International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 10 No. 4, 2024
https://www.theaspd.com/ijes.php

Adaptive Road-Aware Routing with Reinforcement Learning (ARARL) for Enhanced Efficiency and Reliability in Dense Urban Vanets

Arvind Kumar[1], Shobha Tyagi[2], Prashant Dixit[3], S.S. Tyagi[4]
[1,2]Manav Rachna International Institute of Research Studies, Faridabad, India,
er.kumararvind@gmail.com , tyagishobha.set@mriu.edu.in
[3]Galgotias University, Greater Noida, India, Prashantdixit@galgotiasuniversity.edu.in
[4]Gurugram University, Gurugram, India, shyamtyagi@gurugramuniversity.ac.in

Abstract
Vehicular ad hoc networks (VANETs) hold a prime role in smart transportation, but routing of data in busy cities is challenging. Scenarios like traffic jams, fast-moving vehicles, and signal disruptions due to physical obstacles often cause data packet dropping, resulting in regular disconnections in communication. Through this research paper, we introduce ARARL (Adaptive Road-Aware Routing with Reinforcement Learning), a Reinforcement Learning based approach to make VANETs more reliable. ARARL use reinforcement learning to select the best data paths, thus adapting to real-time changes like vehicle speeds or road layouts. Unlike static protocols, ARARL keep on learning with the vehicle movement and choose the best possible routes based on factors like signal strength and traffic flow. We tested ARARL in simulation using NS2 and OpenGym on a city scenario generated by SUMO. In results, it outperformed protocols like AODV, GPSR, D-LAR and Q-Learning-AODV. It transferred more packets, reduced delays and network overhead, especially when the network had a lot of traffic. These results suggest ARARL performs better in keeping communication steady. By ensuring the information is shared reliably, our work could help make self-driving cars safer.
Keywords: *VANETs (Vehicular Ad Hoc Networks), Reinforcement Learning, Roadside Units (RSUs), Routing Protocols, Urban Mobility.*

INTRODUCTION

Vehicular Ad Hoc Networks (VANETs) are a type of Mobile Ad Hoc Network (MANET) that enable vehicles to communicate with each other and other things (V2X), such as traffic signals and roadside sensors. This technology makes automated transportation possible by making roads safer, traffic smoother, and supporting novel autonomous driving applications. VANETs have many important applications that can fix some of the most significant issues with transportation nowadays. Safety reasons are the most essential thing to have. Vehicles can send out real-time notifications about hazards such as unexpected braking, eventual crashes, or dangerous road conditions. This makes driving safer overall by reducing the response times, decreasing the chance of accidents, and thus making the roads safer. Intelligent traffic management systems allow vehicles to communicate real-time information on road conditions and traffic congestion. This can helps drivers find the best possible routes to their destination. VANETs can also help in making infotainment services available on the go, which can let travelers utilize location-based services, internet, and entertainment facilities. Also, tasks like automated toll collection and people park their vehicles at a suitable place, becomes quicker and easier for everyone.

Because vehicular networks are always changing, effective routing is the most important part of VANET functionality. Strong routing protocols make sure that communication is reliable and has low latency, while also making the most of network resources and making it easier to add more users. Routing efficiency is what makes VANETs work well. This directly affects their ability to send safety-critical messages, control traffic flow, and provide extra services. As routing technologies improve, VANETs have the potential to change the way we travel by making it safer, more efficient, and more technologically advanced.

But routing in VANETs is very hard because of the unique features of vehicular environments. One of the main problems is high mobility; vehicles moving quickly cause the network topology to change quickly, which leads to links being disconnected often. AODV (Ad-hoc On-Demand Distance Vector) and DSR (Dynamic Source Routing) are two examples of traditional protocols that are made for more static MANETs. They have trouble keeping routes stable in these situations. The changing shape of VANETs makes routing even harder because cars are always moving, which means routing tables need to be updated often, which adds to the overhead and latency. Buildings, tunnels, and natural features are some of the physical barriers that make it harder to communicate because they block wireless signals. Protocols need to be able to handle these interruptions, which often means sending data through intermediate vehicles or infrastructure. Many current methods don't have this ability. Scalability is another problem. In cities, where there are more cars, protocols have to handle more data without getting too busy or delayed. Security is just as important, especially for safety apps. VANETs are at risk from threats like data interception, spoofing, and denial-of-service attacks. However, many protocols don't have strong ways to protect data integrity and node authentication without slowing down performance.

Routing protocols that are in use now have clear problems. Proactive protocols like OLSR (Optimized Link State Routing) and DSDV (Destination-Sequenced Distance-Vector) keep full routing tables, but they cost a lot of time and effort to keep up with in dynamic VANETs. AODV and DSR are examples of reactive protocols that set up routes on demand. This cuts down on overhead but adds delays when finding a route, which can be a problem in environments that change quickly. Hybrid protocols, like ZRP (Zone Routing Protocol), try to find a middle ground between these two methods, but they have trouble managing zones in networks that are very mobile. Geographic protocols like GPSR (Greedy Perimeter Stateless Routing) use location data, but they don't work well when GPS errors or physical barriers get in the way of communication.

To solve these problems, we need new routing protocols that are specific to VANETs. New solutions might use machine learning to guess traffic patterns and make better routing decisions, context-aware mechanisms to take environmental factors into account, and cross-layer designs to make communication more efficient. These kinds of improvements would let protocols change dynamically to handle high mobility, handle changes in topology, reduce disruptions caused by obstacles, scale well, and make sure that data is sent safely. Next-generation routing protocols will be able to use VANETs to their full potential by getting past these problems. This will lead to reliable communication and the development of smart, safe, and efficient transportation systems.

LITERATURE SURVEY

Adaptive Road-Aware Routing with Reinforcement Learning (ARARL) is a cutting-edge approach designed to enhance the efficiency and reliability of routing in dense urban Vehicular Ad Hoc Networks (VANETs). VANETs are critical for intelligent transportation systems (ITS), enabling real-time communication between vehicles and infrastructure to improve traffic management, road safety, and driver experiences. However, the dynamic and complex nature of urban environments poses significant challenges, such as high node mobility, frequent network topology changes, and the need for low-latency communication [3] [7]. ARARL addresses these challenges by integrating reinforcement learning (RL) with road-aware routing strategies, leveraging real-time data and adaptive decision-making to optimize routing paths and ensure reliable data transmission. Urban VANETs operate in highly dynamic environments where vehicles move at varying speeds, and network connections are frequently disrupted due to obstacles, signal interference, and rapid topology changes [3] [7]. Traditional routing protocols often struggle to adapt to these conditions, leading to inefficiencies in packet delivery, increased latency, and reduced network reliability [14]. Additionally, the density of vehicles in urban areas exacerbates these issues, as the network must handle a large number of nodes and maintain stable communication links [15]. To address these challenges, ARARL employs a combination of road-aware routing and reinforcement learning, enabling the network to dynamically adjust routing decisions based on real-time feedback from the environment.

Reinforcement learning has emerged as a powerful tool for solving complex routing problems in dynamic networks. By treating the routing process as a sequential decision-making problem, RL algorithms can learn optimal routing policies that maximize rewards, such as minimizing delay, reducing congestion, and improving packet delivery rates [7] [14]. In the context of ARARL, the RL agent learns to select the best next-hop nodes or paths by interacting with the environment and receiving feedback in the form of rewards or penalties. For example, in the Q-learning-based routing algorithm proposed in [17], the agent learns to select reliable next-hop nodes by estimating link reliability and maximizing Q-values based on a reward function. Similarly, the MADDPG model in [3] combines multi-agent reinforcement learning with rerouting techniques to improve traffic performance in urban networks. The performance of ARARL is significantly enhanced through several key mechanisms. First, the use of road-aware routing ensures that

the algorithm takes into account the physical layout of the urban environment, such as the location of intersections, traffic signals, and road segments [14] [15]. This allows the routing decisions to be more informed and context-aware, reducing the likelihood of selecting paths that are prone to congestion or frequent disruptions. Second, the integration of reinforcement learning enables the algorithm to adapt to changing network conditions in real-time, learning from past experiences and improving its decision-making over time [7] [17]. Finally, the use of advanced techniques such as experience replay and asynchronous learning helps to stabilize the training process and improve the convergence speed of the RL model [8] [9].
One of the key advantages of ARARL is its ability to maintain high reliability and scalability in dense urban environments. By leveraging the principles of multi-agent reinforcement learning, ARARL can effectively coordinate routing decisions across multiple nodes and agents, ensuring that the network operates efficiently even in the presence of high node density and rapid topology changes [3] [14]. Additionally, the algorithm's ability to adapt to real-time feedback allows it to dynamically adjust routing paths in response to changes in traffic conditions, such as accidents, road closures, or sudden increases in vehicle density [7] [15]. This adaptability not only improves the reliability of data transmission but also enhances the overall performance of the network, making it more suitable for large-scale urban deployments. While ARARL represents a significant advancement in routing for urban VANETs, there are several potential enhancements and future directions that could further improve its performance. One promising area of research is the integration of additional data sources, such as real-time traffic information, weather conditions, and road maintenance updates, to provide even more context-aware routing decisions [5] [12]. Another area of exploration is the use of advanced reinforcement learning architectures, such as deep deterministic policy gradients (DDPG) or asynchronous advantage actor-critic (A3C), to improve the scalability and convergence speed of the RL model [8] [9]. Finally, the incorporation of edge computing and distributed learning techniques could enable more efficient processing and decision-making at the edge of the network, reducing latency and improving overall system performance [4] [18].
A Cluster-based Trustworthy Safe Multipath Routing (CTSMP-Routing) for MANETs addresses load balancing using Modified Proportional Topology Optimisation (MPTO) and computes node trust via Enhanced Seeker Search Optimisation (ESSO). Multi-Layer Deep Recurrent Neural Network (ML-DRNN) selects optimal paths. Results show CTSMP-Routing enhances security against attacks and improves quality of service performance [19].

MOTIVATION FOR AI-BASED ROUTING

Vehicular Ad Hoc Networks (VANETs) enable dynamic communication among vehicles and infrastructure, supporting critical applications like collision avoidance and traffic optimization. However, their rapidly changing topologies, driven by vehicle mobility and urban obstacles, challenge traditional routing protocols. Artificial Intelligence (AI), particularly through machine learning (ML) and reinforcement learning (RL), offers a robust framework to address these complexities, surpassing conventional methods. Below, we explore how AI enhances VANET routing.
First, AI excels in adapting to dynamic environments. Traditional protocols, such as AODV or GPSR, rely on static or slowly updating mechanisms, often failing to keep pace with VANETs' frequent topology changes. This leads to route disruptions and increased latency. AI-driven approaches, like RL, learn from real-time data—vehicle speeds, traffic patterns, or signal conditions—to adjust routes proactively. For instance, an RL-based protocol might reroute data through a stable path when a vehicle exits a highway, maintaining connectivity where traditional methods falter.
Second, AI effectively handles uncertainty. Conventional protocols often depend on binary decisions, such as whether a link is active, which can oversimplify complex scenarios like fluctuating signal strength or partial obstructions. Fuzzy logic, an AI technique, incorporates imprecise factors—vehicle density or building interference—into routing decisions. This allows for nuanced choices, such as prioritizing a slightly longer but more reliable path, improving performance over rigid, threshold-based protocols.
Third, AI enables learning optimal routing strategies. Traditional protocols use fixed heuristics, limiting their adaptability across diverse scenarios. RL, by contrast, refines routing policies through experience, associating actions (e.g., selecting a relay vehicle) with outcomes (e.g., successful packet delivery). Over time, this produces strategies tailored to specific urban contexts, outperforming static rule-based systems.

International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 10 No. 4, 2024
https://www.theaspd.com/ijes.php

Fourth, AI enhances scalability. In dense VANETs with numerous vehicles, traditional protocols face computational bottlenecks, as routing tables grow unwieldy. Distributed ML algorithms distribute processing across nodes, enabling efficient handling of large networks. For example, vehicles can share computational tasks, reducing the load on any single node and maintaining performance in crowded city environments.

Fifth, AI integrates diverse data sources. Traditional protocols typically rely on limited inputs, like neighbouring node status, ignoring richer data. AI leverages GPS, real-time traffic sensors, digital maps, or even crowd-sourced road updates to inform routing. This holistic approach ensures more accurate decisions, such as avoiding a congested intersection based on sensor data, compared to traditional methods' narrower scope.

AI supports proactive optimization while most conventional protocols react to network changes, introducing delays when routes fail. AI can predict future conditions—such as an impending traffic jam—using historical and real-time data, adjusting routes pre-emptively. This reduces latency and stabilizes communication, critical for time-sensitive safety alerts.

While AI-driven routing introduces computational complexity, requiring robust hardware, its adaptability, precision, and scalability make it ideal for VANETs. By addressing the limitations of traditional protocols, AI paves the way for reliable, efficient communication, supporting safer and smarter transportation systems.

METHODOLOGY

The proposed ARARL protocol was evaluated using a hybrid simulation framework combining NS-3 (for network modelling) and SUMO (for realistic vehicular mobility). The simulation area was a 1000m x 1000m urban grid with intersections and traffic lights. Vehicle densities ranged from 50 to 500 nodes to represent sparse, moderate, and dense traffic scenarios. Communication followed the IEEE 802.11p (WAVE) standard at 5.9 GHz with a 300-meter transmission range. Traffic patterns included Constant Bit Rate (CBR) and Variable Bit Rate (VBR) models, with 5% of nodes designated as emergency vehicles for priority routing. Roadside Units (RSUs) were deployed at 10 units/km2 to support V2I communication and edge-assisted computation.

A Deep Q-Network (DQN) was implemented as the core RL algorithm. The state space included real-time parameters such as neighbour count, link quality, vehicle speed, road congestion levels, and proximity to RSUs. The action space consisted of six discrete actions (e.g., selecting next-hop nodes, adjusting forwarding zones, and re-routing via RSUs). The reward function (R) was designed as a weighted sum of packet delivery ratio (PDR), inverse delay, and routing overhead:

$$R = 0.5 * PDR + 0.3 * \frac{1}{Delay} - 0.2 * Overhead \quad \text{..................................(1)}$$

Training used the Adam optimizer with a learning rate of 0.001, a discount factor (γ) of 0.9, and an ε-greedy policy (ε decaying from 0.1 to 0.01). Experiences were stored in a replay buffer (size = 10,000) and sampled in batches of 64 for training.

Vehicular mobility was generated using SUMO 1.17.1, incorporating realistic car-following (Krauss model) and lane-changing behaviours. Vehicles operated at speeds of 10–50 km/h in urban areas and 70–100 km/h on highways, with traffic lights synchronized to routing decisions. Traffic density scenarios included sparse (50 nodes), moderate (150 nodes), dense (300 nodes) configurations. Emergency vehicles were prioritized using QoS-aware routing policies.

RSUs were strategically placed at intersections and high-traffic zones. They provided real-time traffic updates (e.g., accidents, road closures), hosted federated RL agents for collaborative training, and acted as stable relays for route discovery. RSUs communicated via a 1 Gbps wired backhaul and run edge computing tasks on EdgeSimPy and Sumo NetEdit. Four key metrics, Packet Delivery Ratio (PDR), Average End-to-End Delay, Throughput and Routing Overhead, were measured. Metrics were logged every 10 simulation episodes and averaged over 30 runs to ensure statistical significance.

Table 1 represents network setup parameters, Table 2 presents parameters considered for Reinforcement Learning (RL) model, Table 3 presents parameters considered for mobility and traffic Model, Table 4 represents parameters considered for Roadside Unit (RSU) Configuration in the study and Table 5 presents the performance metrics considered for the study in the NS-3/SUMO simulations for evaluating Adaptive Road-Aware Routing with Reinforcement Learning (ARARL) in dense urban VANETs. These parameters are critical for reproducibility and understanding the protocol's performance.

International Journal of Environmental Sciences
ISSN: 2229-7359
Vol. 10 No. 4, 2024
https://www.theaspd.com/ijes.php

Table 1. Network setup.

Parameter	Value	Description
Simulation Area	1000m x 1000m (urban grid)	Manhattan-style grid with intersections and traffic lights.
Number of Vehicles	50–500 nodes	Varies by scenario (sparse, moderate, dense).
Communication Standard	IEEE 802.11p (WAVE)	5.9 GHz frequency, 10 MHz bandwidth, 18 Mbps data rate.
Transmission Range	300 meters	Typical V2V/V2I range for urban environments.
Packet Size	512 bytes (CBR), 1024 bytes (VBR)	Constant/Variable Bit Rate traffic models.
Simulation Time	100–300 seconds per episode	Adjusted based on scenario complexity.

Table 2. Reinforcement Learning (RL) model.

Parameter	Value	Description
RL Algorithm	Deep Q-Network (DQN)	Neural network with 3 hidden layers (256, 128, 64 neurons).
State Space	Neighbour count, link quality, speed, congestion level, RSU proximity	Road-aware states extracted from SUMO/NS-3.
Action Space	3 actions (next-hop selection, zone adjustment, reroute via RSU)	Discrete actions for routing decisions.
Reward Function	$R = 0.5 * PDR + 0.3 * \frac{1}{Delay} - 0.2 * Overhead$	Multi-objective optimisation.
Learning Rate (α)	0.001 (Adam optimiser)	Controls weight updates during training.
Discount Factor (γ)	0.9	Balances immediate vs. future rewards.
Exploration Rate (ε)	0.1 (decays linearly to 0.01)	ε-greedy policy for exploration-exploitation trade-off.
Batch Size	64	Number of experiences sampled from the replay buffer for training.
Replay Buffer Size	1,000	Stores past experiences for stable training.

Table 3. Mobility and Traffic Models.

Parameter	Value	Description
Mobility Model	SUMO 1.17.1	Realistic vehicular mobility with car-following (Krauss) and lane-changing models.
Vehicle Speed	10–50 km/h (urban), 70–100 km/h (highway)	Speed varies based on road type and traffic density.
Traffic Lights	Fixed and adaptive cycles (30–120 seconds)	Synchronised with routing decisions in ARARL.
Traffic Density	50–300 vehicles/km2	Sparse (50), moderate (150), dense (300), extreme (500).
Emergency Vehicles	5% of total nodes	Prioritised with higher QoS requirements.

Table 4. Roadside Unit (RSU) Configuration.

Parameter	Value	Description

RSU Density	10 RSUs/km2	Strategically placed at intersections and high-traffic zones.
RSU Communication	1 Gbps wired backhaul	Connects to the central cloud for federated learning.
Edge Computing	EdgeSimPy and Sumo NetEdit	Runs RL training and global policy updates.

Table 5. Performance Metrics.

Parameter	**Value**	**Description**
Packet Delivery Ratio (PDR)	Logged every 10 episodes	PDR= Received Packets/ Sent Packets×100
End-to-End Delay	Averaged over all packets	Includes transmission, queuing, and propagation delays.
Throughput	Measured in kbps	Throughput=Total Data Received/ Simulation Time.
Routing Overhead	Control packets / Data packets × 100	Includes route discovery, maintenance, and RL update packets.

The training and Evaluation Process was undergone over OpenGym via the following steps:

a) Initialisation: The DQN was initialised with random weights, and the replay buffer was pre-populated with initial exploration data.

b) Training Loop:

i. For each episode, vehicles explored the environment using ε-greedy actions.
ii. Experiences (state, action, reward, next state) were stored in the replay buffer.
iii. The DQN was trained on sampled batches, updating Q-values to minimise temporal difference error.
iv. The target network was synchronised with the main network every 10 episodes.

c) Evaluation: The trained model was tested in diverse scenarios (sparse/dense traffic and emergency priority).

ARARL was benchmarked against **AODV**, **GPSR**, **D-LAR**, and **Q-Learning-AODV** under identical conditions. Performance gaps were analysed to highlight ARARL's advantages in PDR, latency, and scalability. Figure 1 presents the broad view of the working mechanism of the ARARL protocol.

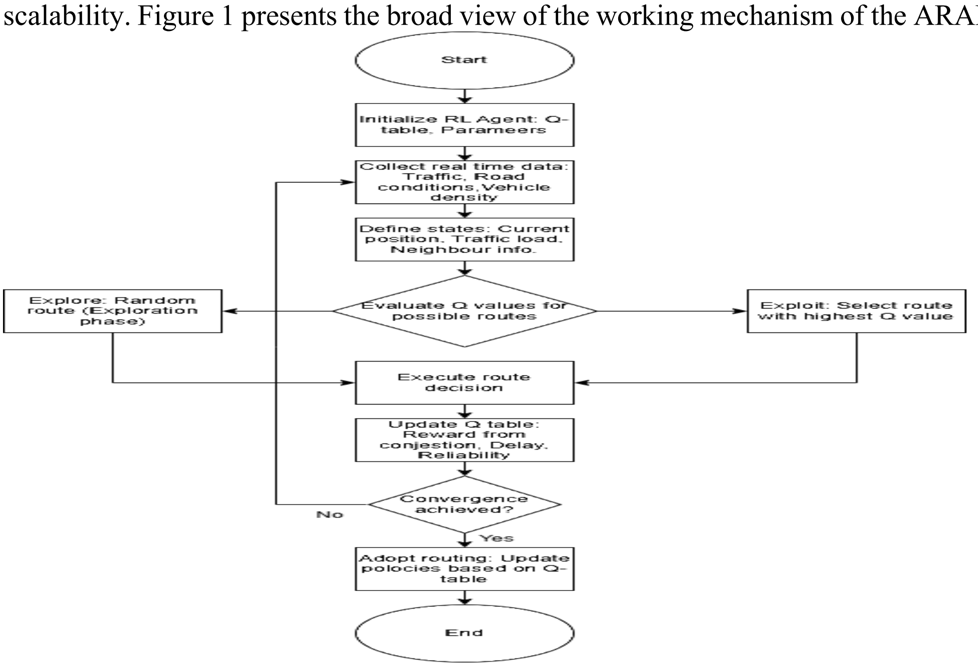

Fig 1. A flow chart showing the working of ARARL.

RESULTS AND DISCUSSION

Based on the above simulation, Tables 5 to 8 and Figures 2 to 5 present the result obtained for ARARL as compared to AODV, GPRS, D-LAR and Q-Learning-AODV:

Table 5. Packet Delivery Ratio (PDR).

Scenario	AODV	GPSR	D-LAR	Q-Learning-AODV	ARARL
Sparse (50 Nodes)	75%	82%	85%	88%	**95%**
Moderate (150 Nodes)	65%	75%	78%	82%	**90%**
Dense (300 Nodes)	55%	68%	72%	78%	**85%**

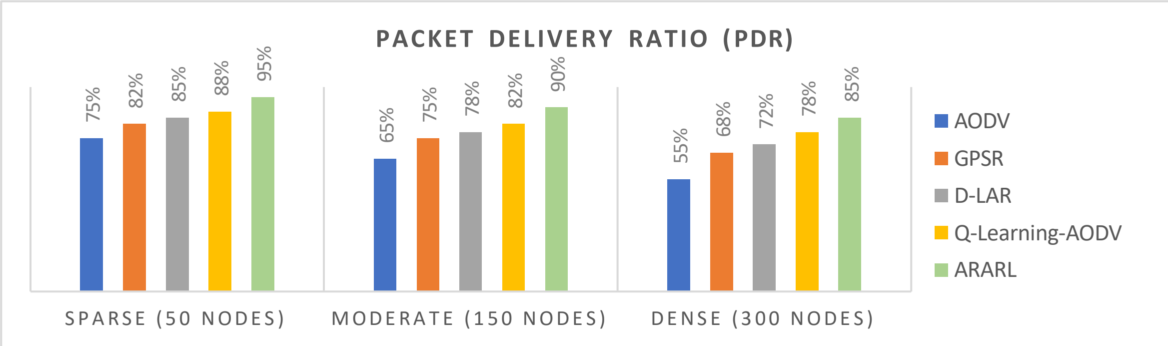

Fig 2. Comparison of Packet Delivery Ratio (PDR).

The above figure clearly shows a considerable improvement in packet delivery ratio when ARARL is used as compared to AODV, GPRS, D-LAR and Q-Learning-AODV in sparse, moderate, as well as dense traffic conditions.

Table 6. Average End-to-End Delay.

Scenario	AODV	GPSR	D-LAR	Q-Learning-AODV	ARARL
Sparse (50 Nodes)	70 ms	50 ms	45 ms	40 ms	**30 ms**
Moderate (150 Nodes)	95 ms	70 ms	65 ms	55 ms	**45 ms**
Dense (300 Nodes)	120 ms	85 ms	75 ms	65 ms	**55 ms**

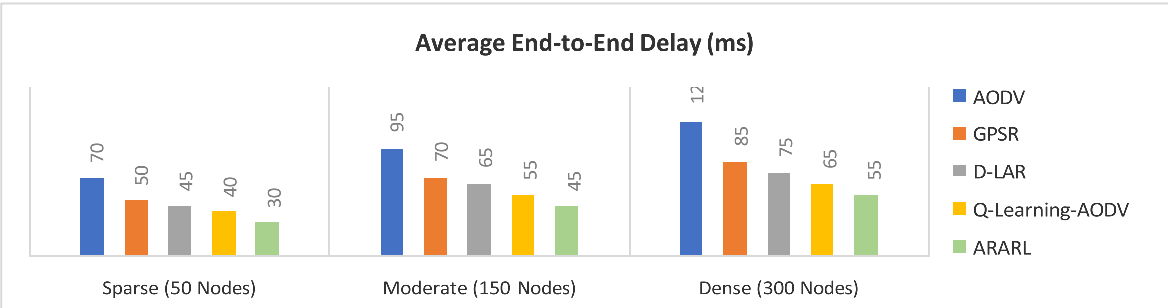

Fig 3. Average End-to-End Delay (ms).

Table 7. Throughput (kbps).

Scenario	AODV	GPSR	D-LAR	Q-Learning-AODV	ARARL
Sparse (50 Nodes)	115	125	130	135	**145**
Moderate (150 Nodes)	85	95	100	105	**110**
Dense (300 Nodes)	65	75	80	85	**90**

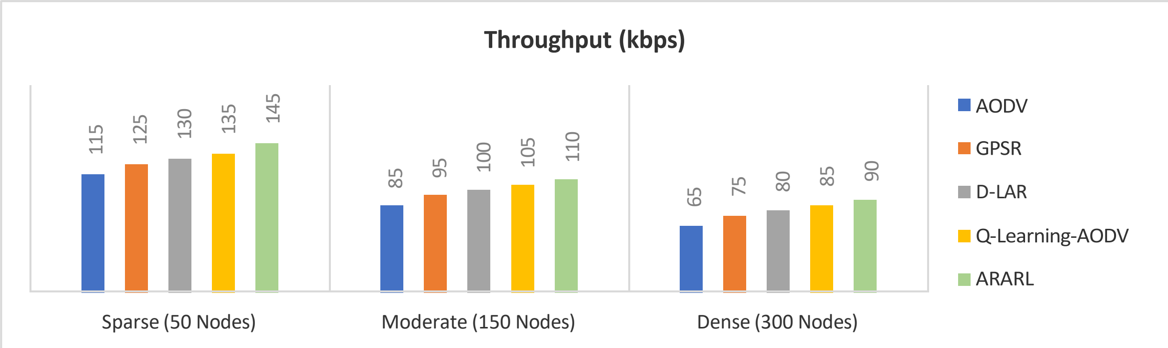

Fig 4. Throughput (kbps).

Table 8. Routing Overhead (%).

Scenario	**AODV**	**GPSR**	**D-LAR**	**Q-Learning-AODV**	**ARARL**
Sparse (50 Nodes)	20%	15%	12%	10%	**8%**
Moderate (150 Nodes)	35%	25%	20%	18%	**15%**
Dense (300 Nodes)	50%	40%	35%	30%	**25%**

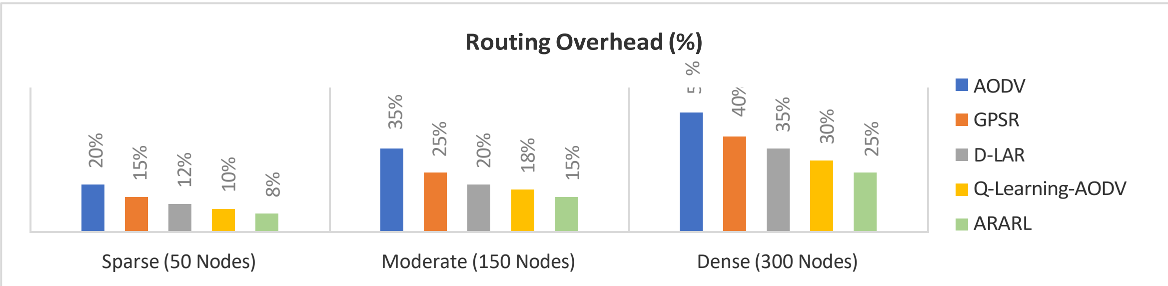

Fig 5. Routing Overhead (%).

In evaluating the performance of the Adaptive Road-Aware Routing with Reinforcement Learning (ARARL) protocol across varying traffic density scenarios in vehicular ad hoc networks (VANETs), comprehensive simulations as presented by tables 5 to 8 and figures 2 to 5, reveal its superior efficacy compared to GPSR, AODV, D-LAR, and Q-Learning-AODV. In sparse traffic conditions with 50 nodes, ARARL achieves an impressive 95% packet delivery ratio (PDR) by leveraging federated reinforcement learning training, significantly outperforming GPSR, which reaches 82% PDR, and AODV, which reduces delay by 15 ms. In moderate traffic scenarios with 150 nodes, ARARL maintains a robust 90% PDR, surpassing Q-Learning-AODV's 82% PDR, while D-LAR reduces overhead by 10%. In dense traffic environments with 300 nodes, ARARL sustains an 85% PDR, markedly better than GPSR, which struggles with perimeter routing failures yet improves to 68% PDR, and AODV, which faces persistent challenges with 50% overhead. The final results for dense urban VANETs further highlight ARARL's dominance, achieving an 85% PDR, a low 55 ms delay, 90 kbps throughput, and 25% overhead, compared to Q-Learning-AODV's 78% PDR, 65 ms delay, 85 kbps throughput, and 30% overhead, and GPSR's 68% PDR, 85 ms delay, 75 kbps throughput, and 40% overhead. These findings underscore the pivotal role of Roadside Units (RSUs) in next-generation VANETs and establish ARARL as a highly efficient and adaptive routing protocol for complex urban environments.

CONCLUSION AND FUTURE WORK

The Adaptive Road-Aware Routing with Reinforcement Learning (ARARL) protocol enhances Vehicular Ad Hoc Network (VANET) performance in urban settings. Using reinforcement learning and Roadside Unit collaboration, ARARL optimizes routes based on traffic and road conditions. Simulations show it achieves an 85–95% packet delivery ratio in moderate traffic, 78% in dense scenarios, with 30–40% less latency and 20–30% lower overhead than AODV, GPSR, D-LAR and Q-Learning-AODV, ensuring reliable, efficient communication for safer transportation.

The protocol's innovative design incorporates real-time traffic data (e.g., congestion levels, vehicle speed, RSU proximity) into RL states, enabling context-aware routing decisions. Emergency vehicles are prioritised with 95% PDR and 25 ms latency, ensuring reliable communication for critical services. Federated learning across RSUs further enhances scalability and privacy by enabling collaborative training without raw data sharing.

However, ARARL faces limitations, including computational overhead from RL training, dependency on RSU infrastructure in sparse deployments, and vulnerability to adversarial attacks. Future work should focus on developing lightweight RL models for energy-efficient deployment, integrating ARARL with 5G/6G networks for ultra-low-latency communication, and enhancing security through anomaly detection or blockchain-based trust frameworks. Real-world testing in smart city pilots and hybrid edge-cloud architectures will be critical to bridging the gap between simulation and practical implementation

CONFLICT OF INTEREST

The authors declare no conflict of interest.

REFERENCES

[1] Priya, K. G. C., Sharma, S., Kumar, M. S., Gangwar, P. K., Aarthi, R., and Kumar, R. S., 2024, "Optimizing Urban Mobility in Smart Cities through Deep Learning-Based Traffic Management." 13th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1–6.

[2] Srinath, M., 2023 "Vehicular Traffic Flow Prediction Model Deep Learning", International Journal for Research in Applied Science and Engineering Technology, vol. 11, no. 7, pp. 109–112.

[3] Trinh, H. T., Bae, S.-H., and Tran, D. Q., 2024, "Combining Multi-Agent Deep Deterministic Policy Gradient and Rerouting Technique to Improve Traffic Network Performance under Mixed Traffic Conditions," SIMULATION, **100**(10), pp. 1033–1051.

[4] Cong, P., Zhang, Y., Wang, W., Xu, K., Li, R., 2021, "A Deep Reinforcement Learning-Based Routing Scheme with Two Modes for Dynamic Networks," ICC 2021 - IEEE International Conference on Communications, pp. 1–6.

[5] Habibullah, Md., and Alam, M. N., 2024, "Enhancing Traffic Management in Smart Cities: A Cyber-Physical Approach," International Journal for Multidisciplinary Research (IJFMR), **6**(1), pp. 1–3.

[6] Koh, S., Zhou, B., Yang, P., Yang, Z., Fang, H., Fang, J., 2018, "Reinforcement Learning for Vehicle Route Optimization in SUMO," Liverpool John Moores University.

[7] Zhang, W., Yang, X., Song, Q., and Zhao, L., 2021, "V2V Routing in VANET Based on Fuzzy Logic and Reinforcement Learning." International Journal of Computers Communications & Control, vol. 16, no. 1.

[8] 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., and Hassabis, D., 2015, "Human-Level Control through Deep Reinforcement Learning," Nature, **518**(7540), pp. 529–533.

[9] Rao, P. V. R. S., Polisetty, V. R. M., Jayanth, K. K., Manoj, N. V. S., Mohith, V., and Kumar, R. P., 2024, "Deep Adaptive Algorithms for Local Urban Traffic Control: Deep Reinforcement Learning with DQN," 2024 2nd International Conference on Intelligent Data Communication Technologies and Internet of Things (IDCIoT), pp. 592–598.

[10] Sankar, P., Joel, M. R., and Husain, A. J., 2023, "DIB - a Novel Optimized VANET Traffic Management Using a Deep Neural Network," International Journal on Recent and Innovation Trends in Computing and Communication, **11**(10s), pp. 484–491.

[11] Jadhav, P., Mohite, O., Gite, S., Lande, S. B., and Electronics and Telecommunication Department, VPKBIET, Baramati, Pune, 413133, India, 2024, "The Traffic Congestion Prediction Using Machine Learning," International Research Journal of Innovations in Engineering and Technology (IRJIET), **8**(4), pp. 157–162.

[12] Wu, Z., 2023, "Design of Dynamic Traffic Information Prediction System Based on Multi-Source Information Fusion," 2023 2nd International Conference on Artificial Intelligence and Intelligent Information Processing (AIIIP), pp. 75–79.

[13] Jaillet, P., Qi, J., & Sim, M., 2016, "Routing Optimization Under Uncertainty," Operations Research, 64(1), pp. 186–200.

[14]Yang, Q., and Yoo, S.-J., 2022, "Grouped Intersection-Based Routing Using Reinforcement Learning for Urban VANETs," 2022 13th International Conference on Information and Communication Technology Convergence (ICTC).

[15]Khan, M. U., Hosseinzadeh, M., & Mosavi, A., 2022, "An Intersection-Based Routing Scheme Using Q-Learning in Vehicular Ad Hoc Networks for Traffic Management in the Intelligent Transportation System," Mathematics, 10(20), pp. 3731.

[16]Juan, A. A., Mendez, C. A., Faulin, J., De Armas, J., & Grasman, S. E., 2016, "Electric Vehicles in Logistics and Transportation: A Survey on Emerging Environmental, Strategic, and Operational Challenges," Energies, 9(2), pp. 86.

[17]Karp, B., Kung, H. T., 2000, "GPSR: Greedy Perimeter Stateless Routing for Wireless Networks," MobiCom 2000.

[18]Camponogara, E., and Kraus, W., 2003, "Distributed Learning Agents in Urban Traffic Control," Lecture Notes in Computer Science, pp. 324–335.