ISSN: 2583-7346
DOI: https://doi-ds.org/doilink/05.2024-34721289/IJAEAST.2024.04.0001, April 2024, Volume-2, Issue-4, pp.115 - 124
International Journal of Advances in Engineering Architecture Science and Technology

Intersection Management and Control using DRL and V2I

Geetha R[1], Benaka Aditya N[2], Amruth Srinivas M G[3], Dhanush C[4], Hemanth N[5]

[1]Assistant Professor, Department of Computer Science & Engineering, Bangalore Institute of Technology, Bengaluru, Karnataka, India

[2,3,4,5]Student, Department of Computer Science & Engineering, Bangalore Institute of Technology, Bengaluru, Karnataka, India

geethar@bit-bangalore.edu.in[1], benakaaditya30@gmail.com[2], amruthsrinivas02@gmail.com[3], dhanush12232002@gmail.com[4], hemanthsolanki12@gmail.com[5]

Received: 21-02-2024 **Accepted:** 20-04-2024 **Published:** 22-04-2024

Abstract

Background: Intersections are critical points on our roads, frequently becoming hotspots for congestion and accidents.
Objectives: Through the integration of DRL and V2I, the initiative seeks to improve traffic circulation, reduce congestion, and boost transportation efficiency in urban areas.
Methods: This initiative leads the way in merging Vehicle-to-Infrastructure (V2I) communication with Deep Reinforcement Learning (DRL) to transform urban transportation, focusing on intersection management.
Statistical Analysis: Traditional methods, such as static signs and traffic lights, often fall short because they focus more on the flow of traffic as a whole, rather than on the specific behaviours of individual vehicles. To tackle this issue, we're introducing a new strategy that employs Deep Reinforcement Learning (DRL) to better manage how vehicles take turns at intersections.
Findings: An optimized DRL algorithm that enhances safety, minimizes congestion, reduce waiting times at a unsignalized intersection.
Applications and Improvements: The proposed intersection management system can be adapted to various intersection layouts (e.g., T-junctions, roundabouts) and diversified traffic participants (e.g., buses, bicycles, pedestrians). Additionally, integration with established traffic management infrastructure like traffic lights or ramp meters can enhance overall traffic efficiency and flow optimization at a city or regional level.

Keywords: Intersection, Deep Reinforcement learning, Vehicle to Infrastructure communication, right of way, Markov's decision process.

1. Introduction

In today's fast-moving world, the integration of advanced technologies has sparked significant changes across various facets of our everyday routines. Among these innovations, one stands out: Vehicle-to Infrastructure (V2I) communication. Beyond its technological competence, V2I communication plays a crucial role modern transportation network, fundamentally altering our approach to road navigation. This technology enables instantaneous communication between vehicles and an infrastructure, facilitating the exchange of vital data like position, velocity, and trajectory. The impact of V2I communication extends to enhancing safety, optimizing traffic flow, promoting environmental sustainability, and advancing the development of autonomous vehicles. Deep Reinforcement Learning (DRL) is an emerging field of research and development that harnesses artificial intelligence and deep learning techniques to enhance the communication performance and efficiency between vehicles on the road.

This project focuses on revolutionizing intersection management by combining V2I communication with Deep Reinforcement Learning. V2I communication allows vehicles to share real-time data at intersections, improving traffic coordination and safety. The project emphasizes the innovative integration of DRL, an artificial intelligence technique that enables vehicles to learn optimal decision-making strategies through trial and error, without relying on labelled expert data. By utilizing DRL, the project aims to improve the efficiency and flexibility of intersection management systems, creating a smarter and more responsive traffic control mechanism. The ultimate aim is to explore how the collaboration between V2I communication and DRL can redefine intersection management, contributing to a safer and more efficient urban transportation system.

Intersection management (IM) presents a formidable challenge due to its association with a high frequency of fatal and injurious collisions, particularly in close proximity to intersections. Moreover, intersections serve as significant traffic flow bottlenecks, leading to congestion and heightened pollutant emissions. Addressing collision prevention necessitates a meticulous and effective management approach for allocating the right-of-way to vehicles. However, conventional solutions, such as static signs or traffic lights, are constrained as they fail to account for traffic dynamics at a microscopic level or the diverse routes taken by vehicles.

This study presents an innovative solution for developing an efficient scheduling policy and compares it with existing cutting-edge alternatives. Our approach focuses on utilizing Deep Reinforcement Learning (DRL) algorithms to achieve this goal. These algorithms leverage deep neural networks to learn nearly optimal control policies for the system through a process of trial and error. Unlike supervised learning, DRL doesn't rely on expert-labelled data; instead, it allows the system to autonomously devise strategies based on simple rewards. This theoretically simplifies the setup process and enhances adaptability across various domains.

The recent success of DRL-based approaches in areas previously untouched by traditional AI and optimization methods can be attributed to these distinctive characteristics. To craft a DRL-based solution, the following steps need to be undertaken:

- Define an environment model (including state representation and action space).
- Define a reward structure.
- Choose a suitable DRL policy.
- Train and assess the policy's performance.

2. Literature Survey

Suganthi and colleagues [1] present a comprehensive approach to enhancing automotive systems through the integration of software-hardware codesign and Vehicle-to-Everything (V2X) communication protocols. By simulating vehicle dynamics using Simulink design in MATLAB and employing real-time data analysis tools such as Grafana, the research effectively evaluates critical performance metrics during electric vehicle drive cycles. Additionally, the implementation of V2X communication enables seamless data exchange between vehicles and infrastructure, facilitating advanced functionalities such as lane detection, range estimation, and collision avoidance. Through computer vision-based algorithms and AI-driven lane detection models, the study demonstrates significant advancements in autonomous driving capabilities. The

indings underscore the importance of integrating advanced technologies to promote safety, efficiency, and connectivity in automotive applications, with potential future applications including over-the-air firmware updates, IoT-based implementations, and cloud analytics.
Óscar Pérez-Gil and his team [2] evaluated and proposed autonomous navigation methodologies utilizing Deep Reinforcement Learning (DRL) algorithms within the CARLA Simulator. Through training and validation stages, the performance of DQN and DDPG algorithms is analyzed. DDPG exhibits faster convergence during training and achieves trajectories comparable to a classic Linear Quadratic Regulator (LQR) controller in the validation phase. Qualitative comparisons highlight the reliability of DDPG-based navigation. These findings underscore the potential of DRL-based approaches in advancing autonomous vehicle navigation, with DDPG showing promising performance and human-like driving characteristics, setting a strong foundation for future research in this area.
Alexandre Lombard and colleagues [3] introduced an innovative method for managing intersections utilizing Deep Reinforcement Learning (DRL) and inter-vehicular communications (CIM). The methodology entails defining a Markov Decision Process and adapting DRL techniques, particularly Deep Q-Networks (DQN), to regulate vehicle right-of way at intersections. Through extensive simulations and performance assessments, the proposed DQN approach is compared with traditional traffic light control and first-come first served (FCFS) cooperative intersection methods, showcasing its superior performance in reducing waiting time, enhancing throughput, and minimizing CO2 emissions, especially in high-traffic scenarios. The study underscores the adaptability of the DRL approach to fluctuating traffic conditions without requiring manual recalibration and its versatility across different intersection layouts with minimal adjustments. Future directions include exploring alternative DRL models for optimization, integrating predictions of vehicle behavior to enhance training, and extending the approach to networks of intersections while mitigating performance impacts.
In a similar vein, Dongho Choi et al. [4] present a grid prediction model that amalgamates Random Forest (RF) and Long Short-Term Memory (LSTM) encoder-decoder architecture to forecast the lane change intention and trajectory of surrounding vehicles (SVs). Data collection involved recording a dataset using a vehicle equipped with V2V communication, camera sensor, and LIDAR, resulting in 932 trajectories obtained in a testbed environment resembling a highway, with a subsequent 70% training and 30% testing data split. The model exhibits notable positional accuracy beyond 1 second, albeit displaying comparatively lower precision before this timeframe due to premature lane change predictions.
Wangpengfei Yu et al. [5] present an innovative approach to train intelligent vehicles for navigating complex intersections using deep reinforcement learning algorithms, including the Distributional DQN, DDQN, and DQN algorithms. Utilizing a PyTorch implementation with CUDA acceleration on NVIDIA GPUs ensures efficient computation. The experimental setup is meticulously designed with detailed hyperparameters, such as learning rate and experience replay memory size, to maintain consistency across trials. Evaluation metrics, such as intersection passing rate and standardized reward, are employed to gauge algorithmic performance. Additionally, a risk assessment technique integrated into the reward function during training aims to enhance driving safety. Furthermore, a proposed state-information based attention network enhances the vehicle's perception capabilities. The experimental findings underscore the superior convergence speed and intersection passing rate achieved by the distributional DQN algorithm.
Dinesh Cyril Selvaraj et al. [6] use a Deep Reinforcement Learning (DRL) method aimed at refining adaptive cruise control (ACC) systems within connected autonomous vehicles. Methodologically, the approach involves integrating and appropriately weighting essential parameters such as headway, longitudinal slip, and jerk to optimize traffic efficiency, safety, and

comfort. Comparative analysis demonstrated significant performance enhancements compared to traditional ACC and cooperative ACC schemes, notably achieving a 36% increase in headway performance and a 47% improvement during lead vehicle speed variation phases. These advancements contribute to elevated traffic flow efficiency across diverse conditions, validated through the utilization of realistic vehicle dynamics modeling and rigorous scenario testing. Importantly, the study underscores the critical role of Vehicle-to-Everything (V2X) communication, particularly regarding lead vehicle acceleration, in augmenting the DRL-based ACC application's performance.

Ilgin Gokasar and colleagues [7] present SWSCAV, an innovative real-time traffic management system utilizing Connected and Autonomous Vehicles (CAVs) to optimize traffic flow during incidents. Evaluated in a simulated urban environment using SUMO, SWSCAV emerges as superior to conventional methods like Lane Control Systems (LCS) and Variable Speed Limits (VSL). SWSCAV proves effective with CAV penetration rates ranging from 10% to 50%, particularly beneficial for incidents in the left lane. However, incidents in the middle and right lanes require higher penetration rates of 60% to 70% for optimal performance. Notably, SWSCAV consistently outperforms LCS and closely matches the average speed improvements seen with VSL, even at lower CAV penetration rates. The potential integration of SWSCAV with VSL suggests opportunities for further enhancement. Stressing the importance of seamless CAV integration, SWSCAV maximizes existing infrastructure without significant additional costs.

Cesar Leonardo González et al. [8] proposed a decentralized coordination system for managing intersections offer a promising solution to mitigate traffic conflicts, particularly in urban environments. By prioritizing emergency vehicles and operating without a central manager susceptible to bottlenecks or failures, the system showcases efficiency in intersection management. Through empirical testing, it has demonstrated superiority over centralized approaches, exhibiting enhanced traffic flow, increased average vehicle speeds, and decreased waiting times at intersections. Recognizing the limitation of assuming one-way traffic lines, it aims to broaden the system's scope to encompass multiple lines for each direction, especially advantageous in larger cities with heightened traffic densities.

Lastly, Anas Berbar et al. [9] developed an innovative strategy utilizing double Q-learning agents within a decentralized reinforcement learning framework to manage platoons of Connected Autonomous Vehicles (CAVs) at signalized intersections within smart city settings. The training process involves the velocity agent, which undergoes training and execution to subsequently train the Signal Agent (SA). This decentralized training approach facilitates efficient coordination of CAV platoons without relying on centralized control. The system's effectiveness is evidenced by two significant improvements: firstly, a notable reduction in the average delay experienced by CAVs passing through urban intersections, resulting in an average improvement of 47.3%. Secondly, an enhancement in fuel efficiency by an average of 13.6%, a critical factor for long-term sustainability. The insights derived from the results suggest that introducing platoons in high-traffic scenarios effectively reduces fuel consumption and average delays.

3. Methodology

a) Overview of an Intersection

At an intersection ("I"), we have several roads meeting up, each road having its own lanes. These lanes are like pathways: some lead into the heart of the intersection (we'll call them "incoming lanes" or entry zones), while others lead away from it (let's call those "outgoing lanes" or exit zones). So, we've got a setup where we can say I = {In, Out}, with In representing the incoming lanes and Out representing the outgoing ones. Now, the area from the end of an incoming lane to the start of an outgoing lane is what we call the conflict zone (CZ). Here's the interesting part: while vehicles cruising along incoming or outgoing lanes stick to their own paths, those in the conflict zone have to navigate from the end of one incoming lane to the start of an outgoing one. A pair (entry, exit) ∈ In × Out is recognized as a route. If there exists an intersection between two routes, denoted as r1 and r2, signifying that a vehicle aiming to reach the exit of r1 from the associated incoming lane may potentially collide with a vehicle following r2, then the two routes are deemed conflicting. It's crucial to emphasize that a route is not conflicting with itself, as two vehicles following the same route can be simplified to a car-following scenario.

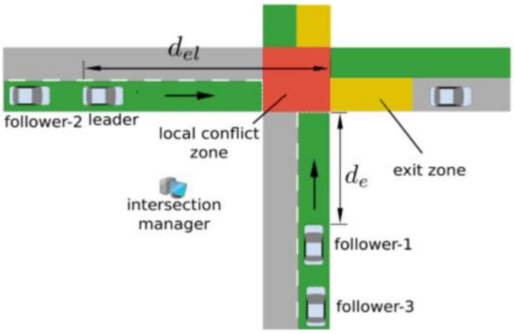


Figure 1. Intersection Scenario

b) Deep reinforcement learning

A typical Reinforcement Learning (RL) system is a Markov Decision Process (MDP) {S, A, T, R} (State, Action, Transition, Reward). To apply RL, a first step is to identify the properties of the system that can be used to design the corresponding MDP. S is the state space, A the action space, T the transition function and R the reward function. The MDP can be divided into two parts: • the agent, which performs an action At ∈ A according to an observed state St ∈ S • the environment, which updates itself according to the action of the agent, and returns a new state St+1 and a corresponding reward Rt+1 ∈ R to the agent In the present situation, the decision process of the agent is driven by a deterministic policy π : S → A which tells the action to apply according to a given state. The purpose of the RL is to find the optimal policy π ∗ which maximizes the reward in the long time. Thus, the first step to apply a RL algorithm to a given system is to define the state space S, the action space A, the transition function T, and the reward function R.
The DQN approach represents a Deep Reinforcement Learning (RL) technique adapted from the Q-Learning algorithm. At its essence, DQN employs a neural network to approximate a Qfunction, which estimates the values of actions for a specific state. The utilization of a neural network in approximating the Q-function facilitates the handling of a continuous state space.

c) Proposed DRL design

Agent

- Decision maker / controller that interacts with the environment.
- The agent is equipped with sensors, such as cameras and lidar, to perceive the surroundings and make decisions based on the observed state.

Environment

- It represents the external system in which agent operates.
- It includes the positions, speeds, and intentions of other vehicles within the intersection, as well as the structure of the intersection itself.

ISSN: 2583-7346
DOI: https://doi-ds.org/doilink/05.2024-34721289/IJAEAST.2024.04.0001, April 2024, Volume-2, Issue-4, pp.115 - 124
International Journal of Advances in Engineering Architecture Science and Technology

State Representation

- The state represents the current situation or configuration of the environment at a given point in time.
- This includes positions, speeds and contextual information necessary for making decisions at the intersection.

Action space

- The action space defines the set of possible actions that agent can take in a given state.
- Actions may include accelerating, decelerating and halting at the intersection

Reward function

- The reward function quantifies the immediate benefit or cost associated with the agent's actions.
- It encourages behaviours that lead to safe exit through the intersection, such as yielding to other vehicles, avoiding collisions, and minimizing travel time.

Policy

- It is a strategy or mapping from states to actions that the agent uses to make decisions.
- It dictates how the vehicle selects actions in different states to maximize the reward.

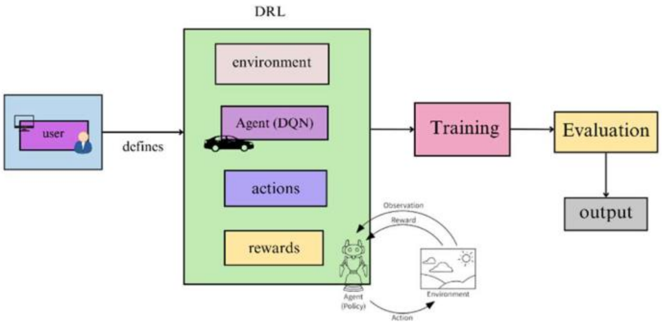


Figure 1. DRL Architecture

d) Algorithm Design

Define the environment

- Identify the intersection layout
- Define the state space
- Define the action space

Set up parameters

- Latency
- Intersection density
- Traffic pattern
- Road conditions

Create a deep learning agent

- Use a suitable neural network architecture.
- Initialize neural network parameters.
- Generate a set of random vehicles and control actions.

Design a reward function

- Design a reward function that encourages efficient traffic flow and minimizes congestion

Training

- Convert the collected state data into input vectors for the neural network.
- Calculate the target values (rewards) for the collected actions.

ISSN: 2583-7346
DOI: https://doi-ds.org/doilink/05.2024-34721289/IJAEAST.2024.04.0001, April 2024, Volume-2, Issue-4, pp.115 - 124
International Journal of Advances in Engineering Architecture Science and Technology

- ➢ Update the neural network parameters.

Validation and evaluate performance

- ➢ Repeat step 4 for a specified number of iterations.
- ➢ Calculate the average reward obtained by the agent for each iteration

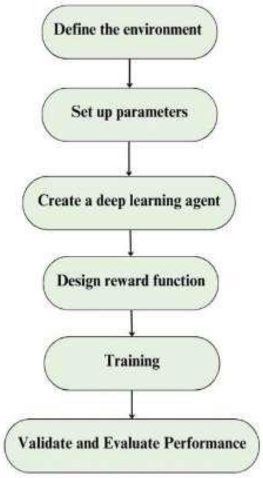


Figure 3. Algorithm Design

4. Experiments

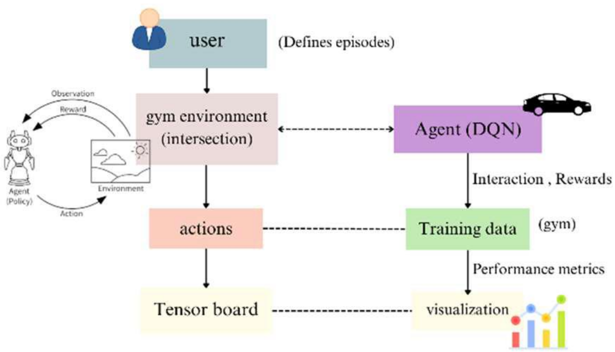


Figure 4. Proposed Block Diagram

This setup involves training a DQN agent within an intersection environment, with rewards provided based on its interactions with the environment

Agent

- ➢ Decision maker / controller that interacts with the environment.
- ➢ Type: The agent is a DQN (Deep Q-Network) agent, which is a type of reinforcement learning algorithm used for learning optimal action-selection policies in sequential decision-making problems.

Environment

- ➢ It represents the external system in which agent operates, more precisely the environment in which the agent operates is a intersection environment, present in the highway-env library.
- ➢ It includes the positions, speeds, and intentions of other vehicles within the intersection, as well as the structure of the intersection itself.

ISSN: 2583-7346
DOI: https://doi-ds.org/doilink/05.2024-34721289/IJAEAST.2024.04.0001, April 2024, Volume-2, Issue-4, pp.115 - 124
International Journal of Advances in Engineering Architecture Science and Technology

Rewards

- Rewards are typically provided to the agent based on its actions and interactions within the environment.
- These rewards could include positive reinforcement for reaching goals, avoiding collisions, maintaining safe driving behavior, etc.

Policy

- The agent follows an epsilon-greedy policy during training.
- This policy balances exploration (trying new actions to discover potentially better strategies) and exploitation (selecting actions that are currently believed to be the best) by occasionally selecting random actions (exploration) instead of always selecting the action with the highest expected reward (exploitation).

Based on the above-mentioned parameters, the code is split into training and testing. The Gymnasium module is imported to make use of rl environments. The environment and agent configuration are achieved by specifying the required parameters in a JSON file. Some of the other modules used are tensorboard and movie.py. Tensorboard is used to visualize the training process in the form of graphs, plots or histograms. MoviePy is a Python module for video editing, which can be used for basic operations (like cuts, concatenations, title insertions), video compositing (a.k.a. non-linear editing), video processing, or to create advanced effects. It can read and write the most common video formats, including GIF.
The JSON file is imported in the training loop for agent and environment configuration. In the testing phase an instance is called at random to validate the trained model. The module movie.py is used to visually demonstrate the testing scenario. Movie.py renders a video output for the test case.

5. Results

Test Case	Scenario	Ego Vehicle	Other Vehicle	Expected Conflict
1	Straight Crossing	(0, 0), Northbound (0, 10)	(50, 0), Southbound (0, -10)	No
2	Left Turn Conflict	(0, 0), Westbound (-5, 0)	(0, 50), Eastbound (-10, 0)	Yes
3	Turning Same Direction (Fast Other)	(-50, 0), Southbound (0, -15)	(50, 0), Southbound (0,-10)	Yes
4	One Vehicle Stopped	(0, 0), Northbound (0, 0)	(50, 0), Westbound (0, 0)	Yes

Figure 5. Expected Test Case Samples

Ego vehicle is the vehicle in an intersection for which the reinforcement learning algorithm is applied and it calculates the right of way taking into consideration the position and speed parameters if other vehicles on the road.

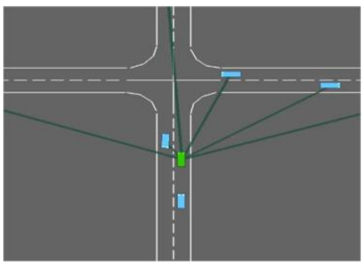


Figure 6. Representation of Ego Vehicle and The Intersection Environment Considered

ISSN: 2583-7346
DOI: https://doi-ds.org/doilink/05.2024-34721289/IJAEAST.2024.04.0001, April 2024, Volume-2, Issue-4, pp.115 - 124
International Journal of Advances in Engineering Architecture Science and Technology

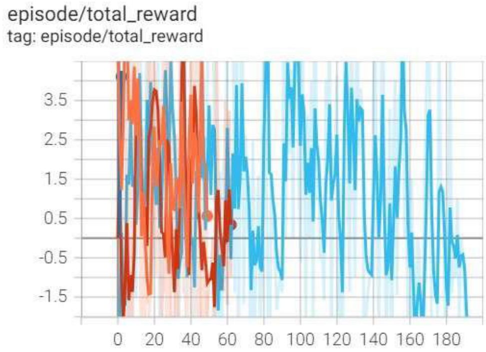
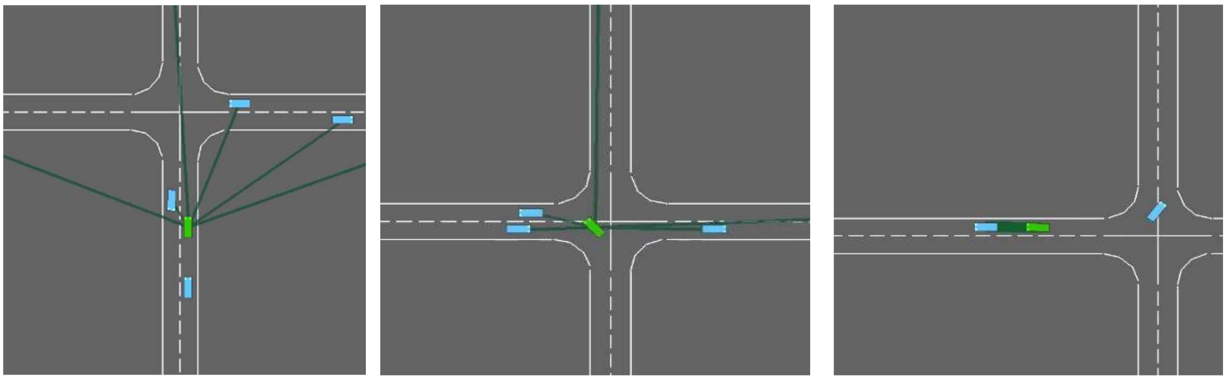


Figure 7. Training Visualization



Vehicle In Storage Zone | Vehicle Entering Conflict Zone | Vehicle Crossing Intersection

Figure 8. Vehicle Zones

6. Conclusion

Today, advanced technologies are changing our lives, and one significant innovation is Vehicle-to-Vehicle (V2I) communication. This allows cars to share real-time information, improving road safety and paving the way for self-driving cars. V2I is vital for smart cities, optimizing traffic signals and making urban transportation more efficient. Deep Reinforcement Learning (DRL) in V2I communication is a new area of research, using artificial intelligence to enhance how cars communicate on the road. DRL relies on deep neural networks, adapting strategies through trial-and-error without expert-labelled data. This groundbreaking technology is reshaping transportation, ensuring safer journeys, efficient traffic systems, and a more interconnected automotive future. V2I communication plays a crucial role in connected and autonomous vehicles (CAVs), relying on data from other vehicles for safe navigation. The integration of V2I and DRL showcases the potential for AI.

References

1. Suganthi, K.; Kumar, M.A.; Harish, N.; HariKrishnan, S.; Rajesh, G.; Reka, S.S "Advanced Driver Assistance System Based on IoT V2V and V2I for Vision Enabled Lane Changing with Futuristic Drivability", MDPI Sensors (2023), 23, 3423, 2023.
2. Óscar Pérez-Gill; Rafael Barea1; Elena López-Guillén1; Luis M. Bergasa1; Carlos Gómez-Huélamo1; Rodrigo Gutiérrez1; Alejandro Díaz-Díaz1, "Deep reinforcement

learning based control for Autonomous Vehicles in CARLA", Springer Multimedia Tools and Applications (2022), 81:3553–3576, 2022.

3. Alexandre Lombard, Ahmed Noubli, Abdeljalil Abbas-Turki, Nicolas Gaud, and Stéphane Galland (2023), "Deep Reinforcement Learning Approach for V2X Managed Intersections of Connected Vehicles", IEEE TRANSACTIONS ON INTELLIGENT TRANSPORTATION SYSTEMS, VOL. 24, NO. 7, JULY 2023.
4. Choi, D.; Yim, J.; Baek, M.; Lee, S, "Machine Learning-Based Vehicle Trajectory Prediction Using V2VCommunications and On-Board Sensors", MDPI Electronics (2021), 10, 420, 2021.
5. Yu, W.; Qian, Y.; Xu, J.; Sun, H.; Wang, J. "Driving Decisions for Autonomous Vehicles in Intersection Environments: Deep Reinforcement Learning Approaches with Risk Assessment", World Electr. Veh. J. (2023), 14, 79, 2023.
6. Selvaraj, D.C.; Hegde, S.; Amati, N.; Deflorio, F.; Chiasserini, C.F "A Deep Reinforcement Learning Approach for Efficient, Safe and Comfortable Driving", MDPI Appl. Sci. (2023) 13, 5272.
7. Ilgin Gokasar , Alperen Timurogullari , Muhammet Deveci , Harish Garg " SWSCAV: Real-time traffic management using connected autonomous vehicles", ELSEVIER ISA Transactions 132 (2023) 24–38.
8. González, C.L.; Delgado, S.L.; Alberola, J.M.; Niño, L.F.; Julián, V. (2022) "Toward Autonomous and Distributed Intersection Management with Emergency Vehicles", MDPI Electronics 2022, 11, 1089.
9. Anas Berbar, Adel Gastli, Nader Meskin, Mohammed A. AL, Jawhar Ghommam, Mostefa Mesbah and Faical Mnif. (2022) "Reinforcement Learning-Based Control of Signalized Intersections Having Platoons", IEEE VOLUME 10,2022.
10. Liwen Wang, Shuo Yang, Kang Yuan, Yanjun Huang, Hong Chen, "A Combined Reinforcement Learning and Model Predictive Control for Car-Following Maneuver of Autonomous Vehicles", Springer 2023 36:80.
11. Yao, Z.; Jiang, H.; Cheng, Y.; Jiang, Y.; Ran, B. Integrated schedule and trajectory optimization for connected automated vehicles in a conflict zone. IEEE Trans. Intell. Transp. Syst. (2020), 23, 1841–1851.
12. Mirheli, A.; Tajalli, M.; Hajibabai, L.; Hajbabaie, A. A consensus-based distributed trajectory control in a signal-free intersection. Transp. Res. Part C Emerg. Technol. (2019), 100, 161–176.