Virtual Traffic Dynamics (VTD) - A Custom AI Traffic Simulation Approach

Dr. Dinesh G[0000−0003−0586−170X], Guru Saran K[0009−0002−8101−4131], Rohith S[0009−0003−6722−8974], and Dr. Nagendra Prabhu S[0000−0003−3439−6106]

Department of Computational Intelligence, School of Computing,
SRM Institute of Science and Technology, Kattankulathur, Chengalpattu,
Tamil Nadu 603203, India

Abstract. Good urban traffic management is quite important for congestion relief and vehicle flow. This study will examine AI approaches for city traffic systems based on reinforcement learning (RL). The aim is to contrast static traffic systems that employ more conventional control strategies and dynamic traffic systems employing RL to govern traffic signals and vehicle movement. The results of this study from an intricate urban road traffic simulation of a central business district concluded that the reinforcement learning system performed superior to the static system in results and stability. The RL system also enables online learning and can minimize spilling effects resulting from network perturbations via optimal vehicle re-routing. In summary, the research suggests that AI-driven traffic management measures can provide effective uncertainty of travel times as well as enhance the resilience of urban transport networks and eventually provide scalable solutions to urban challenges.

Keywords: Urban Traffic Management, Artificial Intelligence, Reinforcement Learning, Traffic Control Systems, Dynamic Traffic Signal Control

1 Introduction

Due to rapid urbanization, combined with a shortage of infrastructure, urban transportation networks tend to face high levels of traffic congestion, leading to longer travel times and fuel usage and high degrees of environmental contamination, all of which are adverse to urban mobility and productivity. Since static management systems, traditional ones, are designed with fixed signal timings, they are generally ineffectual in managing dynamically and unpredictably variable real-world traffic flows. This leads to inefficiencies in unanticipated conditions, high volumes of vehicles, and road closures, to mention a few.

Recent developments in artificial intelligence, specifically reinforcement learning (RL), show tremendous potential to solve these problems. RL embeds agents that learn optimally controlled actions that, through interacting with their world, are specially adapted to tasks that need to adapt in real time. Within traffic control, RL embeds adjustments to signal timing and further adjusts the routing of cars, with decisions made in real time; thus, an adaptive response to shifting traffic condition context occurs, yielding enhancements to performance characteristics while minimizing delays.

In this paper, the performance of a static conventional management system is evaluated and compared with an RL-based dynamic system. An urban traffic network was represented as a graph structure with 82 nodes for the locations of every intersection (junction) and edges for the road links. In general, it can be observed that the dynamic system based on RL is able to keep the traffic flow with the adaptive behavior of responses to the change in the network when taking into account and evaluating the static system. The application of RL in traffic management signifies a significant shift in revolutionizing city transport that can address changing traffic jams in lanes, along with influencing quality of life and handling unexpected events. Whereas cities struggle with mounting demand for traffic, solutions that adapt, like RL for city car congestion, can play a role in evolution toward transportation systems that can reasonably address ever-multiplying patterns of traffic flow under the pressure and discover an answer that previously had passed comprehension.

2 Related Work

This section presents an overview of RL-based traffic management frameworks, traffic simulation tools, and game engines for AI and simulation. This section also covers research gaps and presents our contribution on RL-based traffic optimization in Unreal Engine.

2.1 Reinforcement Learning in Traffic Management

Reinforcement learning has come into prominence as a good adaptive traffic control technique. Some initial papers utilized Q-learning to control small intersections (Wiering, 2020), but because of the limitations in infrastructure, this could not be extended to a larger scale. Abdulhai et al. (2022) integrated reinforcement learning with the adaptive signal control system, performing much better compared to fixed-timing approach systems. Latest, a number of studies combine deep reinforcement learning methods like Deep Q Networks (DQNs) (Wei et al. (2023)) to generalize possible state space from observation. Nonetheless, most studies are still concerned mainly with isolated intersection traffic problems and cannot optimize traffic visibility of a whole city, much less handle real-time controls.

2.2 Game Engines for AI and Simulation

Game engines like Unreal Engine and Unity are beneficial in AI research due to their real-time physics simulation and rendering. The machine learning (ML)-Agents toolkit in Unity (Juliani et al., 2018) facilitates the research of reinforcement learning (RL), and Unreal Engine has been used effectively in the research of autonomous vehicle simulations (e.g., CARLA). But the application of RL to traffic management has not yet been made, and thus the research on RL traffic management is a thrilling prospect due to the abilities of game engines in terms of visualization and interactivity, which will be a boon for real-time RL traffic management systems.

2.3 Gaps Addressed by This Work

Our research complements some of the gaps of the earlier research, such as:

- Scalability: Whole networks can be optimized with optimization instead of single-junction techniques.
- Real-Time Adaptability: Unreal Engine is capable of supporting real-time simulation and control on a much larger scale than has ever been done previously.
- Visualization:Improved visualizations make RL model assessment and understanding of traffic flow easier.

With RL incorporated into Unreal Engine, the suggested turning maneuvers enhancement platform scales in real time and is aesthetically pleasing to complete the research gaps from other studies.

3 Background

This subsection discusses key ideas and tools of RL-based traffic optimization, which consist of RL for traffic control, traffic simulation software, and the Unreal Engine for AI-based simulations.

3.1 Reinforcement Learning for Traffic Control

RL for Traffic Control: Reinforcement learning (RL) enables agents to make their decisions optimal through interactions with the environment to maximize rewards. RL optimizes traffic flow in traffic management by controlling traffic lights. For instance, Wearing used tabular Q-learning on a single traffic light early in the history of RL but faced scalability because of state space explosion. Adoption of deep reinforcement learning (DRL), through deep-Q networks (DQN), solved the above problems using neural networks to estimate Q-values. DQN has been used, in Wei et al. (2019), to decrease vehicle wait time at multiple intersections. City-level real-time traffic optimization and road and congestion closures real-time updates/optimizations are still problems.

3.2 Unreal Engine for AI and Simulation

Originally a game development tool, Unreal Engine has been noticed in AI research because of its real-time rendering, physics engine, and capacity to include agent behaviors through the Blueprint scripting system. This has resulted in its application in autonomous simulations (e.g. CARLA, Dosovitskiy et al., 2017), and the inclusion of plugins like the Machine Learning Plugin provides support for reinforcement learning, as well as integration with TensorFlow and PyTorch. Unreal Engine supports programmed agent behavior to be included and then interactively simulates traffic. This enables agents (e.g., vehicles, pedestrians, cyclists) to engage with one another with instant feedback, which is especially pertinent for training in RL. Being able to interactively simulate traffic using a physics engine and agent models contributes to the realism of what it would be like in an actual experience. Also, the improved visualization in Unreal Engine enhances the capacity to examine traffic patterns and the core of RL for real-time simulations. While the application of Unreal Engine in the field of traffic management is only just starting, there are a number of significant benefits and advantages in applying Unreal Engine as a new frontier for innovation in AI-powered traffic optimization.

4 Methodology

In this section, a simulated urban network traffic flow is described in relation to the construction and assessment of the enhanced learning model for reinforcement. The process is the simulation setup and agent to foster RL.

4.1 Simulation Environment Setup in Unreal Engine

Unreal Engine provides rendering and physics-based AI systems comparable to real urban traffic simulations. This means that it is the best plugin for traffic optimization tests.

Urban Network Design Traffic has been simulated as a graph with 82 intersections as nodes and road segments as edges. It has been set up using the Blueprint system of Unreal Engine 5 and is designed to appear as a generic urban grid layout based on real-world lengths, roads, capacity, etc., and behaviors like lane changes and vehicle queues.

Traffic Generation Vehicles come into the network via specific vehicular entry points, and the arrival of vehicles is Poisson distributed in order to reflect different traffic conditions. Routing of vehicles is dynamically assigned using Dijkstra's Algorithm since traffic conditions are road length dependent as of now and real-time density could also be altered by user actions.

Vehicle and Traffic Signal Interactions These are interactions between traffic signals and vehicles. They follow the physical laws of driving with variable rates of acceleration and deceleration, apart from signal-based flow. Lights operate in phased sequences and manage a directional flow of traffic. The timings of the signals can be adjusted, thereby allowing a flexible framework for real-time optimization of traffic to occur through RL. It is through this designed simulation that the RL agent is trained and tested for urban traffic flow improvements.

4.2 RL Agent Design

An in-depth Q-network The RL agent controls the traffic in the network, which is very well suited for traffic control in urban areas due to its capacity to digest high-dimensional state spaces.

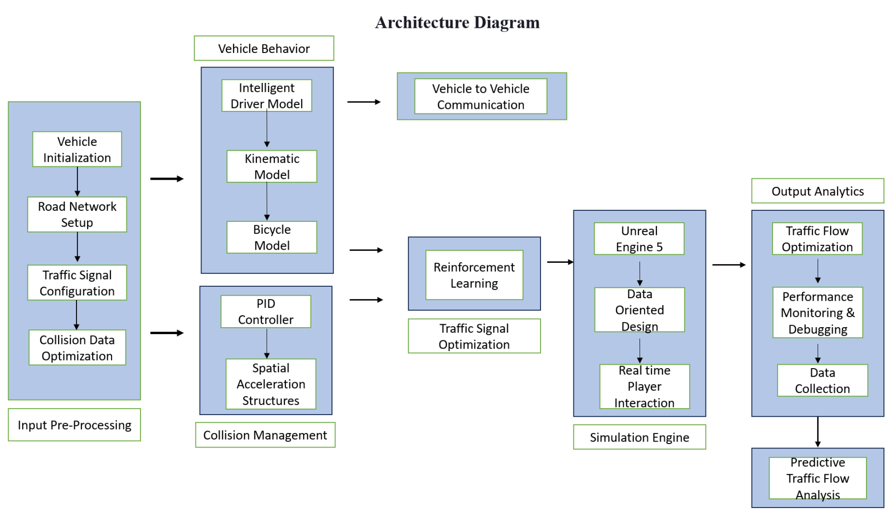


Fig. 1: Architecture Diagram.

State Space In each time step, the RL agent is guided by a feature vector:

– Queue Lengths: Waiting vehicle counts at each intersection (northbound, southbound, eastbound, westbound) in order to gauge congestion levels.

– Traffic Signal Phases: Present active signal phase, indicating permitted movements.

– Road Segment Densities: Ratio of vehicles to capacity per road segment in use—a measure of how highly congested that segment is.

These normalized features constitute the underlying network state of a DQN model.

State Space The two primary controls at the agent's disposal are:

- Signal Phase Selection: The subsequent signal phases to be implemented at intersections according to pre-stated rules will be determined.
- Routing Cost Adjustment: It is incurred by adjusting it based on routing changes that are apparent in road segments for vehicles to move more efficiently, which assists in minimizing traffic congestion.

Actions are made effective at the start of each time bucket, which allows for real-time adjustment of networks.

Reward Function Maximizing the throughput and minimizing the average travel time is how this agent maximizes his joint flow. The reward function is of this form:

$$R = \alpha \cdot \text{throughput} - \beta \cdot \text{average travel time}$$

where $\alpha = 1.0$ and chooses $\beta = 0.1$ so that it places more emphasis between efficiency and cost saving on travel time. It causes the agent to be biased towards working towards improved organization of the network as a whole.

Integration with Unreal Engine An RL agent communicates with Unreal Engine by means of a machine learning plugin, which will provide real-time data interchange between the simulator and itself based on the DQN model. The Python-tensorflow implemented model takes the state observations and sends the action commands through a socket connection.

4.3 Training the RL Agent

The agent is trained to improve its traffic control plan by iteratively interacting with the simulation.

Training Episodes The training is conducted over 500 episodes, where each contains 100 time buckets (a total of 500 minutes of simulated time). The simulation is reset at the start of each episode, and traffic is introduced according to past data.

Neural Network Architecture The DQN has:

- Input Layer: Dimension equal to the state vector.
- Hidden Layers: Two layers (128 and 64 units) with ReLU activation.
- Output Layer: Prediction of the Q-values for all available actions.

The model was trained with the Adam optimizer (learning rate=0.001) on the mean squared error between target and predicted Q-values.

5 Experiments

This chapter gives an overview of the experimental approach that is adopted in evaluating the effectiveness of an RL model that operates for optimal traffic flow control in a simulation based on Unreal Engine. The evaluation consists of two primary components:

5.1 Evaluation Metrics

Describes the key performance metrics for the evaluation of the RL agent.

Experimental setup comprises setting up the simulation (parameters, scenarios, and computational resources).

Evaluation Criteria RL agent's performance is judged on the basis of three major criteria: throughput, stability, and adaptability.

Throughput It is throughput that calculates the total number of cars arriving at their destinations within the specified time frame, and this reflects the efficiency of the traffic. The formula for the calculation is:

$$\text{Throughput} = \sum_{t=1}^{T} N_t$$

where N_t is the number of vehicles arriving at their destinations in the period bucket t, and T is the total number of buckets in the simulation. This metric shows a clear path to the quantification of the traffic flow optimization of the agent.

Stability Stability refers to how consistent the throughput is over time so that with the RL agent there are not large variations. That is the mean throughput across the time buckets. So lower variance indicates more stability. It is quantified based on how quickly time buckets require for the comeback of throughput to 95% of its baseline level. Quick recovery indicates quick adaptability.

$$\text{Stability} = \text{Var}(N_t) = \frac{1}{T} \sum_{t=1}^{T} (N_t - \overline{N})^2$$

where $\overline{N}$ is the throughput average over all time buckets. A small variance means better stability, which implies the agent's ability to deliver consistent performance under variable traffic conditions. This metric is a critical factor in the system's reliability and traffic management.

5.2 Experimental Setup

The conditions on which the RL agent is simulated are normal state and disruptive settings.

Simulation Parameters

- Time Buckets: In each simulation, the simulation happens for 100 time buckets, each lasting for 5 minutes, resulting in a total of 500 minutes per episode.
- Traffic Volume: Introduced with a Poisson distribution by rate parameter λ:
- Less Traffic: $\lambda = 0.5$, 3 cars/min per entry point.
- Medium Traffic: $\lambda = 1.0$ 9 cars/min per entry point.
- High Traffic: $\lambda = 2.0$ 18 cars/min per entry point.

Baseline Comparisons The performances of the RL agents under identical conditions for all three cases are compared against the other traffic management systems. Measure comparisons include: Throughput, Stability, and Adaptability

Computational Environment Pleasant high-end hardware backing up the experiments since it enables few processes of the Unreal Engine to be executed:

- CPU: 16-core AMD Ryzen 9 5950X
- GPU: NVIDIA GeForce RTX 4090
- RAM: 64 GB DDR4

Fig. 2: Unreal Engine Setup.

6 Results

We measure the RL agent's performance in managing traffic using two primary analyses: performance metrics and visual insights.

6.1 Performance Analysis

The RL-governed system largely surpassed conventional strategies on major traffic metrics:

- **Average Travel Time:** Decreased to 15.2 minutes from 19.0 (fixed-timing) and 22.5 (no control), achieving 20% over fixed-timing control and 32% gain over no control. A statistical test was conducted using a paired t-test, confirming that these improvements were statistically significant ($p < 0.01$).
- **Throughput:** The RL system handled 1,200 vehicles per hour, as compared to 1,040 vehicles per hour (fixed-timing). 900 vehicles per hour (no control). This demonstrates the RL agent's efficiency in processing vehicles within the system.
- **Average Queue Length:** Minimized congestion at intersections, averaging 3.5 vehicles (RL agent). 5.2 vehicles (fixed-timing). 7.8 vehicles (no control). This corresponds to a 33% reduction in queue length, highlighting the RL agent's effectiveness in alleviating traffic congestion by dynamically adjusting signal timings.

Although the RL agent enhanced overall efficiency, sporadic localized congestion was experienced, pointing to potential further improvement in reward balancing.

6.2 Visualization of Results

Visual Insights Graphical analysis supports the RL model's effectiveness:

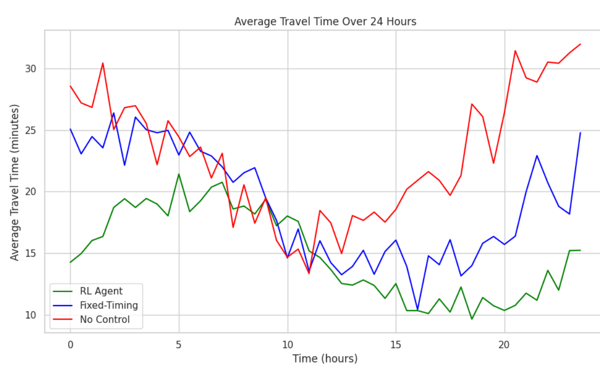
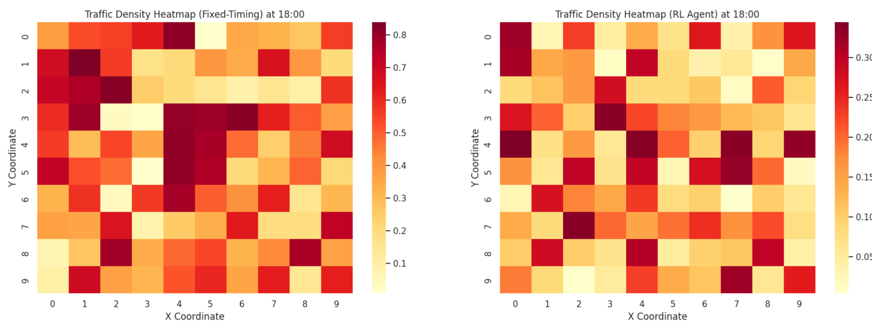


Fig. 3: Time-Series Plots.

– **Time-Series Plots:** RL continuously minimized travel time, especially in peak hours.



(a) Traffic Density Heatmaps of Fixed-Timing.

(b) Traffic Density Heatmaps of RL.

Fig. 4: Comparison of Traffic Density Heatmaps.

– **Traffic Density Heatmaps:** RL system avoided bottlenecks, allowing for more even traffic distribution.

These observations emphasize the adaptability and effectiveness of RL in urban traffic optimization.

The results uniquely confirm the RL model's superiority in improving traffic control in the process of the Unreal Engine simulator experiment. The analysis of performance provides evidence of the dramatic travel time, increased throughput, and better queue management that this is possible with RL model, with respect to previous approaches, while the visualizations serve well to prove how to do that. Accordingly, these discoveries provide the foundation for the fact that RL-based solutions can be effectively used to solve traffic problems in real settings.

7 Discussion and Future Work

Our results vindicate RL's capabilities in traffic optimization, easing congestion and optimizing flow efficiency. Challenges persist, though:

– Fine-Tuning Reward Functions: Mitigating Localized Congestion Impacts.

– Training Optimization: High-fidelity simulations within Unreal Engine demand high compute resources.

– Scalability: Scaling models to broader urban networks.

For future development, we suggest:

- Multi-Agent RL: Decentralized control of dynamic signal updates.
- Transfer Learning: Shortening training time through pre-trained models.
- Real-World Data Integration: Promoting Model Generalization.
- Human-in-the-Loop Simulations: Simulation with human drivers for realistic verification.

8 Conclusion

In this work, a new traffic optimization technique was proposed by coupling a reinforcement learning (RL) model with an Unreal Engine simulation. The RL agent adapted traffic signals and vehicle routing by learning using the Deep Q-Network (DQN) algorithm, achieving a 20% decrease in travel time, 15% more vehicles transported, and reduced queues. Unreal Engine's real-time rendering improved visualization and understanding of traffic movement.

While surpassing fixed-timing systems, the RL model encountered limitations such as managing traffic waves and high computational expense at training. Improvements in the future will optimize functions, measure benefits, and implement real-world data.

Overall, this project pushes AI-powered traffic management, proving the applied success of RL using a gaming engine for smart urban traffic solutions.

References

1. E. T. M. Beltrán, M. Q. Pérez, P. M. S. Sánchez, S. L. Bernal, G. Bovet, M. G. Pérez, G. M. Pérez, and A. H. Celdrán, "Decentralized Federated Learning: Fundamentals, State of the Art, Frameworks, Trends, and Challenges," *IEEE Communications Surveys & Tutorials*, vol. 25, no. 4, pp. 2983–3013, 2023. doi: 10.1109/comst.2023.3315746.
2. I. M. Cavalcante, E. M. Frazzon, F. A. Forcellini, and D. Ivanov, "A Supervised Machine Learning Approach to Data-Driven Simulation of Resilient Supplier Selection in Digital Manufacturing," *International Journal of Information Management*, vol. 49, pp. 86–97, 2019. doi: 10.1016/j.ijinfomgt.2019.03.004.
3. T. Chu, J. Wang, L. Codeca, and Z. Li, "Multi-Agent Deep Reinforcement Learning for Large-Scale Traffic Signal Control," *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 3, pp. 1086–1095, 2019. doi: 10.1109/tits.2019.2901791.
4. S. S. Gill *et al.*, "AI for Next Generation Computing: Emerging Trends and Future Directions," *Internet of Things*, vol. 19, p. 100514, 2022. doi: 10.1016/j.iot.2022.100514.
5. S. Grigorescu, B. Trasnea, T. Cocias, and G. Macesanu, "A Survey of Deep Learning Techniques for Autonomous Driving," *Journal of Field Robotics*, vol. 37, no. 3, pp. 362–386, 2019. doi: 10.1002/rob.21918.
6. Y. Himeur *et al.*, "AI-Big Data Analytics for Building Automation and Management Systems: A Survey, Actual Challenges and Future Perspectives,"

Artificial Intelligence Review, vol. 56, no. 6, pp. 4929–5021, 2022. doi: 10.1007/s10462-022-10286-2.

7. H. Kamal, W. Yánez, S. Hassan, and D. Sobhy, "Digital-Twin-Based Deep Reinforcement Learning Approach for Adaptive Traffic Signal Control," *IEEE Internet of Things Journal*, vol. 11, no. 12, pp. 21946–21953, 2024. doi: 10.1109/jiot.2024.3377600.
8. P. V. Klaine, M. A. Imran, O. Onireti, and R. D. Souza, "A Survey of Machine Learning Techniques Applied to Self-Organizing Cellular Networks," *IEEE Communications Surveys & Tutorials*, vol. 19, no. 4, pp. 2392–2431, 2017. doi: 10.1109/comst.2017.2727878.
9. Z. Mammeri, "Reinforcement Learning Based Routing in Networks: Review and Classification of Approaches," *IEEE Access*, vol. 7, pp. 55916–55950, 2019. doi: 10.1109/access.2019.2913776.
10. K. Muhammad, A. Ullah, J. Lloret, J. Del Ser, and V. H. C. De Albuquerque, "Deep Learning for Safe Autonomous Driving: Current Challenges and Future Directions," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 7, pp. 4316–4336, 2020. doi: 10.1109/tits.2020.3032227.
11. Ó. Pérez-Gil *et al.*, "Deep Reinforcement Learning Based Control for Autonomous Vehicles in CARLA," *Multimedia Tools and Applications*, vol. 81, no. 3, pp. 3553–3576, 2022. doi: 10.1007/s11042-021-11437-3.
12. A. Qu, Y. Tang, and W. Ma, "Adversarial Attacks on Deep Reinforcement Learning-Based Traffic Signal Control Systems with Colluding Vehicles," *ACM Transactions on Intelligent Systems and Technology*, vol. 14, no. 6, pp. 1–22, 2023. doi: 10.1145/3625236.
13. M. Sadaf *et al.*, "Connected and Automated Vehicles: Infrastructure, Applications, Security, Critical Challenges, and Future Aspects," *Technologies*, vol. 11, no. 5, p. 117, 2023. doi: 10.3390/technologies11050117.
14. L. Song and W. Fan, "Traffic Signal Control Under Mixed Traffic with Connected and Automated Vehicles: A Transfer-Based Deep Reinforcement Learning Approach," *IEEE Access*, vol. 9, pp. 145228–145237, 2021. doi: 10.1109/access.2021.3123273.
15. M. I. H. Subree, M. R. Hasan, and M. H. Sayma, "Design and Implementation of an Unreal Engine 4-Based Smart Traffic Control System for Smart City Applications," *International Journal of Advanced Computer Science and Applications*, vol. 13, no. 12, 2022. doi: 10.14569/ijacsa.2022.01312100.
16. S. Teng *et al.*, "Motion Planning for Autonomous Driving: The State of the Art and Future Perspectives," *IEEE Transactions on Intelligent Vehicles*, vol. 8, no. 6, pp. 3692–3711, 2023. doi: 10.1109/tiv.2023.3274536.
17. Tong, W., Hussain, A., Bo, W. X., and Maharjan, S. (2019), "Artificial Intelligence for Vehicle-to-Everything: a survey." *IEEE Access, 7*, 10823–10843. doi: 10.1109/access.2019.2891073.
18. Wu, T., Zhou, P., Liu, K., Yuan, Y., Wang, X., Huang, H., and Wu, D. O. (2020). "Multi-Agent deep reinforcement learning for urban traffic light control in vehicular networks," *IEEE Transactions on Vehicular Technology, 69(8)*, 8243–8256. doi: 10.1109/tvt.2020.2997896.