rate conditioned on at PHY's faster dynamics of PHY control.

Antenna integration: Both omnidirectional and 4-sector configurations are supported (Table 9). For sectoral, only front/rear sectors are RL-controlled, while left/right remain static. For fairness, all optimization experiments use zero-gain settings; nominal gains are analyzed separately in Section IV-F.

3) Reward Design and Coordination: The reward functions are designed to decouple PHY and MAC objectives while ensuring cross-layer coordination. Both agents operate on density-adaptive state spaces and are constrained by IEEE 802.11bd power and MCS limits [6]. Previous RL approaches for VANETs mainly used simple rewards focused on a single metric, such as latency, throughput, CBR, SINR, or reliability [27, 49]. While straightforward, these designs often create large state–action spaces and potentially become unstable under congestion. In contrast, our reward design adds density-aware shaping, smoothness penalties, and cross-layer alignment, allowing PHY and MAC agents to work toward complementary goals. This richer structure improves stability and scalability, but it also requires careful parameter tuning and assumes that PHY and MAC objectives remain aligned, which may limit generalization [78] The resulting reward structure is as follows:

***PHY Agent Reward*:** The PHY agent optimizes transmission power with emphasis on SINR quality and energy efficiency

$$r_{\mathrm{PHY}} = R_{\mathrm{SINR}} + R_{\mathrm{pow\text{-}eff}} + R_{\mathrm{coord}} + R_{\mathrm{smooth}} \tag{21}$$

where, R_{SINR} is a piecewise function: linear below the target (12 dB) and diminishing above, $R_{\mathrm{pow\text{-}eff}}$ penalizes deviation from density-adaptive optimal power, R_{coord} provides a bonus when PHY actions complement MAC beacon decisions, and where R_{SINR} is a piecewise function: linear below the target (12 dB) and diminishing above, $R_{\mathrm{pow\text{-}eff}}$ penalizes deviation from density-adaptive optimal power, R_{coord} provides a bonus when PHY actions complement MAC beacon decisions, and *R*smooth penalizes large power jumps.

***MAC Agent Reward*:** The MAC agent regulates beacon rate based on channel load and neighbor density:

$$r_{\mathrm{MAC}} = R_{\mathrm{CBR}} + R_{\mathrm{beacon}} + 0.3R_{\mathrm{PHY}} + R_{\mathrm{align}} + R_{\mathrm{smooth}} \tag{22}$$

where, R_{CBR} rewards keeping channel load near 0.6, R_{beacon} penalizes deviation from density-adaptive optimal rates, the PHY reward is partially inherited ($0.3R_{\mathrm{PHY}}$) to promote coordination, R_{align} rewards complementary moves (power↑ with beacon ↓), and Rsmooth penalizes beacon oscillations. Table 6 summarizes all reward components and their optimization intent.

Table 6. Reward Components and Intent

Component	Definition / Intent
R_{SINR}	Shaped function of SINR; linear below target, diminishing above.
$R_{\mathrm{pow\text{-}eff}}$	Penalizes deviation from density-optimal normalized power.
R_{CBR}	Rewards keeping CBR near CBR target.
R_{beacon}	Penalizes deviation from density-optimal beacon rate.
R_{coord}	PHY reward bonus passed to MAC for cross-layer coupling.
R_{align}	Rewards complementary PHY/MAC actions.
R_{smooth}	Penalizes large ΔP or ΔB.

***Cross-layer Coordination*:** Both agents share (CBR, SINR, N) and act hierarchically (PHY first). Alignment rewards ensure stability, while density-aware exploration prevents over-aggressive behaviour in congested networks. The final reward is clipped to bounded ranges ([−25, 25]), ensuring stable Q-learning updates [79].

4} Algorithm Workflow: **Step-by-step flow (per control interval, 100 ms):**

1. *Sense:* Each vehicle measures local CBR, SINR, neighbor count *N*, and reads its current (*P*,*B*).
2. *MCS lookup:* Select $MCS \leftarrow f_{\mathrm{lookup}}(\mathrm{SINR}-\mathrm{margin})$ using Table 4.
3. *PHY action:* PHY agent chooses $a_{\mathrm{PHY}}{}^{t}$; update $P \leftarrow \mathrm{clip}(P+\Delta P$, density bounds from Table 5).
4. *MAC action:* MAC agent conditions on $a_{\mathrm{PHY}}{}^{t}$ and chooses $a_{\mathrm{MAC}}{}^{t}$; update $B \leftarrow \mathrm{clip}(B+\Delta B,1,20)$.
5. *Transmit:* The stack executes with updated (*P*,*B*,*MCS*); channel models produce new CBR, SINR, PDR, and latency.
6. *Learn:* Compute r_{PHY}, r_{MAC} (reward design described earlier); update Q_{PHY}, Q_{MAC} by temporal-difference learning.

Log: Store (*s*,*a*,*r*,*s'*) and record KPIs (PDR, latency, power, throughput)

Pseudo-code:

Algorithm 1. Coordinated Dual-Agent Q-Learning (Main Loop)

```
Inputs: action sets 𝒜PHY, 𝒜MAC; power bounds table (Table~\ref{tab:density_power_constraints}); MCS lookup (Table~\ref{tab:mcs_sinr_lookup})
Init: QPHY ← 0, QMAC ← 0; α = 0.15 (sectoral 0.18), γ = 0.95, ε = 1.0; replay/log buffers 𝒟
for episode e = 1 to 15000 do
    Reset env.; observe s^0 = (CBR^0, SINR^0, N^0, P^0, B^0)
    for t = 0 to T do
        // MCS selection (standard-compliant)
        MCS^t ← flookup(SINR^t - margin)
        // PHY action & density-adaptive power bounds
        a^tPHY ← ε-greedy(QPHY(s^tPHY, ·))
        (Pmin, Pmax) ← BoundsFromDensity(N^t)     // Table~\ref{tab:density_power_constraints}
        P^{t+1} ← clip(P^t + ΔP(a^tPHY), Pmin, Pmax)
        // MAC action conditioned on PHY
        a^tMAC ← ε-greedy(QMAC(s^tMAC, ·))
        B^{t+1} ← clip(B^t + ΔB(a^tMAC), 1, 20)
        // Step environment with updated (P, B, MCS)
        Execute TX/RX; observe s^{t+1}, KPIs (PDR, latency, throughput), and (CBR^{t+1}, SINR^{t+1})
        // Rewards with coordination and smoothing (defined below)
        (r^tPHY, r^tMAC) ← ComputeRewards(s^t, s^{t+1}, P^t → P^{t+1}, B^t → B^{t+1})
        // Tabular TD updates
        QPHY ← QPHY + α[r^tPHY + γ max_a QPHY(s^{t+1}PHY, a) - QPHY(s^tPHY, a^tPHY)]
        QMAC ← QMAC + α[r^tMAC + γ max_a QMAC(s^{t+1}MAC, a) - QMAC(s^tMAC, a^tMAC)]
        // Log for analysis & convergence checks
        Append (s^t, a^tPHY, a^tMAC, r^tPHY, r^tMAC, s^{t+1}) and KPIs to 𝒟
    end for
    // Exploration schedule
    ε ← max(0.1, ε × 0.9995)
    // Optional: early-stop if reward/TD error stable (≤5% over last 100 episodes)
end for
```

Training setup: Episodes: 15000 for Q-Learning algorithms and 45000 for DQN algorithm; steps/episode: 100; interval: 100 ms. Exploration: ϵ-greedy with exponential decay to 0.1. Learning rate α: 0.15 (sectoral 0.18); discount γ: 0.95. Actions: discrete step sets as defined earlier. MCS: deterministic lookup (Table 5). Power bounds: density-dependent (Table 6). Metrics logged: PDR, latency, throughput, power, CBR, SINR.

Story of progression:

- *Single-agent Q-learning (power-only):* Adjusts *P* targeting CBR; reveals instability in dense networks.
- *Single-agent Q-learning (power+beacon):* Joint (*P*, *B*) control; suffers from state-action explosion and conflicting objectives.
- *Dual-agent Q-learning (proposed):* Decouples PHY/MAC control with coordination; improves latency and power efficiency while keeping PDR acceptable.
- *Dual-agent DQN (comparison):* Same decomposition, but with neural function approximators. Competitive results but higher resource demand and slower convergence.

Algorithm 2. ComputeRewards (PHY/MAC, with Coordination)

Inputs: s^t, s^{t+1} (CBR, SINR, N, P, B), deltas ΔP, ΔB

PHY terms: R_{SINR} from $SINR^{t+1}$, $R_{power-eff}$ from density-normalized P^{t+1}, R_{smooth} from $|\Delta P|$

MAC terms: R_{CBR} from $|CBR^{t+1} - 0.6|$, $R_{beacon-opt}$ from density target $B_{opt}(N^{t+1})$, R_{smooth} from $|\Delta B|$

Coordination: $R_{align} = +2$, $\Delta P > 0$ & $\Delta B \leq 0$

$+2$, $\Delta P < 0$ & $\Delta B \geq 0$

$+1$, $|\Delta P| \leq 1$ & $|\Delta B| \leq 1$

0, otherwise

$r_{PHY} \leftarrow w_1 R_{SINR} + w_2 R_{power-eff} + w_3 R_{align} + w_4 R_{smooth}$

$r_{MAC} \leftarrow w_5 R_{CBR} + w_6 R_{beacon-opt} + w_7 R_{align} + w_8 R_{smooth}$

return (r_{PHY}, r_{MAC})

Implementation notes: Agents are integrated into the CANVAS/UBX-V2X simulator, synchronized with the PHY/MAC stack. Background traffic and density-driven channel models follow the methodology described earlier. Antenna mode (omnidirectional or sectoral) is set per scenario; zero gain is used during optimization experiments for fair algorithmic comparison.

1) Comparative Theoretical Analysis:

Complexity: Decomposing control into two agents reduces the tabular *Q* size from a joint state–action product to a sum:

$$|\mathcal{S}_{\text{joint}}| \cdot |\mathcal{A}_{\text{PHY}}| \cdot |\mathcal{A}_{\text{MAC}}| \rightarrow |\mathcal{S}_{\text{PHY}}| \cdot |\mathcal{A}_{\text{PHY}}| + |\mathcal{S}_{\text{MAC}}| \cdot |\mathcal{A}_{\text{MAC}}| \tag{23}$$

With the discretization used earlier ($|S_{PHY}|$= 60,000, $|A_{PHY}|$= 13; $|S_{MAC}|$ = 520,000, $|A_{MAC}|$ = 11), the proposed dual-agent tabular *Q* has ≈ 0.78 M + 5.72 M ≈ 6.5 M entries. A single-agent joint controller (same state factors combined, joint action set) typically falls in $O(10^8–10^9)$ entries depending on granularity. DQN replaces tables with neural approximators (parameter count set by the chosen MLP), reducing memory but increasing per-step compute.

Convergence: Classical tabular Q-learning converges to $Q^\star$ under standard assumptions (stationary MDP, diminishing step sizes, and sufficient state–action visitation). In practice, constant learning rates are used for responsiveness; we rely on empirical stability checks (reward/TD-error plateaus, policy consistency). Function approximation has no general convergence guarantees and may diverge without careful tuning (target networks, replay buffers, and regularization). The dual-agent decomposition also mitigates non-stationarity between layers by (i) hierarchical action timing and (ii) a small alignment term in the rewards [80].

Scalability: The dual-agent design scales *additively* with new controls (extra agent adds $|S_k||A_k|$), rather than *multi* as in joint control. This preserves tractability when extending with additional MAC knobs or antenna-side parameters. Per-step action selection is also lighter: each agent scans only its own action set (13 for PHY, 11 for MAC), versus a joint scan over their Cartesian product [81].

Advantages vs. single-agent and DQN:

- *Versus single-agent tabular:* Smaller Q spaces,

Faster/steadier learning, fewer objective conflicts (PHY vs. MAC), better interpretability (layer-specific tables) [79, 81].

- *Versus dual-agent DQN:* Lower per-step compute, simpler tuning, transparent policies; DQN can be more sample-efficient in continuous/high-res settings but needs larger training budgets and careful stabilization [82, 83].

Memory Footprint and Hardware Feasibility:

Using 32-bit floating-point values with 4 bytes per entry, total memory consumption is approximately 25 MB. This is significantly smaller than typical deep learning models deployed on edge devices, such as VGG-16 (512 MB) or AlexNet (217 MB) [84]. Such compact models are well-suited for resource-constrained vehicular edge devices [85] and fit comfortably within typical vehicular On-Board Unit (OBU) specifications. Modern vehicular OBUs commonly provide 1 GB to 2 GB RAM, as demonstrated in commercial products such as the Commsignia OBU Lite (2 GB DDR3 SDRAM) [31] and Ajeevi OBU (1 GB RAM) [32], While research-grade platforms feature 8-24 GB of memory (as evidenced by the RTX 3070Ti with 8GB and RTX 4090 with 24GB used in recent vehicular edge computing research),

with system RAM typically ranging from 16-32 GB or more for deep reinforcement learning applications [86, 87]. In contrast, single-agent joint control, requiring 100-700 million entries, would need 400 MB to 2.7 GB. While this fits within modern commercial OBUs (1-2 GB RAM), it consumes 20-100% of available memory, leaving insufficient resources for the operating system, communication stack, and other vehicular applications.

The dual-agent approach at 25 MB uses only 1.25-2.5% of OBU memory, enabling efficient resource sharing. Per-step inference uses tabular Q-learning methods where value functions are represented as arrays [79], enabling fast direct lookups. Each agent scans its action set (13 actions for PHY, 11 for MAC) to find the maximum Q-value, requiring 24 table lookups total per decision cycle. This is computationally more efficient than the neural network forward passes required by deep RL methods. ARM-based embedded processors commonly used in vehicular systems (such as 4-core 1.2 GHz processors) [87] have limited computational capabilities for complex tasks. Due to the computationally intensive nature of deep learning inference and real-time data processing, vehicles must offload tasks to edge servers to meet latency requirements [86, 87] The vehicular edge computing architecture enables task processing within acceptable delay constraints, with task intervals typically around 100 ms [87] and maximum tolerated delays of 5-8 seconds for compute-intensive operations [86]. Scalability limits depend on state space growth. The current discretization supports vehicle densities up to 100 vehicles per kilometer as tested in our experiments. Beyond this density, neighbor count bins would require expansion, and finer CBR discretization might be needed, increasing memory requirements linearly with state space size [79]. We estimate the practical limit at approximately 150 vehicles per kilometer, at which point the state-space expansion would increase the memory footprint from 25 MB to approximately 80-100 MB. While this remains within the capacity of modern commercial OBUs (1-2 GB RAM), it would consume 5-8% of available memory compared to the current 1.25-2.5%, reducing headroom for other applications. These characteristics demonstrate that the dual-agent Q-learning approach is feasible for on-board deployment in modern vehicular OBUs, requiring minimal memory footprint (1.25-2.5% of available RAM) and computational resources.

Limitations and Restrictions:

- Tabular methods depend on discretization; very fine grids reintroduce large *Q* tables [79].
- Dual-agent learning still experiences early-phase non-stationarity; alignment rewards and synchronized timing reduce, but do not eliminate, this effect [81].
- DQN can compress state spaces and support richer observations but increases engineering complexity and lacks general convergence guarantees [82, 88].

These results explain the empirical behavior observed earlier: the proposed dual-agent tabular method offers a practical balance, much smaller *Q* spaces than single-agent tabular, faster stabilization than DQN in our setting, and sufficient flexibility to meet latency/power goals while keeping reliability acceptable (with deterministic, standard-compliant MCS selection). In general, the complexity comparison between algorithms is shown in Table 7.

Table 7. Algorithms Complexity Comparison

Approach	Q-table size / Parameters	Per-step computation	Sample complexity (order)
Single-agent Tabular	$O(10^8 - 10^9)$ entries	~|A_PHY||A_MAC| scans	Proportional to table size
Dual-agent Tabular (proposed)	$\approx 6.5\times10^6$ entries	~|A_PHY| + |A_MAC| scans	Proportional to sum of table sizes
Dual-agent DQN	$\sim10^5$ parameters per agent (typical)	Forward/backward passes per step	Depends on optimizer and replay buffer

F. Performance Metrics and Simulation Parameters

Performance Metrics: The evaluation employs standard communication KPIs and additional coordination measures tailored to the dual-agent framework.

Core KPIs:

- *Packet Delivery* Ratio *(PDR)* quantifies reliability as

$$\mathrm{PDR} = (N_{\mathrm{received}} / N_{\mathrm{transmitted}}) \times 100\% \quad (24)$$

where, N_{received} and $N_{\mathrm{transmitted}}$ denote packets decoded and transmitted within the communication range.

- *Channel Busy Ratio (CBR)* captures medium occupancy through

$$\mathrm{CBR} = T_{\mathrm{busy}} / T_{\mathrm{observation}} \quad (25)$$

where, T_{busy} is the cumulative busy time over the observation window Tobservation.

- *Signal-to-Interference-plus-Noise Ratio (SINR)* characterizes link quality as

$$\mathrm{SINR} = P_{\mathrm{signal}} / (P_{\mathrm{interference}} + P_{\mathrm{noise}}) \quad (26)$$

Throughput measures effective data delivery,

$$\eta = (N_{\text{received}} \cdot L_{\text{packet}}) \,/\, T_{\text{observation}} \quad (27)$$

with L_{packet} as packet length in bits.

- *Latency* is the average end-to-end transmission delay between vehicles, measured at the application layer.
- *Power Efficiency* captures the energy-performance trade-off as

$$\eta_P = \text{Throughput} \,/\, \text{Power Consumption} \quad (28)$$

Coordination Metrics:

Cooperation Rate is the fraction of time agents choose complementary actions,

$$\xi_{\text{coop}} = N_{\text{cooperative}} \,/\, N_{\text{total}} \quad (29)$$

where, cooperative actions are defined as power increases paired with beacon stability/decreases, or power decreases with beacon stability/increases.

- *Reward Correlation* uses Pearson's correlation coefficient (r) between rPHY and rMAC to quantify the alignment of objectives. Positive values indicate cooperative behavior.
- *Convergence Time* is the number of episodes required for policies to stabilize within 5% of their long-term average performance.

2) Reward Configurations and Simulation Parameters: In the experiment with the dual-agent Q-learning algorithm, five reward configurations (Config 1–5) are defined to investigate different optimization priorities. The configurations are created by adjusting reward weights, penalty terms, and exploration limits within the algorithm. Config 1 (throughput-oriented) pushes higher MCS levels and beaconing rates, with lighter penalties on power consumption. Config 2 (conservative) implements strong power-related penalties and discourages beacon rate increases, creating stable and low-power operation. Config 3 (PDR-oriented) increases the weight of SINR and reliability margins, making the controller more sensitive to packet success rate. Config 4 (balanced) assigns equalized weights across latency, power efficiency, and throughput objectives. Finally, Config 5 (high-density) introduces density-aware exploration limits (smaller power and beacon adjustments at high vehicle densities) and stronger PHY–MAC coordination terms to sustain performance under congestion. The experiment framework integrates these reward configurations with a traffic mobility model, an IEEE 802.11bd-compliant communication stack, and the proposed dual-agent RL controller. The main simulation parameters are summarized in Table 8, while the antenna configurations are in Table 9.

Table 8. Simulation Configuration Parameters

Category	Parameter	Value / Description
Mobility	Road Layout	500 m, 6-lane highway
	Vehicle Density	20–90 (40–180 veh/km/dir)
	Speed Range	5–120 km/h (density adaptive)
	Vehicle Length	5 m
	Position Update	100 ms
	Simulation Duration	1500 s
Communication	Standard	IEEE 802.11bd (10 MHz)
	Carrier Frequency	5.9 GHz
	Transmission Power	1–30 dBm (RL controlled)
	Beacon Rate	1–20 Hz (RL controlled)
	MCS Range	0–9 (SINR-based lookup, Table 5)
	Channel/Fading	AWGN, NH-LOS, H-LOS and Nakagami-m (m = 1–5)
Dual-Agent RL	Algorithm	Tabular
	Learning Rate	Q-learning
	Discount Factor	0.15 (0.18 for sectoral)
	Exploration	0.95
	Episodes / Steps	ϵ-greedy with decay to 0.1
	Update Frequency	15000 episodes, 100 steps every step
Traffic	Background Load	Gaussian CBR $\mu \in [0.15, 0.95]$, σ = 0.02
	Management Traffic	0.5–1.0 Mbps per vehicle
	Infotainment Traffic	1–5 Mbps per vehicle

Table 9. Antenna Configuration Comparison

Parameter	Omnidirectional	Sectoral (4-sector)
Gain	2.14 dBi (uniform)	10 dBi (F/R), 8 dBi (L/R)
Beamwidth	360°	90° per sector
Coverage	Full	Directional with sidelobes
Effective Neighbors	N	$N \times 0.3 \times 1.2$
SINR Effect	Baseline	+3 dB typical
Power Control	1–30 dBm	1–30 dBm (F/R), 15 dBm (L/R)
RL Learning Rate	0.15	0.18
Coordination Weight	0.25	0.30

3)Traffic and Background Modeling: In addition to periodic safety beacons, the simulation incorporates a realistic background load to capture management and infotainment traffic. Background CBR is modelled as a Gaussian random

variable with density-dependent mean $\mu \in [0.15, 0.95]$ and standard deviation $\sigma = 0.02$, representing conditions from sparse flow to heavy congestion. Management traffic ranges from 0.5–1.0 Mbps per vehicle, while infotainment traffic spans 1–5 Mbps per vehicle. Safety-critical messages are generated probabilistically, with trigger rates increasing from 1% in sparse traffic to 10% under congestion. This setup ensures that agent optimization is evaluated under diverse and realistic VANET load conditions.

Validation: To ensure validity and reproducibility, all experiments follow a structured training–evaluation process with statistical testing.

Training phase: Each configuration (antenna type, vehicle density, algorithm variant) is trained for 15000 episodes. Exploration uses ϵ-greedy with exponential decay to 0.1. Convergence is verified through three criteria: (i) reward stability, with the running average varying by no more than 5% across the last 100 episodes; (ii) action consistency, where the distribution of (ΔP, ΔB) stabilizes; and (iii) change-point detection on temporal-difference error to confirm plateauing behavior.

Evaluation phase: After convergence, policies are frozen and evaluated in greedy mode (ϵ=0). Each configuration is run 30 independent times with randomized seeds for mobility, fading, and initialization. A 2 s warm-up period is discarded to eliminate transients. Key performance indicators (PDR, latency, throughput, SINR, CBR, power efficiency) are recorded every 100 ms, while coordination metrics (Cooperation Rate, Reward Correlation, Convergence Time) are logged per control step.

Statistical analysis: Reported results include mean values and variability measures (standard deviation and selected distributional statistics) aggregated across experiments. Confidence intervals and formal hypothesis testing are not applied in this work; instead, comparisons between algorithms are based on performance trends observed across metrics.

Fairness controls: All algorithms share identical mobility traces, background traffic, channel seeds, and antenna settings in each trial. IEEE 802.11bd compliance is enforced through deterministic SINR–MCS mapping (Table 5) and density-adaptive power limits (Table 4). Traffic loads follow the parameters in Table 8.

Reproducibility: Mobility trace, simulator version, and configuration are logged for every run. All figures and tables in the Performance Evaluation section are generated directly from these logged artifacts.

4- Performance Evaluation and Validation

This section evaluates the proposed dual-agent Q-learning against static baselines, single-agent Q-learning, and dual-agent DQN under IEEE 802.11bd settings, using both omnidirectional and sectoral antenna configurations. All experiments use the parameters defined in Table 8.

4-1-Core KPI Performance

1) ***PDR**:* Static omnidirectional achieves the highest PDR (94.2%); sectoral static achieves 93.6%. Dual-agent Q-learning maintains 88–89% PDR (5 percentage points vs. static), with modest degradation at 20 vehicles (2–3 points) and larger at 90 vehicles (8–12 points). Single-agent variants fall below 25%, confirming their instability in dense VANETs as depicted in Figure 8.

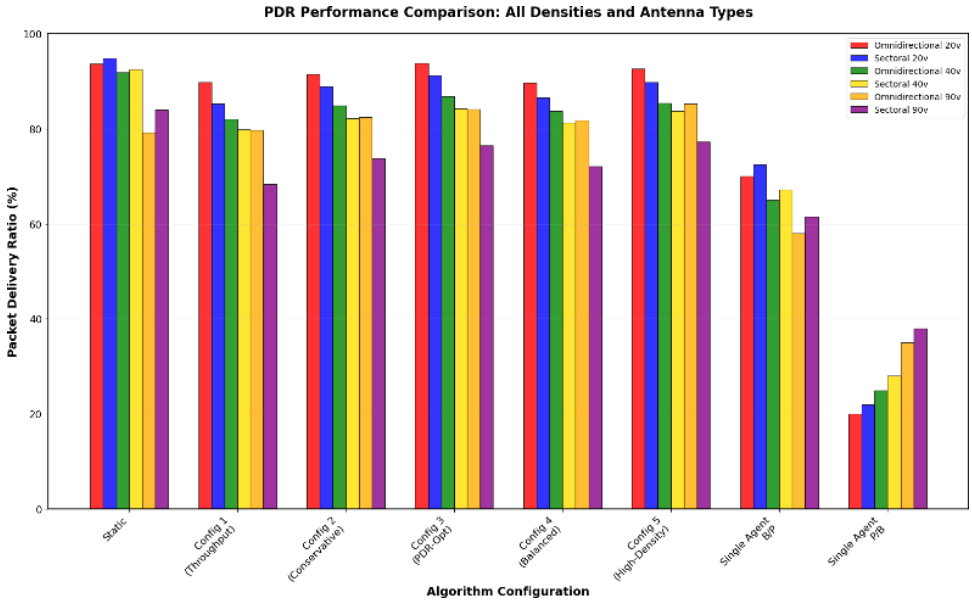

Figure 8. PDR Across Vehicle Densities (20–90) for Static, Single-Agent, Dual-Agent Q-Learning, and Dual-Agent DQN. Zero-Gain Antenna Setting During Optimization

The PDR reduction stems from aggressive power control in dense scenarios. Static configurations maintain 15-20 dBm to ensure connectivity, while dual-agent optimization reduces power to 3-9 dBm to minimize interference. This creates a reliability-efficiency trade off. At low density (20 vehicles/km), the PDR drop is only 2-3% because channel congestion is limited, and power reduction primarily saves energy. At high density (90 vehicles/km), the 11% drop occurred because lower transmit power increases hidden terminal problems when agents independently reduce power without global coordination. The achieved 88.6% PDR still exceeds the SAE J2945/1 minimum requirements 85% for cooperative awareness applications.

2) ***Latency:*** Dual-agent control reduces average latency from 31.1 ms (static) to 17.2 ms (44.6%). The best configuration achieves 12.75 ms while sustaining 89% PDR. Gains are density-dependent: 35–40% improvement at low density, 50–55% at high density. Single-agent runs show unstable latency, with spikes exceeding 300 ms as depicted in Figure 9.

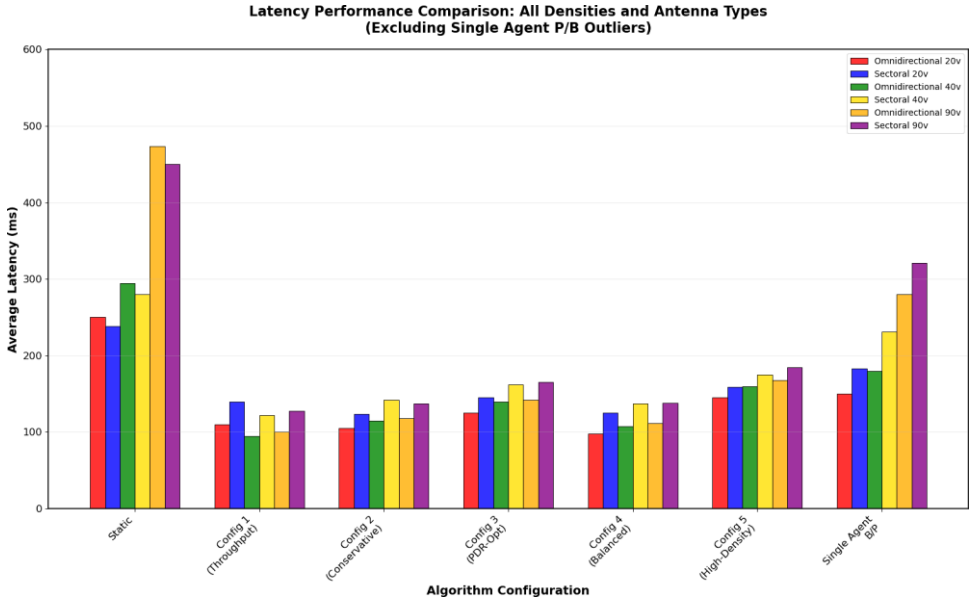

Figure 9. Latency across configurations and densities

Latency improvement comes from two sources. First, the MAC agent manages the channel busy ratio near the target 0.6, reducing contention and backoff delays. Second, the PHY agent's power control lowers interference, allowing faster CSMA/CA resolution. The 45% reduction or 14ms improvement is density-dependent: at low density, gains come mainly from optimized beacon timing, with a 35-40% improvement, while at high density, interference

reduction dominates, with a 50-55% improvement. The achieved 17.2ms average approaches the IEEE 802.11bd theoretical minimum (DIFS + SIFS + transmission time ≈ 12ms), suggesting limited room for further optimization without protocol changes.

3) ***Throughput****:* Static configuration achieves a mean throughput of 10.35 Mbps; dual-agent achieves 9.6 Mbps (7.7%).

Sectoral antennas outperform omnidirectional antennas by 10–15%, even under zero-gain conditions, reflecting reduced interference. In real deployments with 3–6 dB gain, greater improvements are expected as depicted in Figure 10.

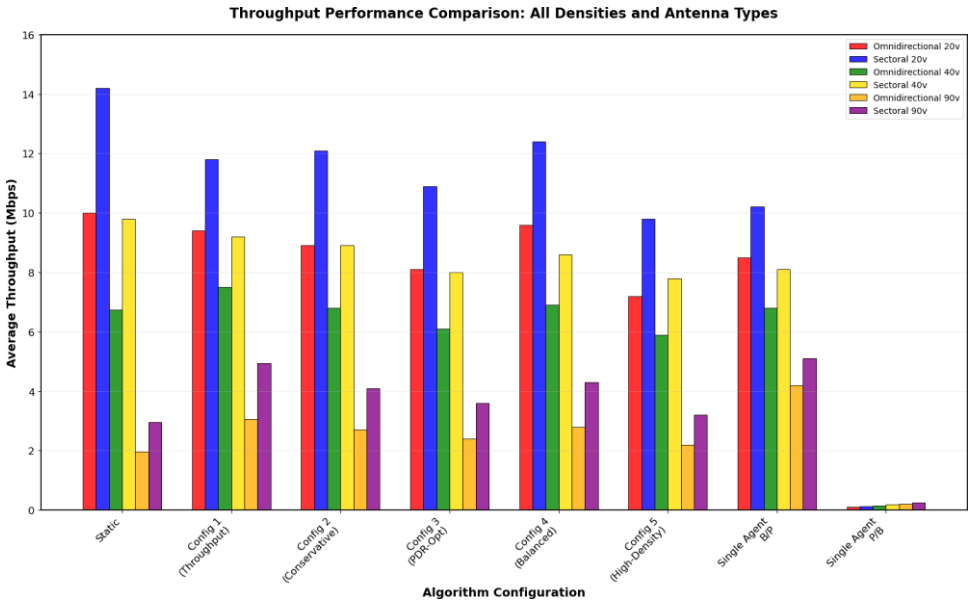

Figure 10. Throughput across configurations

Static configuration achieves a mean throughput of 10.35 Mbps, while dual-agent achieves 9.6 Mbps, a 7.7% reduction. This happened because the dual-agent prioritizes latency and power efficiency over raw throughput; lower transmit power, 9 dBm vs 15-20 dBm, reduces SINR margins, limiting MCS selection, and the MAC agent targets a CBR of 0.6 rather than maximizing channel utilization. Sectoral antennas offset this loss by 10-15% through reduced spatial interference. The reduction is acceptable for safety applications where message delivery and low latency matter more than high data rates.

4) ***Power Consumption****:* Dual-agent optimization achieves substantial savings: average transmit power of 9 dBm versus 15–20 *dBm* for static (55% reduction). Density-adaptive control limits power to 1–3 dBm under heavy load as depicted in Figure 11

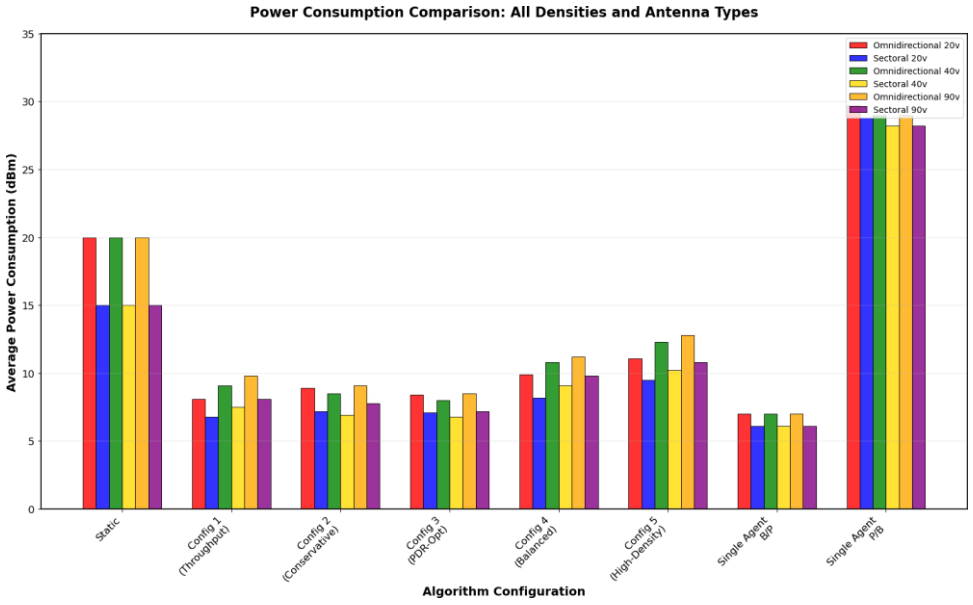

Figure 11. Power Consumption Across Configurations

Power reduction scales with density following the reward function's density-adaptive targets. At low density with 20-30 vehicles/km, 40% savings occur because path loss dominates and less power is required for connectivity. At high density with 90 vehicles/km, 55% savings result from aggressive interference management, where even small power reductions significantly improve the signal-to-interference ratio for other vehicles. The dual-agent approach reduces average power from 15-20 dBm to 9 dBm, with density-adaptive constraints limiting power to 1-3 dBm under heavy load to prevent channel saturation.

4-2- Signal Quality and Channel Utilization

Dual-agent control keeps CBR close to the target range (0.4–0.6) across all densities, while static reaches 0.6–0.8 at 90 vehicles. SINR values are lower than those of static (12–16 dB vs. 18.5–21.8 dB) but remain sufficient for reliable decoding under IEEE 802.11bd MCS mapping. These trends explain the observed latency gains and modest PDR trade-off as depicted in Figures 12 and 13.

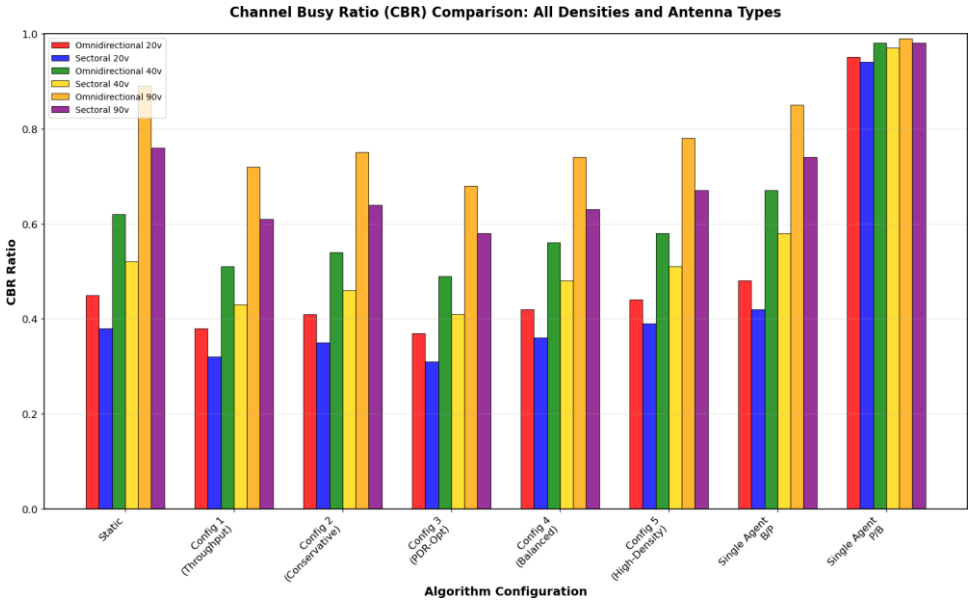

Figure 12. Channel Busy Ratio (CBR) Across Densities and Antenna Modes

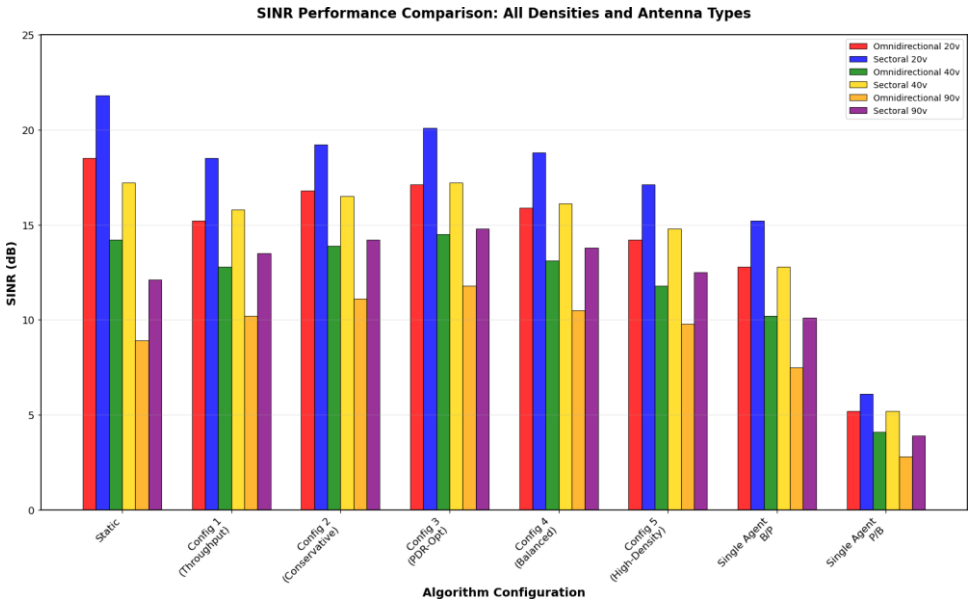

Figure 13. SINR Across Densities and Antenna Modes

Dual-agent control keeps CBR close to the target range 0.4–0.6 across all densities, while static reaches 0.6–0.8 at 90 vehicles. This CBR management directly enables the latency reduction, lower channel occupancy means less contention, and faster CSMA/CA resolution. SINR values are lower than static (12–16 dB vs. 18.5–21.8 dB) due to reduced transmit power but remain sufficient for reliable decoding under IEEE 802.11bd MCS mapping. The 12 dB minimum keeps the

system above the MCS-3 threshold by 10 dB, allowing successful packet reception while sacrificing margin for higher modulation schemes. This SINR-PDR relationship explains the modest 5-11% PDR trade-off: lower SINR increases packet error probability slightly but avoids excessive interference that would cause greater network-wide degradation.

4-3- Learning Behavior and Convergence

The dual-agent converges within 8,500 episodes (TD error 0.05), faster than single-agent Q-learning (12,500) and DQN (14k–35k). Training is smoother with fewer oscillations, producing stable policies consistent across runs, as depicted in Figure 14. These results align with the complexity reduction reported in Section VII.

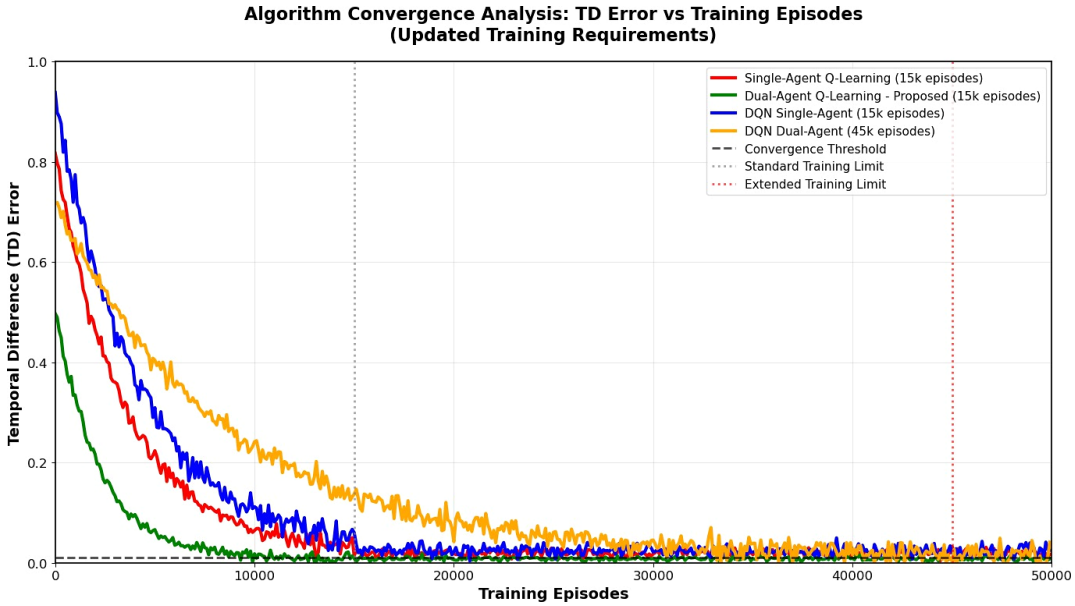

Figure 14. Convergence: TD Error vs. Training Episodes

Faster convergence occurs because separating PHY and MAC control reduces each agent's learning complexity and eliminates conflicting objective gradients. Single-agent approaches suffer from temporal mismatches: PHY and MAC operate on different timescales; PHY needs fast SINR stabilization, while MAC requires stable rate decisions. The dual-agent architecture naturally accommodates these different timescales, resulting in smoother training with fewer oscillations and stable policies that are consistent across runs.

4-4- Overall Performance and Rankings

Despite a five-point PDR reduction, the dual-agent achieves the highest overall score (83–87) by combining latency and power efficiency benefits. Sectoral configurations rank best (86.6) due to improved throughput, as depicted in Figure 15.

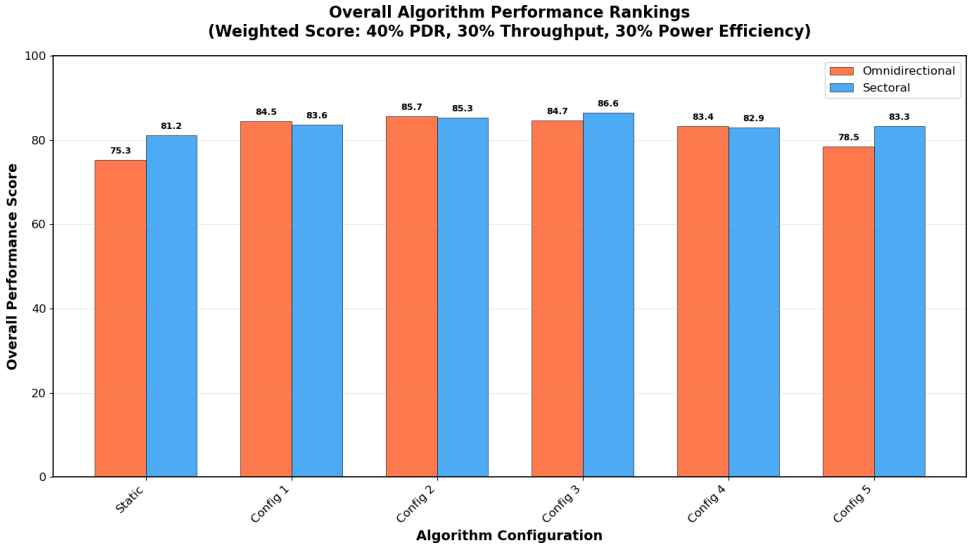

Figure 15. Overall Ranking Using Weighted Score (40% PDR, 30% Throughput, 30% Power Efficiency)

4-5-Comparison with Advanced DRL Baselines

Although this study primarily benchmarks against static, single-agent Q-learning and DQN baselines, we also consider results using advanced policy-gradient methods, such as PPO and SAC, for VANET parameter control. These approaches demonstrate strong adaptability in high-dimensional state spaces, enabling them to achieve competitive throughput and reliability. However, they also present drawbacks: (i) longer training budgets, often requiring far more episodes to converge; (ii) high computational demands from deep neural architectures; and (iii) reduced interpretability and weaker convergence guarantees, limiting their practicality for real-time safety-critical VANETs. In contrast, the proposed dual-agent tabular Q-learning converges within 8,500 episodes, produces stable policies with simple table lookups, and achieves favourable trade-offs between latency reduction (44.6%) and power efficiency (55%) while maintaining acceptable reliability (88–89% PDR). These results highlight that, while PPO and SAC are attractive in research contexts, the dual-agent tabular design offers a more lightweight and deployable solution for IEEE 802.11bd VANETs.

4-6-Antenna Configuration Impact

Sectoral setups outperform omni in throughput (+10–15%) and power efficiency (+2–3%) even under zero-gain. With a realistic 3-6 dB gain, dual-agent could reduce power consumption below five dBm while maintaining ≥85% PDR. This highlights the strong connection between algorithmic control and antenna design, and this comparison can be seen in Table 9.

Sectoral antennas outperform omnidirectional by 10-15% in throughput and 2-3% in power efficiency even under zero-gain conditions. This improvement stems from two factors. First, spatial selectivity reduces interference; the 90° beamwidth per sector focuses energy on relevant directions, front/rear for highway traffic, while minimizing interference to lateral vehicles. Second, the effective neighbor count is reduced by a factor of 0.3, meaning each vehicle sees fewer simultaneous transmitters, lowering collision probability. The dual-agent framework adapts to these antenna characteristics by using different RL learning rates (0.15 for omnidirectional, 0.18 for sectoral) and coordination weights (0.25 vs 0.30), tuned to match each antenna's interference profile. With realistic 3-6 dBi gain, link budget improvements could enable power reduction below five dBm while maintaining 85%+ PDR. However, this requires validation with actual antenna patterns rather than the uniform zero-gain used for fair algorithmic comparison.

4-7-Application Suitability

High-reliability applications (>90% PDR): Static configuration achieves 94.2% PDR and is suitable for applications requiring high message delivery. Latency-sensitive applications: Dual-agent reduces latency by 45% with 88.6% PDR, making it suitable for cooperative awareness and traffic coordination services. Power-constrained: Dual-agent's 55% savings benefit infrastructure nodes and large-scale deployments.

4-8-Standards Compliance

IEEE 802.11bd compliance is maintained through deterministic SINR–MCS mapping (Table 5). Over-aggressive selections occur <5% of the time, compared to 60–95% in unstable single-agent runs. This ensures interoperability in heterogeneous VANETs.

4-9-Limitations and Deployment Considerations

- Reliability trade-off: 4–11% PDR reduction, most severe at densities >70 vehicles.
- Implementation complexity: Retuning, coordinating, and training infrastructure.
- Training period: 8,500 episodes before convergence; performance is suboptimal during early learning.
- Scability limits: degradation accelerates beyond 70 vehicles, constraining dense-urban applicability.

4-10-Performance Comparison Against Existing Works

While direct quantitative comparisons are limited by fundamental differences in communication standards (802.11bd vs. 802.11p/cellular), simulation platforms, and experimental scenarios, we provide context from the recent literature. Tabular Q-learning approaches using 802.11p have achieved 88–89% PDR, although with a different parameter optimization scope [11]. Deep RL approaches have also reported strong results in their respective evaluation environments. The effectiveness of DNN-based methods for 802.11p has been demonstrated in Aznar-Poveda et al. [27]. Latency minimization for general V2V scenarios has been investigated in Ye et al. [49], achieving 90–95% satisfied links. A DDQNet-based approach reported 92.8% PDR and 12.5 ms latency [30], while an attention-enhanced MADRL framework achieved 92.8% PDR [60]. However, all of these methods require GPU-intensive deep neural networks. Federated multi-agent deep RL has been shown to improve spectrum efficiency by 19.1%, although it relies on

distributed infrastructure [28]. Similarly, multi-agent RL combined with LSTM-based prediction and digital twin models demonstrated approximately a 44% capacity improvement in IoV networks [29]. These approaches, while achieving strong performance in their evaluation contexts, require computational resources (GPU acceleration, prediction models, federated infrastructure) that exceed typical OBU capabilities. Our dual-agent tabular Q-learning achieves 88-89% PDR, 44.6% latency reduction, and 55% power savings (vs. our static baseline) with only 25 MB memory footprint, enabling immediate deployment on commercial OBUs without additional infrastructure. While cross-study numerical comparisons are not directly valid, our results demonstrate that practical deployment feasibility can be achieved while maintaining competitive performance characteristics within the 802.11bd evaluation context, as depicted in Table 10.

Table 10. Results Comparison with Previous Works

Study	PDR/Success Rate	Latency	Throughput / CBR	Method Complexity	Deployment Feasibility
Aznar-Poveda et al. [11]	88-89% PDR	44.6% reduction	CBR maintained	Low (tabular Q-learning, 23 MB)	High (Distributed, non-cooperative)
Ye et al. [49]	90-95% satisfied links	-	-	Low (DRL with Q-learning)	High (distributed)
Aznar-Poveda et al. [27]	90% PDR	-	CBR: 0.6-0.7	Low (DNN with DRL, single-agent)	High (distributed, non-cooperative)
Elloumi et al. [29]	-	-	~44% capacity increase	High (multi-agent RL + predictions)	Medium (needs LSTM, digital twin, clustering)
Cui [30]	92.8% PDR	12.5 ms	8.7 Mbps / 7.2% collision	Very High (DDQNet)	Low (requires significant computation)
Lui & Deng [60]	92.8% PDR	12.5 ms	8.7 Mbps / 7.2% collision	Very High (AMAFDRL)	Low (requires deep RL infrastructure)
Liu & Ma [28]	V2V: 9.3% improvement	V2I improved	19.1% spectrum efficiency	Very High (Federated MADDPG)	Low (requires federated infrastructure)
Our Algorithm (DAQN)	88-89% PDR	44.6% reduction	CBR maintained	Low (tabular Q-learning, 25 MB)	High (Distributed, semi-cooperative, OBU-deployable)

4-11-Deployment Recommendations

Dual-agent Q-learning is recommended for mixed-criticality VANETs that emphasize latency and power efficiency, particularly at medium vehicle densities (20–70 vehicles) and in infrastructure nodes with constrained energy. Table 11 summarizes the mean performance metrics across all configurations and densities. For purely safety-critical cases, static high-power remains essential. Hybrid deployments, static control for safety beacons, and dual-agent traffic management offer a pragmatic path forward. To mitigate the 5–11% PDR reduction, especially in high-density scenarios (>70 vehicles), hybrid strategies are recommended. One option is to reserve static power configurations for safety-critical beacons, while applying dual-agent Q-learning only to latency or power-sensitive traffic. Density-aware adjustments of the learning policy, such as more conservative exploration at extreme densities, can further reduce reliability degradation. These mechanisms allow deployments to retain latency and power efficiency benefits without compromising safety requirements.

Table 11. Summary Results Across Configurations (Mean Values, 20-90 Vehicles)

Metric	Static Omni	Static Sec-toral	Dual-Agent Omni	Dual-Agent Sectoral
PDR (%)	94.2	93.6	89.2	88.5
Latency (ms)	31.1	29.8	17.2	16.5
Throughput (Mbps)	10.35	10.7	9.6	10.2
Power (dBm)	15–20	15–20	9.0	8.7

5- Conclusion

Dense vehicular networks struggle to deliver safety messages reliably due to channel congestion and interference. We developed a dual-agent Q-learning framework that splits control between two specialized agents: one handles PHY-layer power, the other manages MAC-layer beacon rates. This separation reduces the problem size from 710 million to 6.5 million state-action pairs, making the system tractable while remaining fast enough for real-time vehicle control. Testing across different traffic densities (20 to 90 vehicles per km) shows clear benefits: latency drops by 45% (from 31.1ms to 17.2ms), power consumption falls by 55% (from 15-20 dBm to 9 dBm), and the system learns faster, converging in 8,500 training episodes compared to 12,500 for single-agent Q-learning and 14,000-35,000 for dual-agent DQN. The trade-off is a 5-11% drop in packet delivery ratio (from 94.2% to 88.6%), but this remains acceptable for applications where energy savings and low latency matter more than perfect reliability. The framework fully complies with IEEE 802.11bd standards through deterministic MCS selection, ensuring compatibility with existing equipment. Adding sectoral antennas improved throughput by 10-15% even with zero-gain test conditions, suggesting further gains

with real antenna patterns. However, the approach has limitations that constrain deployment. Packet delivery degrades notably when density exceeds 70 vehicles per km, making it unsuitable for critical safety applications without modifications. We tested with zero-gain antennas for a fair algorithmic comparison, but real antennas with 3-6 dBi gain would likely perform better, potentially reducing the required power to below five dBm. We have also not validated the simulator against actual IEEE 802.11bd hardware, so that real-world performance might differ from these simulation results, particularly regarding interference patterns and propagation characteristics in urban environments.

Several improvements could address current limitations and extend applicability. A hybrid system could switch strategies based on message importance, using dual-agent optimization for routine traffic while reserving high-power static transmission for emergency warnings, potentially achieving both efficiency and reliability. Learning could accelerate through transfer learning, where knowledge from one traffic scenario helps in another, or meta-reinforcement learning for rapid adaptation, potentially cutting the current 8,500-episode training requirement in half. In very dense scenarios with more than 70 vehicles per km, vehicles could cooperate by sharing learned policies or experience, helping with coordination when individual optimization is not enough. Looking further ahead, the framework could integrate with 6G-V2X systems, use semantic communication to reduce message overhead, and incorporate formal verification methods to prove safe behavior before deployment. Field tests with actual IEEE 802.11bd equipment are essential to validate simulation results and identify deployment-specific challenges, such as hardware constraints, real-world propagation conditions, and interoperability issues. This work shows that dual-agent Q-learning offers a practical way to optimize vehicular networks, though it is not perfect for every situation. It works best when we need low latency and power savings. For purely safety-critical messages requiring 95%+ delivery, static high-power systems remain necessary. No single approach fits all vehicular network needs, but dual-agent Q-learning fills an important gap for efficiency-focused applications where the reliability-efficiency trade-off aligns with operational requirements.

6- Declarations

6-1- Author Contributions

Conceptualization, G.N.N. and E.M.; methodology, G.N.N., A.S., and E.M.; software, G.N.N.; validation, S. and A.T.; formal analysis, G.N.N. and A.T.; investigation, G.N.N.; resources, S., A.S., and N.A.; data curation, A.T.; writing—original draft preparation, G.N.N.; writing—review and editing, S., R.M., A.S., N.A., and N.R.S.; visualization, G.N.N.; supervision, R.M., B.P., E.M., and N.R.S.; project administration, R.M., B.P., and N.R.S.; funding acquisition, N.A. All authors have read and agreed to the published version of the manuscript.

6-2- Data Availability Statement

The reinforcement-learning, deep reinforcement-learning script, CANVAS VANET simulation scripts used in this study, are available from the corresponding author upon request. SUMO mobility traces and configuration files were generated as part of the experiments and can be shared for academic research purposes.

6-3- Funding

The authors gratefully acknowledge the financial support provided by Telkom University and Bandung Institute of Technology (ITB), Indonesia. This research is funded by Telkom University, Hibah PPMI ITB, and the APNIC Foundation through the SOI Asia initiative.

6-4- Acknowledgments

The authors would like to thank the SOI Asia consortium, the APNIC Foundation, and participating universities for their collaboration and support. Special appreciation is given to the research teams at BRIN and STEI ITB for providing resources and technical assistance during the development and validation of the testbed.

6-5- Institutional Review Board Statement

Not applicable.

6-6- Informed Consent Statement

Not applicable.

6-7- Conflicts of Interest

The authors declare that there is no conflict of interest regarding the publication of this manuscript. In addition, the ethical issues, including plagiarism, informed consent, misconduct, data fabrication and/or falsification, double publication and/or submission, and redundancies have been completely observed by the authors.

7- References

[1] W.H.O. (2023). Despite Notable Progress, Road Safety Remains Urgent Global Issue. World Health Organization, Geneva, Switzerland. Available: Available online: https://www.who.int/news/item/13-12-2023-despite-notable-progress-road-safety-remains-urgent-global-issue (accessed on May 2026).

[2] Fu, Y., Li, C., Yu, F. R., Luan, T. H., & Zhang, Y. (2022). A Survey of Driving Safety with Sensing, Vehicular Communications, and Artificial Intelligence-Based Collision Avoidance. IEEE Transactions on Intelligent Transportation Systems, 23(7), 6142–6163. doi:10.1109/TITS.2021.3083927.

[3] Joerer, S., Segata, M., Bloessl, B., Cigno, R. Lo, Sommer, C., & Dressler, F. (2014). A vehicular networking perspective on estimating vehicle collision probability at intersections. IEEE Transactions on Vehicular Technology, 63(4), 1802–1812. doi:10.1109/TVT.2013.2287343.

[4] SAE Standard J2945/2_201810. (2018). Dedicated Short Range Communications (DSRC) Performance Requirements for V2V Safety Awareness. SAE International, Warrendale, United States. doi:10.4271/J2945/2_201810.

[5] Kenney, J. B. (2011). Dedicated short-range communications (DSRC) standards in the United States. Proceedings of the IEEE, 99(7), 1162–1182. doi:10.1109/JPROC.2011.2132790.

[6] IEEE Computer Society LAN/MAN Standards Committee. (2009). IEEE Standard for Information Technology-Telecommunication and Information Exchange between Systems-Local and Metropolitan Area Networks-Specific Requirements Part11: Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specifications Amendment1: Radio Resource Measurement of Wireless LANs, 1-148. Available online: http://standards.ieee.org/getieee802/download/802.11n-2009.pdf (accessed on May 2026).

[7] Maaloul, S., Aniss, H., Kassab, M., & Berbineau, M. (2021). Classification of C-ITS Services in Vehicular Environments. IEEE Access, 9, 117868–117879. doi:10.1109/ACCESS.2021.3105815.

[8] Molina-Masegosa, R., Gozalvez, J., & Sepulcre, M. (2020). Comparison of IEEE 802.11p and LTE-V2X: An Evaluation with Periodic and Aperiodic Messages of Constant and Variable Size. IEEE Access, 8, 121526–121548. doi:10.1109/ACCESS.2020.3007115.

[9] Iliopoulos, C., Iossifides, A., Foh, C. H., & Chatzimisios, P. (2025). IEEE 802.11BD for Next-Generation V2X Communications: From Protocol to Services. IEEE Communications Standards Magazine, 9(2), 88–98. doi:10.1109/MCOMSTD.2025.3569015.

[10] Kumar, S., Kumar, A., Tyagi, V., & Kumar, A. (2020). Impact of Network Density on AODV protocol in VANET. 2020 IEEE 5th International Conference on Computing Communication and Automation, ICCCA 2020, 559–564. doi:10.1109/ICCCA49541.2020.9250898.

[11] Aznar-Poveda, J., Garcia-Sanchez, A. J., Egea-Lopez, E., & Garcia-Haro, J. (2021). MDPRP: A Q-Learning Approach for the Joint Control of Beaconing Rate and Transmission Power in VANETs. IEEE Access, 9, 10166–10178. doi:10.1109/ACCESS.2021.3050625.

[12] Triwinarko, A., Dayoub, I., & Cherkaoui, S. (2021). PHY layer enhancements for next generation V2X communication. Vehicular Communications, 32, 100385. doi:10.1016/j.vehcom.2021.100385.

[13] Kaul, S., Ramachandran, K., Shankar, P., Oh, S., Gruteser, M., Seskar, I., & Nadeem, T. (2007). Effect of antenna placement and diversity on vehicular network communications. 2007 4th Annual IEEE Communications Society Conference on Sensor, Mesh and Ad Hoc Communications and Networks, 112-121. doi:10.1109/SAHCN.2007.4292823.

[14] Xie, X., Huang, B., Yang, S., & Lv, T. (2009). Adaptive multi-channel MAC protocol for dense VANET with directional antennas. In 2009 6th IEEE Consumer Communications and Networking Conference, 1-5. doi:10.1109/CCNC.2009.4784948.

[15] Ren, J., Zhang, G., & Li, D. (2017). Multicast Capacity for VANETs with Directional Antenna and Delay Constraint under Random Walk Mobility Model. IEEE Access, 5, 3958–3970. doi:10.1109/ACCESS.2017.2683718.

[16] Subramanian, A. P., Navda, V., Deshpande, P., & Das, S. R. (2008). A measurement study of inter-vehicular communication using steerable beam directional antenna. Proceedings of the Fifth ACM International Workshop on VehiculAr Inter-NETworking, 7–16. doi:10.1145/1410043.1410046.

[17] Li, H., & Xu, Z. (2018). Routing Protocol in VANETs Equipped with Directional Antennas: Topology-Based Neighbor Discovery and Routing Analysis. Wireless Communications and Mobile Computing, 2018(1), 7635143. doi:10.1155/2018/7635143.

[18] Yanbin, W., Zhuofei, W., Jing, Z., Zhijuan, L., & Xiaomin, M. (2020). Analysis and adaptive optimization of vehicular safety message communications at intersections. Ad Hoc Networks, 107, 102241. doi:10.1016/j.adhoc.2020.102241.

[19] Sepulcre, M., Gozalvez, J., & Miralles, H. (2019). Context-Aware Beaconing for Cooperative Awareness in Vehicular Networks. IEEE Transactions on Intelligent Transportation Systems, 20(2), 726–740. doi:10.1109/TITS.2018.2853644.

[20] Ma, X., & Trivedi, K. S. (2021). SINR-Based Analysis of IEEE 802.11p/bd Broadcast VANETs for Safety Services. IEEE Transactions on Network and Service Management, 18(3), 2672–2686. doi:10.1109/TNSM.2021.3069206.

[21] Popovski, P., Stefanovic, C., Nielsen, J. J., de Carvalho, E., Angjelichinoski, M., Trillingsgaard, K. F., & Bana, A.-S. (2019). Wireless Access in Ultra-Reliable Low-Latency Communication (URLLC). IEEE Transactions on Communications, 67(8), 5783–5801. doi:10.1109/tcomm.2019.2914652.

[22] Chang, H., Song, Y. E., Kim, H., & Jung, H. (2018). Distributed transmission power control for communication congestion control and awareness enhancement in VANETs. PLoS ONE, 13(9), 203261. doi:10.1371/journal.pone.0203261.

[23] Jain, A., Mehrotra, A., Rewariya, A., & Kumar, S. (2022). A Systematic Study of Deep Q-Networks and Its Variations. 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering, ICACITE 2022, 2157–2162. doi:10.1109/ICACITE53722.2022.9823631.

[24] Gu, Y., Cheng, Y., Chen, C. L. P., & Wang, X. (2022). Proximal Policy Optimization with Policy Feedback. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 52(7), 4600–4610. doi:10.1109/TSMC.2021.3098451.

[25] Haarnoja, T., Zhou, A., Abbeel, P., & Levine, S. (2018). Soft actor-critic: Off-policy maximum entropy deep reinforcement learning with a stochastic actor. 35th International Conference on Machine Learning, ICML 2018, 5, 2976–2989.

[26] Galliera, R., Morelli, A., Fronteddu, R., & Suri, N. (2023). MARLIN: Soft Actor-Critic based Reinforcement Learning for Congestion Control in Real Networks. Proceedings of IEEE/IFIP Network Operations and Management Symposium 2023, NOMS 2023, 1–10. doi:10.1109/NOMS56928.2023.10154210.

[27] Aznar-Poveda, J., Garcia-Sanchez, A. J., Egea-Lopez, E., & Garcia-Haro, J. (2021). Simultaneous Data Rate and Transmission Power Adaptation in V2V Communications: A Deep Reinforcement Learning Approach. IEEE Access, 9, 122067–122081. doi:10.1109/ACCESS.2021.3109422.

[28] Liu, Q., & Ma, Y. (2025). Communication resource allocation method in vehicular networks based on federated multi-agent deep reinforcement learning. Scientific Reports, 15(1). doi:10.1038/s41598-025-15982-x.

[29] Elloumi, M., Hassan, Z. Z., & Kaddoum, G. (2025). Spectrum Sharing in Internet-of-Vehicles Networks: Digital Twin-Empowered Proactive Interference Management Approach. IEEE Transactions on Network and Service Management, 22(4), 3228–3248. doi:10.1109/TNSM.2025.3541977.

[30] Cui, J. (2025). A Deep Reinforcement Learning Approach for Dynamic Resource Allocation in VANETs with Human–Centric Interaction Interfaces. Transactions on Emerging Telecommunications Technologies, 36(8), e70221. doi:10.1002/ett.70221.

[31] Commsignia. (2024). OBU Lite - Powerful V2X Onboard Unit. Commsignia, Budapest, Hungary. Available online: https://commsignia.com/products/obu (accesed on May 2026).

[32] Ajeevi Technologies. (2024). On-Board Unit (AJV-IOT-OBU-001). Ajeevi Technologies, Noida, India. Available online: https://ajeevi.com/wp-content/uploads/2024/02/OBU.docx.pdf (accessed on May 2026).

[33] Jang, B., Kim, M., Harerimana, G., & Kim, J. W. (2019). Q-Learning Algorithms: A Comprehensive Classification and Applications. IEEE Access, 7, 133653–133667. doi:10.1109/ACCESS.2019.2941229.

[34] Sarris, I. (2018). ubx-v2x. GitHub, Inc, San Francisco, United States. Available online: https://github.com/u-blox/ubx-v2x (accessed on May 2026).

[35] Turcanu, I., Salvo, P., Baiocchi, A., Cuomo, F., & Engel, T. (2020). A multi-hop broadcast wave approach for floating car data collection in vehicular networks. Vehicular Communications, 24, 100232. doi:10.1016/j.vehcom.2020.100232.

[36] ETSI TR 102 638. (2011). Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Definitions. European Telecommunications Standards Institute, Sophia, France. Available online: https://www.etsi.org/deliver/etsi_tr/102600_102699/102638/01.01.01_60/tr_102638v010101p.pdf (accessed on May 2026).

[37] Abboud, K., Omar, H. A., & Zhuang, W. (2016). Interworking of DSRC and Cellular Network Technologies for V2X Communications: A Survey. IEEE Transactions on Vehicular Technology, 65(12), 9457–9470. doi:10.1109/TVT.2016.2591558.

[38] CSN ETSI EN 302 637-3 V1.3.1. (2019). Intelligent Transport Systems (ITS); Vehicular Communications; Basic Set of Applications; Part 3: Specifications of Decentralized Environmental Notification Basic Service. European Standard, Brussels, Belgium.

[39] Su, Z., Xu, Q., & Qi, Q. (2016). Big data in mobile social networks: A QoE-oriented framework. IEEE Network, 30(1), 52–57. doi:10.1109/MNET.2016.7389831.

[40] Yang, H., Zheng, K., Zhang, K., Mei, J., & Qian, Y. (2020). Ultra-Reliable and Low-Latency Communications for Connected Vehicles: Challenges and Solutions. IEEE Network, 34(3), 92–100. doi:10.1109/MNET.011.1900242.

[41] Miao, Z., Li, C., Zhu, L., Han, X., Wang, M., Cai, X., Liu, Z., & Xiong, L. (2016). On resource management in vehicular Ad Hoc networks: A fuzzy optimization scheme. IEEE Vehicular Technology Conference, 2016-July, 1–5. doi:10.1109/VTCSpring.2016.7504373.

[42] Haitao, Z., Yuting, Z., Hongbo, Z., & Dapeng, L. (2018). Resource Management in Vehicular Ad Hoc Networks: Multi-parameter Fuzzy Optimization Scheme. Procedia Computer Science, 129, 443–448. doi:10.1016/j.procs.2018.03.022.

[43] Goudarzi, F., Asgari, H., & Al-Raweshidy, H. S. (2019). Fair and stable joint beacon frequency and power control for connected vehicles. Wireless Networks, 25(8), 4979–4990. doi:10.1007/s11276-019-02076-6.

[44] Kapade, N. (2015). TLC: Trust Point Load Balancing Method using Coalitional Game Theory for message forwarding in VANET. Proceedings - 2014 IEEE Global Conference on Wireless Computing and Networking, GCWCN 2014, 160–164. doi:10.1109/GCWCN.2014.7030870.

[45] Cho, B. M., Jang, M. S., & Park, K. J. (2020). Channel-Aware Congestion Control in Vehicular Cyber-Physical Systems. IEEE Access, 8, 73193–73203. doi:10.1109/ACCESS.2020.2987416.

[46] Wei, L. J., & Lim, J. M. Y. (2019). Identifying Transmission Opportunity through Transmission Power and Bit Rate for Improved VANET Efficiency. Mobile Networks and Applications, 24(5), 1630–1638. doi:10.1007/s11036-018-1180-2.

[47] Aygun, B., Boban, M., & Wyglinski, A. M. (2016). ECPR: Environment-and context-aware combined power and rate distributed congestion control for vehicular communications. Computer Communications, 93, 3–16. doi:10.1016/j.comcom.2016.05.015.

[48] Triwinarko, A., Dayoub, I., Zwingelstein-Colin, M., Gharbi, M., & Bouraoui, B. (2020). A PHY/MAC cross-layer design with transmit antenna selection and power adaptation for receiver blocking problem in dense VANETs. Vehicular Communications, 24, 100233. doi:10.1016/j.vehcom.2020.100233.

[49] Ye, H., Li, G. Y., & Juang, B. H. F. (2019). Deep Reinforcement Learning Based Resource Allocation for V2V Communications. IEEE Transactions on Vehicular Technology, 68(4), 3163–3173. doi:10.1109/TVT.2019.2897134.

[50] Tian, J., An, S. H., Islam, A., & Chang, K. H. (2023). A Hybrid Power-Rate Management Strategy in Distributed Congestion Control for 5G-NR-V2X Sidelink Communications. Sensors, 23(15). doi:10.3390/s23156657.

[51] Egea-Lopez, E. (2016). Fair distributed Congestion Control with transmit power for vehicular networks. 2016 IEEE 17th International Symposium on a World of Wireless, Mobile and Multimedia Networks (WoWMoM), 1–6. doi:10.1109/WoWMoM.2016.7523571.

[52] Caizzone, G., Giacomazzi, P., Musumeci, L., & Verticale, G. (2005). A power control algorithm with high channel availability for vehicular ad hoc networks. IEEE International Conference on Communications, 5, 3171–3176. doi:10.1109/icc.2005.1494999.

[53] Chaab, W. Al, Ismail, M., Altahrawi, M. A., Mahdi, H., & Ramli, N. (2017). Efficient rate adaptation algorithm in high-dense vehicular ad hoc network. 2017 IEEE 13th Malaysia International Conference on Communications, MICC 2017, 2017-November, 23–28. doi:10.1109/MICC.2017.8311725.

[54] Tielert, T., Jiang, D., Chen, Q., Delgrossi, L., & Hartenstein, H. (2011). Design methodology and evaluation of rate adaptation based congestion control for vehicle safety communications. IEEE Vehicular Networking Conference, VNC, 116–123. doi:10.1109/VNC.2011.6117132.

[55] Amer, H., Al-Kashoash, H., Khami, M. J., Mayfield, M., & Mihaylova, L. (2020). Non-cooperative game based congestion control for data rate optimization in vehicular ad hoc networks. Ad Hoc Networks, 107, 102181. doi:10.1016/j.adhoc.2020.102181.

[56] Mande, S., Ramachandran, N., Salma Asiya Begum, S., & Moreira, F. (2024). Optimized Reinforcement Learning for Resource Allocation in Vehicular Ad Hoc Networks. IEEE Access, 12, 167040–167048. doi:10.1109/ACCESS.2024.3489395.

[57] Jacob, M., Gopika, S., Ravindran, D., & Veerachamy, V. (2025). Implementation of Reinforcement Learning-Optimized Communication Protocols for VANETs: Challenges and Solutions. Advances in Communication and Applications, ERCICA 2024. Lecture Notes in Electrical Engineering, vol 1398, Springer, Singapore. doi:10.1007/978-981-96-4679-1_19.

[58] Ramesh, S. S. S., Banu, J. F., Kavitha, V. R., & Ramesh, T. (2025). Enhancing Intelligent Transportation Systems in Smart Cities Using VANETs With Deep Reinforcement Transfer Learning and Explainable AI. Transactions on Emerging Telecommunications Technologies, 36(8), e70219. doi:10.1002/ett.70219.

[59] Kai, C., & Liang, S. (2025). Control Strategy for VANET Autonomous Driving Vehicles in Emergency Situations Based on Deep Learning. Transactions on Emerging Telecommunications Technologies, 36(12), e70302. doi:10.1002/ett.70302.

[60] Liu, Z., & Deng, Y. (2025). Resource allocation strategy for vehicular communication networks based on multi-agent deep reinforcement learning. Vehicular Communications, 53. doi:10.1016/j.vehcom.2025.100895.

[61] Akinlade, O. (2018). Adaptive transmission power with vehicle density for congestion control. Master Thesis, University of Windsor, Windsor, Canada.

[62] Bansal, G., Kenney, J. B., & Rohrs, C. E. (2013). LIMERIC: A linear adaptive message rate algorithm for DSRC congestion control. IEEE Transactions on Vehicular Technology, 62(9), 4182–4197. doi:10.1109/TVT.2013.2275014.

[63] Aznar-Poveda, J., Egea-Lopez, E., Garcia-Sanchez, A. J., & Pavon-Marino, P. (2019). Time-to-collision-based awareness and congestion control for vehicular communications. IEEE Access, 7, 154192–154208. doi:10.1109/ACCESS.2019.2949131.

[64] Abdolahi, F., Mišić, J., & Mišić, V. B. (2025). Aligning Priorities: Interconnecting Vehicular Cloud Using IEEE 802.11bd Communications. IEEE International Conference on Communications, 5431–5436. doi:10.1109/ICC52391.2025.11162117.

[65] Yacheur, B. Y., Ahmed, T., & Mosbah, M. (2020). Analysis and Comparison of IEEE 802.11p and IEEE 802.11bd. Communication Technologies for Vehicles. Nets4Cars/Nets4Trains/Nets4Aircraft 2020. Lecture Notes in Computer Science, Volume 12574, Springer, Cham, Switzerland. doi:10.1007/978-3-030-66030-7_5.

[66] Ehsanfar, S., Moessner, K., Gizzini, A. K., & Chafii, M. (2022). Performance Comparison of IEEE 802.11p, 802.11bd-draft and a Unique-Word-based PHY in Doubly-Dispersive Channels. IEEE Wireless Communications and Networking Conference, WCNC, 2022-April, 1815–1820. doi:10.1109/WCNC51071.2022.9771810.

[67] Ye, H., Li, G. Y., & Juang, B.-H. (2019). Deep Reinforcement Learning for V2V Communications with Dynamic Vehicle Environments. IEEE Transactions on Vehicular Technology, 68(4), 3163–3173. doi:10.1109/TVT.2019.2896055.

[68] Haider, A., & Hwang, S. H. (2019). Adaptive transmit power control algorithm for sensing-based semi-persistent scheduling in C-V2X mode 4 communication. Electronics (Switzerland), 8(8), 846. doi:10.3390/electronics8080846.

[69] Joseph, M., Liu, X., & Jaekel, A. (2018). An adaptive power level control algorithm for DSRC congestion control. DIVANet 2018 - Proceedings of the 8th ACM Symposium on Design and Analysis of Intelligent Vehicular Networks and Applications, 57–62. doi:10.1145/3272036.3272041.

[70] Aslani, R., & Rasti, M. (2020). A Distributed Power Control Algorithm for Energy Efficiency Maximization in Wireless Cellular Networks. IEEE Wireless Communications Letters, 9(11), 1975–1979. doi:10.1109/LWC.2020.3010156.

[71] Wang, M., Chen, T., Du, F., Wang, J., Yin, G., & Zhang, Y. (2022). Research on adaptive beacon message transmission power in VANETs. Journal of Ambient Intelligence and Humanized Computing, 13(3), 1307–1319. doi:10.1007/s12652-020-02575-x.

[72] Shwetha, A., & Sankar, P. (2018). Queue management scheme to control congestion in a vehicular based sensor network. 2018 2nd International Conference on Inventive Systems and Control (ICISC), 917–921. doi:10.1109/ICISC.2018.8398933.

[73] Tayyaba, S. K., Khattak, H. A., Almogren, A., Shah, M. A., Ud Din, I., Alkhalifa, I., & Guizani, M. (2020). 5G vehicular network resource management for improving radio access through machine learning. IEEE Access, 8, 6792–6800. doi:10.1109/ACCESS.2020.2964697.

[74] Eckhoff, D., Brummer, A., & Sommer, C. (2016). On the impact of antenna patterns on VANET simulation. IEEE Vehicular Networking Conference, VNC, 0, 1–4. doi:10.1109/VNC.2016.7835925.

[75] F.H.A. (2018). Traffic Data Computation Method Pocket Guide. FHWA-PL-18-027, Federal Highway Administration (F.H.A.), Washington, USA. Available online: https://www.fhwa.dot.gov/policyinformation/pubs/pl18027_traffic_data_pocket_guide.pdf (accessed on May 2026).

[76] Dharsandiya, A. N., & Patel, R. M. (2016). A review on MAC protocols of Vehicular Ad Hoc Networks. 2016 International Conference on Wireless Communications, Signal Processing and Networking (WiSPNET), 1040–1045. doi:10.1109/WiSPNET.2016.7566295.

[77] Rappaport, T. S. (2010). Wireless communications: Principles and practice, 2/E. Pearson Education India, Bengaluru, India.

[78] Ng, A. Y., Harada, D., & Russell, S. (1999). Policy invariance under reward transformations: Theory and application to reward shaping. Proceedings of the Sixteenth International Conference on Machine Learning (ICML 1999), 27-30 June, Bled, Slovenia.

[79] Sutton, R.S. and Barto, A.G. (2018) Reinforcement Learning: An Introduction. MIT Press, Cambridge, United States.

[80] Tsitsiklis, J. N., & Van Roy, B. (1997). An analysis of temporal-difference learning with function approximation. IEEE Transactions on Automatic Control, 42(5), 674–690. doi:10.1109/9.580874.

[81] Buşoniu, L., Babuška, R., & De Schutter, B. (2008). A comprehensive survey of multiagent reinforcement learning. IEEE Transactions on Systems, Man and Cybernetics Part C: Applications and Reviews, 38(2), 156-172. doi:10.1109/TSMCC.2007.913919.

[82] Mnih, V., Kavukcuoglu, K., Silver, D., Rusu, A. A., Veness, J., Bellemare, M. G., Graves, A., Riedmiller, M., Fidjeland, A. K., Ostrovski, G., Petersen, S., Beattie, C., Sadik, A., Antonoglou, I., King, H., Kumaran, D., Wierstra, D., Legg, S., & Hassabis, D. (2015). Human-level control through deep reinforcement learning. Nature, 518(7540), 529–533. doi:10.1038/nature14236.

[83] Hessel, M., Modayil, J., Van Hasselt, H., Schaul, T., Ostrovski, G., Dabney, W., Horgan, D., Piot, B., Azar, M., & Silver, D. (2018). Rainbow: Combining improvements in deep reinforcement learning. 32nd AAAI Conference on Artificial Intelligence, AAAI 2018, 3215–3222. doi:10.1609/aaai.v32i1.11796.

[84] Baccour, E., Mhaisen, N., Abdellatif, A. A., Erbad, A., Mohamed, A., Hamdi, M., & Guizani, M. (2022). Pervasive AI for IoT Applications: A Survey on Resource-Efficient Distributed Artificial Intelligence. IEEE Communications Surveys and Tutorials, 24(4), 2366–2418. doi:10.1109/COMST.2022.3200740.

[85] Gill, S. S., Golec, M., Hu, J., Xu, M., Du, J., Wu, H., Walia, G. K., Murugesan, S. S., Ali, B., Kumar, M., Ye, K., Verma, P., Kumar, S., Cuadrado, F., & Uhlig, S. (2025). Edge AI: A Taxonomy, Systematic Review and Future Directions. Cluster Computing, 28(1), 18. doi:10.1007/s10586-024-04686-y.

[86] Bo, J., & Zhao, X. (2025). Vehicle Edge Computing Task Offloading Strategy Based on Multi-Agent Deep Reinforcement Learning. Journal of Grid Computing, 23(2), 13. doi:10.1007/s10723-025-09800-x.

[87] Tian, H., Zhu, L., & Tan, L. (2025). A joint task caching and computation offloading scheme based on deep reinforcement learning. Peer-to-Peer Networking and Applications, 18(1), 1–19. doi:10.1007/s12083-024-01836-2.

[88] Agarwal, R., Schwarzer, M., Castro, P. S., Courville, A. C., & Bellemare, M. (2021). Deep reinforcement learning at the edge of the statistical precipice. Advances in Neural Information Processing Systems, 34, 29304-29320.