applied sciences

MDPI

Article

Comprehensive Automated Driving Maneuvers under a Non-Signalized Intersection Adopting Deep Reinforcement Learning

Quang-Duy Tran [1] and Sang-Hoon Bae [2,*]

[1] Faculty of Civil Engineering, Nha Trang University, Nha Trang 57000, Vietnam
[2] Department of Spatial Information Engineering, Pukyong National University, Busan 48513, Korea
* Correspondence: sbae@pknu.ac.kr

Abstract: Automated driving systems have become a potential approach to mitigating collisions, emissions, and human errors in mixed-traffic environments. This study proposes the use of a deep reinforcement learning method to verify the effects of comprehensive automated vehicle movements at a non-signalized intersection according to training policy and measures of effectiveness. This method integrates multilayer perceptron and partially observable Markov decision process algorithms to generate a proper decision-making algorithm for automated vehicles. This study also evaluates the efficiency of proximal policy optimization hyperparameters for the performance of the training process. Firstly, we set initial parameters and create simulation scenarios. Secondly, the SUMO simulator executes and exports observations. Thirdly, the Flow tool transfers these observations into the states of reinforcement learning agents. Next, the multilayer perceptron algorithm trains the input data and updates policies to generate the proper actions. Finally, this training checks the termination and iteration process. These proposed experiments not only increase the speeds of vehicles but also decrease the emissions at a higher market penetration rate and a lower traffic volume. We demonstrate that the fully autonomous condition increased the average speed 1.49 times compared to the entirely human-driven experiment.

Keywords: automated driving; comprehensive maneuvers; deep reinforcement learning

Citation: Tran, Q.-D.; Bae, S.-H. Comprehensive Automated Driving Maneuvers under a Non-Signalized Intersection Adopting Deep Reinforcement Learning. *Appl. Sci.* **2022**, *12*, 9653. https://doi.org/10.3390/app12199653

Academic Editors: Alfredo Gimelli, Daniela Anna Misul and Gabriele Di Blasio

Received: 26 July 2022
Accepted: 23 September 2022
Published: 26 September 2022

1. Introduction

Traffic collisions have a considerable effect on human health and the economy. According to the World Health Organization, 50 million people were injured and 1.5 million died in 2018 due to transport-related incidents [1]. A total of 51.1% of transport deaths were related to urban areas and 49.8% of collisions occurred in intersections, according to the road safety report of Korea in 2019 [2]. Intersections have complex safety implications, with numerous complicated trajectories—namely left turns, right turns, lane changes, stop controls, and yield controls. These intersections consist of signalized junctions and non-signalized junctions. In particular, non-signalized intersections involve more complex interactions and have higher collision rates. Furthermore, the rules of junctions are usually broken by a few aggressive human drivers. Hence, traffic safety at non-signalized intersections is an important component of urban traffic safety. Furthermore, comprehensive maneuvers need to be considered in order for the model to approximate reality. However, in this study, we limited our focus to going straight and left turns for a non-signalized intersection. More importantly, 94% of traffic collisions were mainly caused by human factors, according to data from the National Highway Traffic Safety Administration [3]. To address this issue, automated vehicles (AVs), which can maneuver without or with less human input using fusion sensors (e.g., LIDAR, radar, cameras, etc.), seem to be an effective means to improve traffic flow and mitigate human driver error [4]. The automated driving

(AD) field has seen breakthroughs due to the development of deep learning and studies on vehicle dynamics [5], as well as due to new sensor technologies, such as LIDAR [6].

The goal of micro-simulation is to represent the behaviors of vehicles regarding car-following and lane-change behaviors. Additionally, car-following models, which are a basic component of micro-simulations, can be used to describe driver behaviors under vehicle platooning. They show the behavior of following vehicles according to that of leading vehicles. The main purpose of car-following models is to generate the desired velocity and acceleration based on longitudinal components. Mathew and Ravishankar [7] implemented a car-following behavior model under a mixed-traffic environment according to longitudinal behavior. In particular, adaptive cruise control (ACC) was originally a car-following model that described the relative longitudinal distance between vehicles. The ACC model was also introduced to the advantage driver assistance system (ADAS) embedded into a Tesla autopilot vehicle in 2014. Rajamani and Zhu [8] implemented the ACC model for a semi-automated vehicle according to leading vehicles in the same lane. In addition, David [9] introduced a cooperative ACC system related to leading vehicles in the same lane as well as in different lanes. Nevertheless, they relied upon maintaining a constant distance. In order to address this issue, the intelligent driver model (IDM) suggested by Treiber and Kesting [10] has real parameters. Moreover, IDM was also used for a BMW autopilot feature.

In addition, AD obtained a higher performance with the development of the deep learning field. In particular, the challenge of AD is how to train and validate AVs in a real-world situation. To address these problems, simulation has become a positive approach as a digital twin. For example, VISSIM was used to model connected and automated vehicles (CAVs) to investigate their safety [11]. Nevertheless, VISSIM focused on parameter settings. Additionally, Wymann et al. [12] implemented the open racing car simulator (TORCS) through an application programming interface (API). The car learning to act (CARLA), which supported imitation learning and reinforcement learning (RL), was applied for training and evaluating AD [13]. A simulation of urban mobility (SUMO), which is an open-source simulator, can model various scenarios for a large area [14]. The recently developed Flow tool is an open-source framework that connects a simulation environment (e.g., SUMO, AIMSUM) and an RL library. The Flow tool was implemented for the AD in a mixed-traffic environment [15]. Thus, the fusion of SUMO and Flow represents an effective means to improve the AD domain.

Importantly, many recent innovations in artificial intelligence (AI) have tried to enhance the development of the AD domain. For instance, the long short-term memory (LSTM) algorithm was applied to improve the navigation of AVs [16]. Moreover, the RL algorithm obtained an effective performance in improving complex policies in the AD field under an uncertain environment. The RL algorithm differs from unsupervised learning and supervised learning by optimizing the reward from the observation and state. The Markov decision process (MDP), which was suggested by Bellman [17], formalized sequential decision making for AD. Hubmann et al. [18] used the multi-agent MDP in a mixed-traffic environment. However, the MDP paradigm is well-suited for full observation. For implementing AVs under dynamic conditions, the partially observable Markov decision process (POMDP) is a partial MDP observation [19]. This forecasts the surrounding participants for AD under mixed-traffic conditions. More recently, breakthroughs in deep neural networks (DNNs) have allowed us to enhance extraction features using multiple hidden layers. Based on this development, deep reinforcement learning (DRL), which integrates DNN and RL, can achieve effective decision making in dynamic environments. For instance, Tan et al. [20] used the DRL method for adaptive traffic signal control. Gu et al. [21] implemented the Double Deep Q-Network (DDQN) to achieve a better traffic signal policy. Therefore, it is necessary to study the AD in a mixed-traffic environment using the DRL method.

Regarding comprehensive movement applications at non-signalized intersections, Shu et al. [22] suggested the critical turning point method, which integrates a high-level candidate path generator through the POMDP algorithm. They considered incoming vehicle

trajectories, such as right turns, left turns, and going straight ahead. Liu et al. [23] suggested the DRL method for left-turn movements at a non-signalized intersection. They compared deep Q-learning (DQL) and double DQL based on the right-of-way rule. These results showed an improvement in the decision-making strategy, collision rate, and efficiency of traffic. Tran and Bae [24] proposed the use of a hybrid method that integrated POMDP and responsibility-sensitive safety algorithms. This method was applied to evaluate the efficiency of one automated vehicle (AV) turning left at a non-signalized intersection. Our results showed an improvement in the time performance and smoothing performance indices. However, the scenarios were simple with only one AV.

As for hyperparameter turning studies, Tran and Bae [25,26] applied a DRL to evaluate the efficiency of using AVs at a non-signalized intersection and in an urban network. We also suggested a set of proximal policy optimization (PPO) hyperparameters to improve the performance of DRL training. In particular, PPO with the adaptive Kullback–Leibler (KL) penalty was used for a non-signalized intersection, and clipped PPO was applied for an urban network. Our results presented improvements in the DRL training policy, smoothing velocity, mobility, and energy. Nevertheless, we only focused on the effectiveness of vehicles going straight ahead.

In this work, we concentrate on the efficiency of comprehensive AD maneuvers (e.g., go straight ahead and left turn) at a non-signalized junction over different market penetration rates (MPRs) and real traffic volumes. Firstly, we set initial parameters and created the left-turn simulation scenarios. Secondly, the SUMO simulator ran and exported the observations. Thirdly, the Flow tool transferred these observations into the states of the reinforcement learning agents. Next, the multilayer perceptron (MLP) algorithm trained the input data and the PPO paradigm updated policies to generate the proper actions. Finally, the DRL training checked the termination and iterated the process. The main contributions of this work are as follows:

- Considering the efficiency of comprehensive AD maneuvers (e.g., going straight ahead and left turn) at a non-signalized intersection over different MPRs and real traffic volumes by adopting the DRL method;
- Evaluating the performance of the set of clipped PPO hyperparameters in the context of comprehensive AD maneuvers for a non-signalized intersection;
- The meaningful development of traffic quality at a non-signalized intersection with a higher market penetration rate (MPR) and a lower traffic volume.

This paper is structured as follows: the car-following model, reinforcement learning algorithm, and proposed method architecture are presented in the following section. Section 3 refers to the hyperparameter setting and evaluation indicators. Then, Section 4 contains the experiments and results. Finally, the discussions and conclusions are shown in Section 5.

2. Methods

2.1. Car-Following Model

The car-following model, which is also called the time-continuous model, can simulate the behavior of a driver following a leading vehicle. Another form of car-following model is safety distance or collision avoidance. This means that the driver tries to maintain a safe distance from the leading vehicle to avoid collisions. The car-following model was determined by an ordinary differential equation that showed the position dynamics as follows:

$$\ddot{x}_a(t) = \dot{v}_a(t) = F(v_a(t), s_a(t), v_{a-1}(t)), \quad (1)$$

where s_a refers to the bumper-to-bumper distance to the leading vehicle, v_a is a host vehicle's actual speed, and v_{a-1} refers to the leading vehicle's speed.

To deal with realistic driving interactions, we applied the IDM paradigm to this traffic simulation, considering the longitudinal dynamic for the behavior of human-driven vehicles (HVs) [27]. In this work, the velocity, headway, and identification of the leading

vehicle were obtained by the "get" function in the SUMO simulator. The command of acceleration (a_{IDM}) was determined as follows:

$$a_{IDM} = a\left[1 - \left(\frac{v}{v_0}\right)^{\delta} - \left(\frac{s^*(v,\Delta v)}{s}\right)^2\right], \quad (2)$$

where v refers to the actual velocity, v_0 is the desired velocity, δ is the acceleration exponent, s^* is the desired headway, and s is the actual headway. In addition, the formula for desired headway is expressed as follows:

$$s^*(v,\Delta v) = s_0 + max\left(0, vT + \frac{v\Delta v}{2\sqrt{ab}}\right), \quad (3)$$

where s_0 refers to the minimum headway, T is the time headway, Δv is the different velocity, a is the acceleration term, and b is the comfortable deceleration.

In this study, the typical parameters for IDM were taken from Treiber and Kesting [10] and are shown in Table 1.

Table 1. The typical parameters for the IDM algorithm at a non-signalized intersection.

Parameters	Value
Desired velocity	15 m/s
Time headway	1.0 s
Minimum headway	2.0 m
Acceleration exponent	4.0
Acceleration	1.0 m/s^2
Comfortable acceleration	1.5 m/s^2
Desired velocity	15 m/s

2.2. Reinforcement Learning Algorithm

RL is one of main types of machine learning and associates automated agents with their surrounding environment to maximize the reward. The goal of the RL algorithm is to carry out appropriate decision making. To deal with uncertain conditions for AVs, the POMDP paradigm enables us to carry out appropriate decision making in an updated belief state. The major components of the POMDP algorithm consist of a set of belief states (B), actions (A), transition functions of states (T), rewards (R), sets of observations (Z), and probability functions of observations. Note that the belief states must be updated after implementing the action and obtaining the observation over time. The POMDP paradigm refers as a tuple (B, A, T, R, Z, O). The expected return relied on the Monte Carlo algorithm, as follows [18]:

$$Q(b,a) \approx R(b,a) + \gamma \sum_{o\epsilon O} \tau(b,a,o)\, V^*(\tau(b,a,o)), \quad (4)$$

$$\pi(b) := argmax_a Q(b,a), \quad (5)$$

where $\pi(b)$ refers to the optimal policy, γ refers to the discount factor that ranges from 0 to 1, V^* is the optimal value function according to the Bellman equation, τ is the transition function of the belief state, and *argmax* presents the optimal long-term return.

2.3. Proposed Method Architecture

More recently, the DNN algorithm enables automatic feature extraction through multiple layers. This means that different hidden layers can show different transformations according to their input. For instance, the artificial neural network (ANN) was designed to obtain a reliable performance from more data. The ANN paradigm tries to achieve the desired output through the input data. In this study, the MLP was designed to generate a

set of acceleration policies related to updated states converted from the SUMO simulator over time. The MLP method was suggested by Rumelhart et al. [28].

Additionally, the backpropagation process was also used to optimize the policy through the PPO algorithm. The PPO algorithm relies upon policy gradient methods to estimate the parameterized policy function to improve the convergence under nonlinear functions. The policy gradient is expressed as follows:

$$g = E[\nabla_\theta log\pi_\theta(a_t|s_t)A^{\pi,\gamma}(a_t|s_t)], \quad (6)$$

where $E[.]$ refers to the expectation operator, π_θ refers to the stochastic policy, $log\pi_\theta$ refers to the probabilities of the stochastic policy, and $A^{\pi,\gamma}$ refers to the advantage function.

The PPO algorithm, which was supported by the Flow tool through the RLlib library, ensures that the new policy is not too dissimilar from the old policy [29]. The surrogate loss function is applied to update the policy using the Gaussian distribution. The PPO paradigm consists of clipped PPO and the adaptive KL penalty. In this study, the clipped PPO, which performs well under complicated conditions, was implemented to obtain a new objective function through stochastic gradient descent (SGD).

$$L_{\theta_k}^{CLIP} = E\left[\sum_{t=0}^{T}\left[min(r_t(\theta))A_t^{\pi_k}, clip(r_t(\theta), 1 - \varepsilon, 1+\varepsilon)A_t^{\pi_k}\right]\right], \quad (7)$$

where θ refers to the policy parameter and ε refers to the clipping threshold. This algorithm considers the probability ratio between the new policy and the old policy. When this ratio is outside the variation between $(1 - \varepsilon)$ and $(1 + \varepsilon)$, the advantage function will be cut.

Through the integration of the RL and MLP algorithms, the DRL method was applied to verify the efficiency of comprehensive AD maneuvers at a non-signalized intersection with different MPRs and real traffic volumes. This study focused on a closed-loop online optimization through a simulation environment (SUMO), the Flow tool, and DRL agents. Figure 1 expresses the research proposal flow. The DRL algorithm attempts to maximize the expected cumulative discounted reward, which is defined as follows:

$$\theta* := argmax_\theta \eta(\pi_0), \quad (8)$$

where $\eta(\pi_0)$ expresses the expected cumulative discounted reward, defined as follows:

$$\eta(\pi_0) = \sum_{i=0}^{T} \gamma_i r_i, \quad (9)$$

where γ_i expresses the discount factor and r_i expresses the reward.

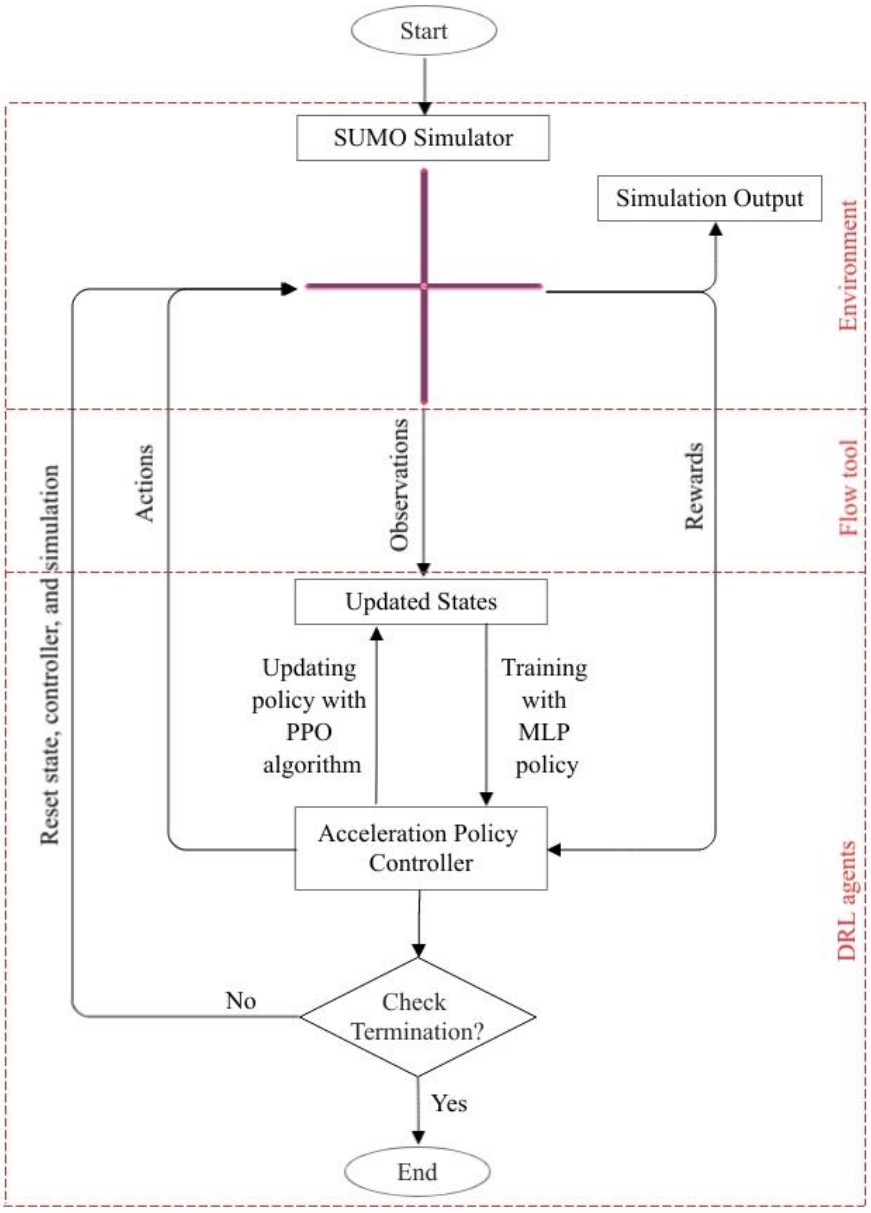


Figure 1. The research proposal flow architecture.

The SUMO simulator, which is an open-source microscopic simulation, uses a traffic control interface (TraCI) to enable the DRL method through Python programming. Figure 2 expresses a typical simulation in SUMO.

In addition, the Flow tool, which was suggested by UC Berkeley, is a bridge between the simulation environment and DRL agents. The advantage of the Flow tool is that it is easy to implement across various types of roads. This process explores the important features of state information to generate the optimal acceleration policy. The training process includes six major parts: initial setting, state, action, observation, reward, and termination.

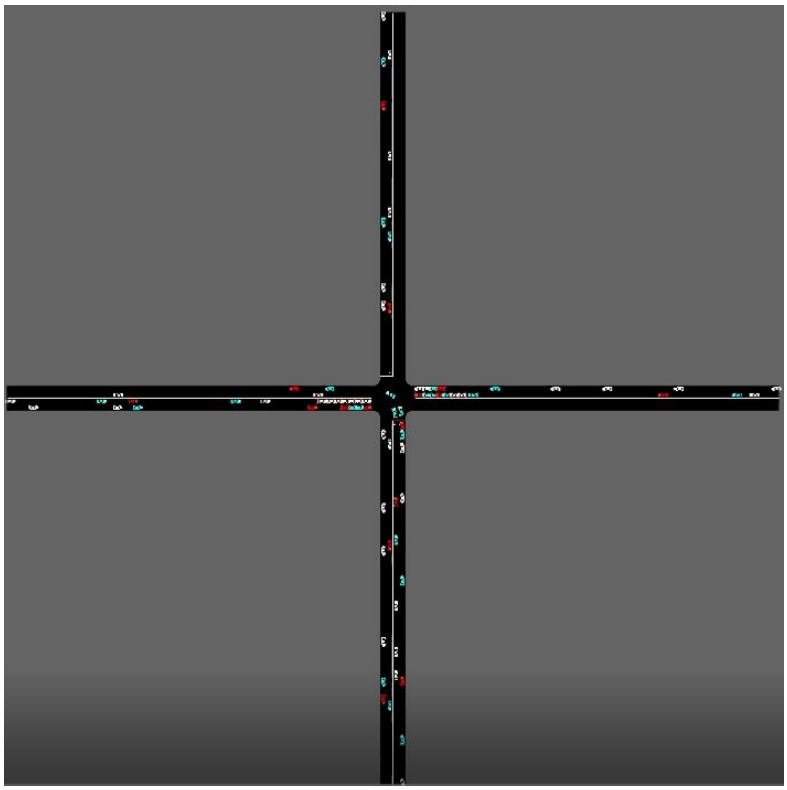


Figure 2. A typical simulation in SUMO.

2.3.1. Initial Setting

There are various factors in the initial setting, such as beginning points, acceleration, velocity, trajectories, traffic volume, IDM parameters, and PPO hyperparameters. In this study, the SUMO simulator defines the nodes, edges, and routes. Then, the Flow tool controls the acceleration policy of HVs. In addition, the RLlib library governs acceleration decision making of AVs.

2.3.2. State

The state represents the information of AVs and neighborhood vehicles—namely, AVs' positions, AVs' speeds, and the headways and speeds of leading and following vehicles. In this study, the entire identification of vehicles was obtained by the get_state function. Then, the positions and speeds of all vehicles were generated for the updated states. The states are defined as follows:

$$S = \begin{pmatrix} x_0 \\ v_0 \\ d_l \\ v_l \\ d_f \\ v_f \end{pmatrix}, \tag{10}$$

where x_0 defines the coordinate of AVs, v_0 defines the speeds of AVs, d_l defines the bumper-to-bumper headways of leading vehicles, v_l defines the speeds of leading vehicles, d_f defines the bumper-to-bumper headways of following vehicles, and v_f defines the speeds of following vehicles.

2.3.3. Action

The openAI gym, which ranges from minimum acceleration to maximum acceleration, controls the acceleration actions. This means that the decision making supports the acceleration actions of AVs under a mixed-traffic environment. In this work, the actual

action is transferred from the specific acceleration command through the apply_RL_actions function.

2.3.4. Observation

After executing the actions, the process collects the information of AVs and neighborhood vehicles—namely, AVs' positions, AVs' speeds, and the speeds of leading and following vehicles. Then, the updated state, which is the input of MLP training, is generated from these observations. Figure 3 shows a typical observation [25].

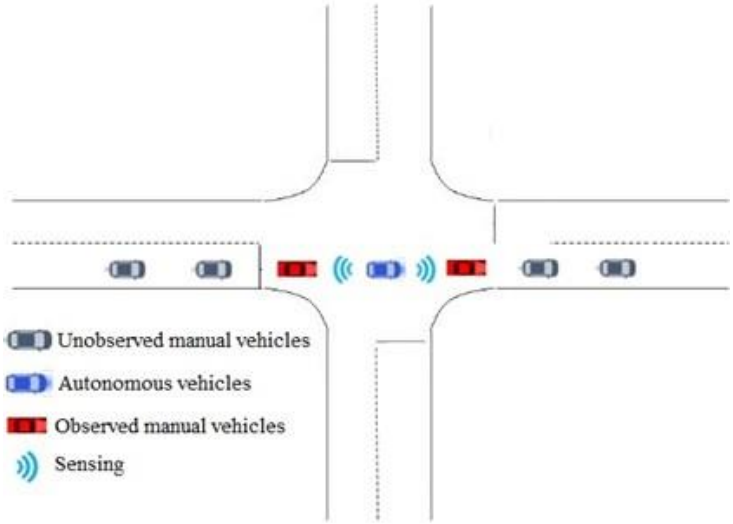


Figure 3. A typical observation.

2.3.5. Reward

The reward is the most critical coefficient used to optimize the DRL training. The average speed is an essential metric for training and evaluating the performance of the AD in real conditions. In this study, we tried to achieve a higher average speed and punish traffic collisions. The L2 norm, which estimates the vector's length in Euclidean space, was implemented to measure the positive distance regarding desired speed. Additionally, the average speed was transferred from the current speed through the get_speed function. The reward is defined as follows [15]:

$$r_t := max\left(\|v_{des}\ 1^k\|_2 - \|v_{des} - v\|_2, 0\right) / \|v_{des}\ 1^k\|_2 \quad (11)$$

where v_{des} defines the arbitrary desired speed and $v \in R_k$ defines the speeds of all vehicles at a non-signalized intersection.

2.3.6. Termination

The rollout termination is finally over when the training iteration is finished or a collision has occurred.

3. Hyperparameter Setting and Evaluation Indicators

3.1. Hyperparameter Setting

We applied a set of PPO hyperparameters to enhance the performance of the DRL method. Hyperparameter turning plays a critical role in obtaining a perfect simulation training architecture. Nevertheless, there is no specific rule to generate the best hyperparameter. To solve this problem, the trial-and-error method is the most reliable in the context of a specific scenario. In this work, we evaluated the effects of the clipped PPO hyperparameters that were used for an urban network [26]. For instance, the time horizon is 6000. The number of hidden layers significantly affects the accuracy and training time.

Many neurons in the hidden layers lead to overfitting and increase the training time. The term "256 × 256 × 256" defines setting three hidden layers with 256 neurons per hidden layer. The GAE lambda defines the smoothing rate used to maintain stable training. The clip parameter ranges from 0.1 to 0.3. The number of SGD iterations defines the epochs per training round. A set of clipped PPO hyperparameters for a non-signalized intersection is presented in Table 2 [26].

Table 2. A set of clipped PPO hyperparameters for a non-signalized intersection.

Parameters	Value
Number of training iterations	200
Time horizon per training iteration	6000
Hidden layers	256 × 256 × 256
GAE Lambda	1.0
Clip parameter	0.2
Step size	5×10^4
Value function clip parameter	10^4
Number of SGD iterations	10

3.2. Evaluation Indicators

The average reward and measures of effectiveness (MOEs) were applied to verify the performance of the DRL training. For instance, the DRL training worked better with a higher reward value over different MPRs and real traffic volumes. In the MOEs evaluation, the emissions were estimated by the Handbook Emission Factors for Road Transport (HBEFA), which defined a timeline of speeds/accelerations for each vehicle. The MOEs are defined as follows:

- Average speed: the mean value of the speed for all vehicles;
- Emissions: the mean value of emissions for all vehicles, including nitrogen oxide (NOx) and hydrocarbons (HC).

4. Experiments and Results

4.1. Simulation Experiments

A non-signalized intersection has interrupted flow. In particular, all vehicles rely upon the right-of-way rule that follows the traffic regulations and avoids collisions. The DRL agents updated the state every time step of 0.1 s. The number of iterations was 200. The continuous routing function was implemented to keep all vehicles within the network. Table 3 shows the initialized parameters for the simulation [25].

Table 3. Initialized parameters for the simulation.

Parameters	Value
Lane width	3.2 m
Number of lanes in each direction	2
Length in each direction	420 m
Maximum acceleration	3 m/s^2
Minimum acceleration	−3 m/s^2
Maximum speed	12 m/s
Horizon	600
Traffic volume	1000 veh/h

We replicated the automated decision making through the SUMO simulator, RLlib library, and the Flow tool. The simulation process tries to execute various scenarios in as real a manner as possible to obtain the appropriate interaction of AVs through a variety of assumptions. This has become a potential way to assess the AD under a mixed-traffic environment. Nevertheless, there is a variety of experiments that we must consider in

reality. Thus, we cannot model a totally real experiment. Nevertheless, within the limits of our work, these experiments considered going straight and left-turn movements that affected a non-signalized intersection. We demonstrated how an increase in the MPRs improved the performance of the proposed method. Figure 4 shows the leading AVs at a non-signalized intersection. The experiments consider platoons of mixed traffic in each direction. One vehicle platoon approaches an unsignalized junction and makes a left turn. In addition, three vehicle platoons driving straight forward follow different directions. These experiments consider mixed-traffic conditions and fully autonomous conditions.

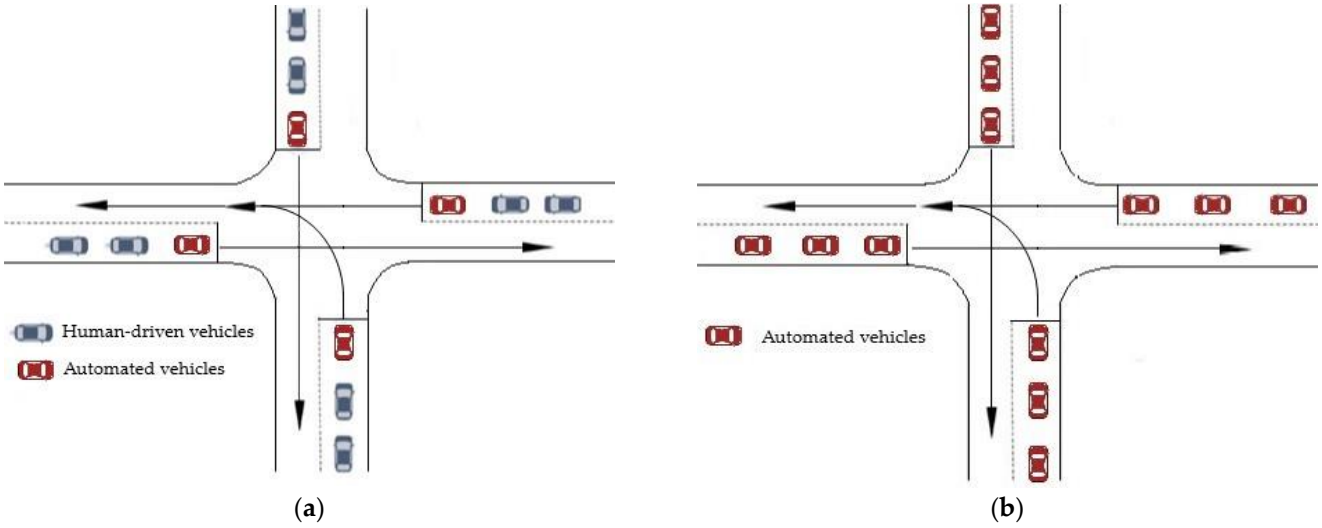


Figure 4. The leading automated vehicle experiments in the context of a non-signalized intersection: (**a**) mixed-traffic conditions with market penetration rates ranging from 20% to 80%; (**b**) fully autonomous conditions with a 100% market penetration rate.

The entirely human-driven vehicle (HV) experiment was implemented to evaluate the superiority of the leading AV experiment (the proposed experiment) based on the DRL method. This experiment corresponds to 0% MPR over different real traffic volumes, such as 100 veh/h, 500 veh/h, and 1000 veh/h. Figure 5 expresses the entirely HV experiment in the context of a non-signalized intersection.

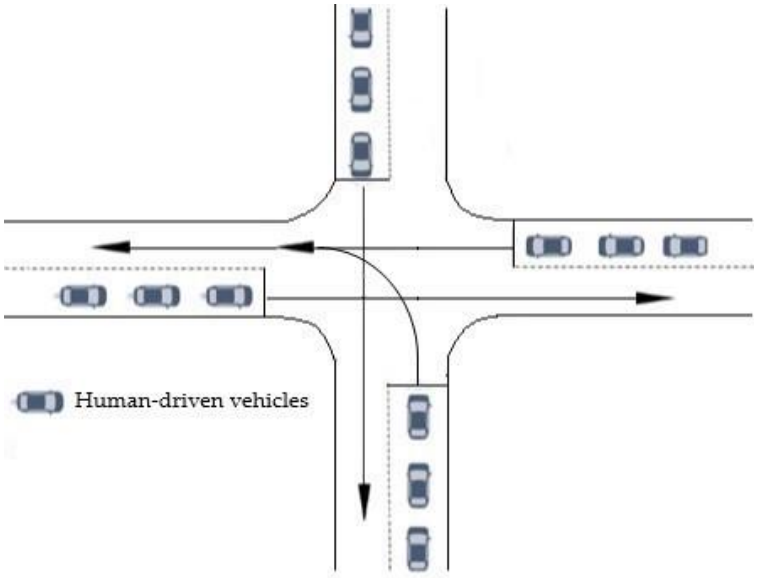
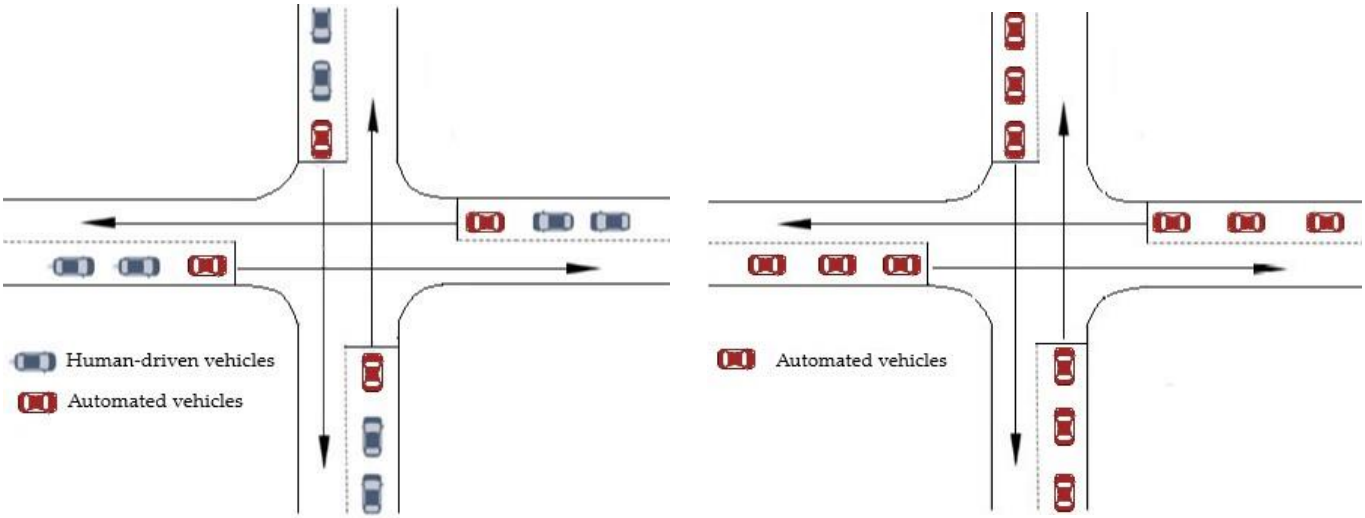


Figure 5. The entirely human-driven vehicle experiment in the context of a non-signalized intersection.

To evaluate the performance of hyperparameters and the effects of comprehensive AD movements for traffic flow, the proposed experiment was compared to the experiment with a going straight movement according to Tran and Bae [25] based on the DRL method. Figure 6 expresses a comparison experiment in the context of a non-signalized intersection.



(**a**) (**b**)

Figure 6. A comparison experiment with go straight movements in the context of a non-signalized intersection: (**a**) mixed-traffic conditions with market penetration rates ranging from 20% to 80%; (**b**) fully autonomous conditions with a 100% market penetration rate.

4.2. Experimental Results

The results express the DRL performance in terms of training policy, mobility, and emissions. We carried out the training and verified the efficiency of leading AVs under mixed-traffic conditions through different MPRs and real traffic volumes. This simulation also evaluated the effect of a set of PPO hyperparameters in the context of comprehensive AD maneuvers for a non-signalized intersection.

4.2.1. Training Policy Evaluation

To verify the DRL training policy, the average reward was implemented over different MPRs and real traffic volumes. The training policy became better as the average reward became higher. Figure 7 expresses the average reward with different MPRs and real traffic volumes. As seen in this figure, the reward achieved a higher value as the MPRs and traffic volumes increased. In particular, the fully autonomous condition outperformed the others. This means that the fully autonomous condition obtained the highest value of reward of 43,087.05 with a traffic volume of 1000 veh/h. Moreover, the fully autonomous condition increased the average reward 4.7 times compared to the 20% MPR. Therefore, the efficiency of the leading AVs was clearer at a higher MPR and traffic volume.

4.2.2. The Efficiency of Leading AVs According to Measures of Effectiveness

The MOEs were applied to verify the efficiency of leading AVs according to their mobility and emissions. The MOEs verification results over different MPRs and traffic volumes are expressed in Figures 8 and 9. Referring to mobility, the average speed improved at a higher MPR and a lower traffic volume. As seen in Figure 8, the fully autonomous condition increased the average speed 1.19 times compared to the 20% MPR. In particular, the fully autonomous condition obtained the highest value of average speed of 10.4 with a traffic volume of 100 veh/h.

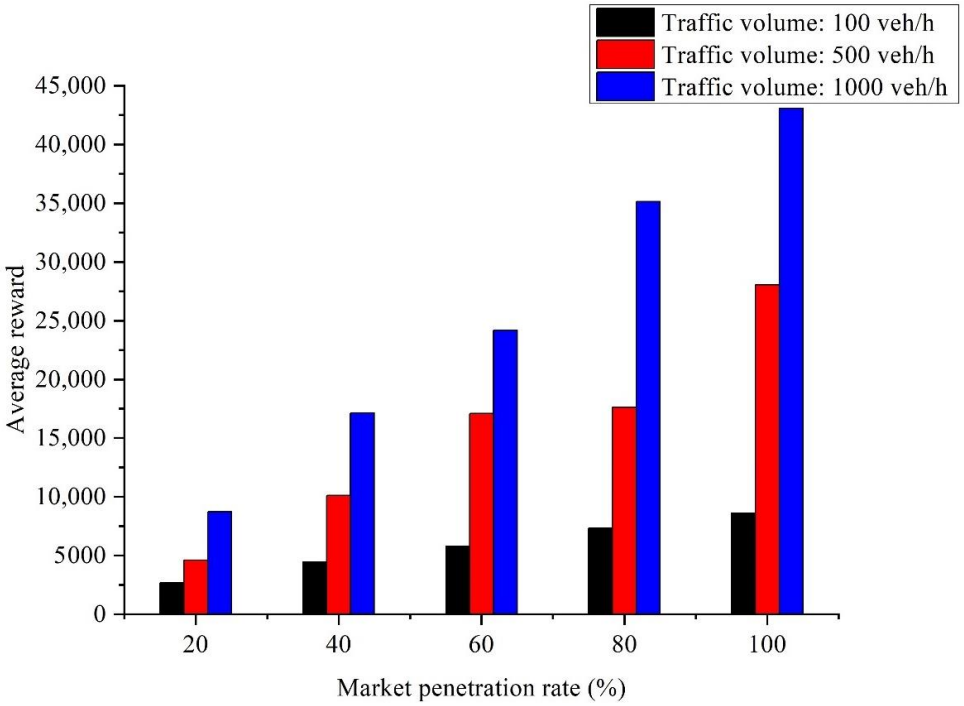


Figure 7. Average rewards with different market penetration rates and real traffic volumes.

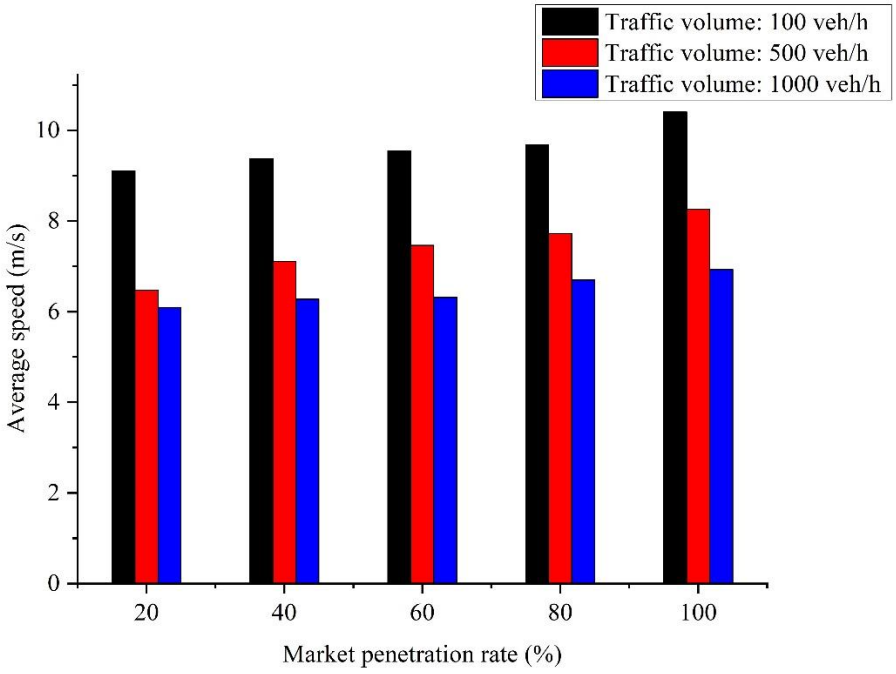


Figure 8. Average speeds with different market penetration rates and real traffic volumes.

Referring to energy, the emissions decreased at a higher MPR and a lower traffic volume. As seen in Figure 9a, the fully autonomous condition increased the NOx by 1.13 times compared to the 20% MPR. The fully autonomous condition obtained the lowest value of NOx of 1.01 with a traffic volume of 100 veh/h. As seen in Figure 9b, the fully autonomous condition increased the HC 1.46 times compared to the 20% MPR. In particular, the fully autonomous condition obtained the lowest value of HC of 0.16 with a traffic volume of 100 veh/h. Hence, the efficiency of leading AVs was clearer at a higher MPR and a lower traffic volume regarding MOEs verification.

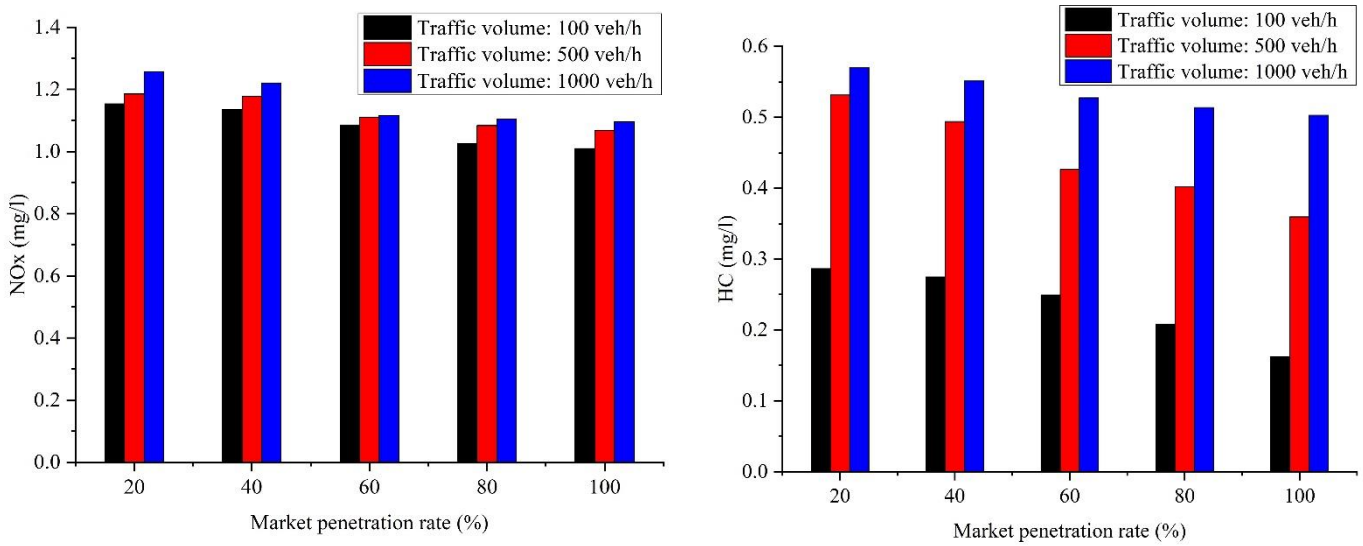


Figure 9. Emissions with different market penetration rates and real traffic volumes: (**a**) NOx; (**b**) HC.

4.2.3. Comparison of the Entirely Human-Driven Vehicle Experiment

To verify the superiority of leading AVs, the entirely HV experiment was considered over different real traffic volumes. Figure 10 shows the results of the entirely HV experiment compared to those of the proposed experiment. As seen in Figure 10a, the fully autonomous condition increased the average speed 1.49 times compared to the entirely HV experiment. In particular, the entirely HV experiment obtained the lowest value for average speed of 4.4 m/s with a traffic volume of 1000 veh/h. As seen in Figure 10b, the fully autonomous condition increased the average reward 55.70 times compared to the entirely HV experiment. In particular, the entirely HV experiment obtained the lowest value for average reward of 401.26 with a traffic volume of 1000 veh/h. Therefore, the leading AVs outperformed the entirely HV experiment regarding mobility and training policy. Moreover, the average speed and average reward decrease when the traffic volumes increase.

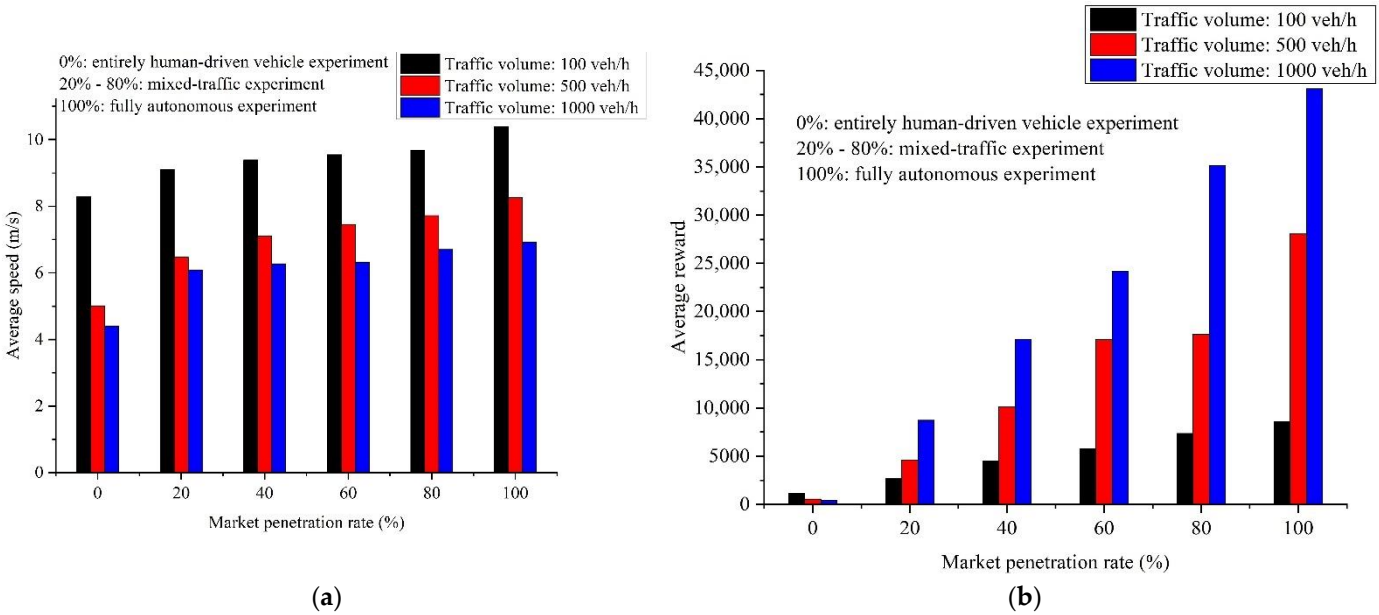


Figure 10. A comparison of the entirely human-driven vehicle experiment with different traffic volumes: (**a**) Average speeds; (**b**) Average rewards.

4.2.4. Comparison of the Experiment with Go Straight Movements

The clipped PPO hyperparameter, which was applied for an urban network with only go straight movements [26], was used for a non-signalized intersection with both left-turn and going straight movements to verify the effects of the hyperparameter on the performance of the DRL training and left-turn movements for the traffic flow. We compared the non-signalized intersection results between the proposed experiment and the experiment withgoing straight movement reported by Tran and Bae [25]. Table 4 expresses a comparison of the proposed experiment and the experiment with going straight movements with a traffic volume of 1000 veh/h.

Table 4. A comparison of the proposed experiment and the experiment with going straight movements with a traffic volume of 1000 veh/h.

Market Penetration Rates (%)	Average Speed (m/s)	
	Proposed Experiment	Experiment with Going Straight Movements
20	6.09	6.56
40	6.27	6.71
60	6.31	7.01
80	6.70	7.24
100	6.93	7.51

As seen in Table 4, the proposed experiment obtained a lower average speed compared to the experiment with going straight movements. In particular, the proposed experiment decreased the average speed 1.08 times compared to the experiment with going straight movements. Thus, the comprehensive AD movements lead to a lower speed compared to the experiment with going straight movements.

5. Conclusions

In this paper, we demonstrated that the leading AVs became more meaningful regarding training policy and MOEs at a higher MPR and a lower traffic volume. In other words, the traffic flow outperformed others at a higher MPR and a lower traffic volume. For instance, the fully autonomous condition increased the average speed 1.19 times compared to the 20% MPR. The fully autonomous condition increased the average reward 4.7 times compared to the 20% MPR. Additionally, the fully autonomous condition increased the average speed 1.49 times compared to the entirely HV experiment. In particular, the proposed experiment decreased the average speed 1.08 times compared to the experiment with going straight movements.

In summary, the leading AVs outperformed the other experiments by adopting the DRL method in the context of comprehensive AD maneuvers at a non-signalized intersection. This proposed experiment becomes more meaningful at a higher MPR and a lower traffic volume. Additionally, the leading AVs can help to mitigate delay time and emissions. The primary contribution of this work is the DRL approach for comprehensive AD maneuvers under a mixed-traffic environment. These experimental results have proven effective for the interaction between AVs and HVs in the near future through different MPRs. The comprehensive AD also makes automated decisions more realistic. In further work, we will consider the current study regarding the following aspects:

- Consider the effects of the comprehensive turning (e.g., left turn, right turn, going straight, and lane change) for an urban network. Hence, it is necessary to make complex scenarios as real as possible;
- Compare our method to other machine learning algorithms aiming to achieve better performance of decision making for AVs under a mixed-traffic environment.

Author Contributions: The authors jointly proposed the idea and contributed equally to the writing of the manuscript. Q.-D.T. designed the algorithms and performed the simulation. S.-H.B., the corresponding author, supervised the research and revised the manuscript. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. World Health Organization. Global Status Report in Road Safety. Available online: https://www.who.int/publications/i/item/9789241565684 (accessed on 17 June 2018).
2. Traffic Accident Analysis System. Available online: https://taas.koroad.or.kr/web/bdm/srs/selectStaticalReportsDetail.do (accessed on 30 December 2019).
3. Singh, S. Critical Reasons for Crashes Investigated in the National Motor Vehicle Crash Causation Survey. NHTSA's National Center for Statistics and Analysis. Available online: https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812506 (accessed on 1 March 2018).
4. Crayton, T.J.; Meier, B.M. Autonomous vehicles: Developing a public health research agenda to frame the future of transportation policy. *J. Transp. Health* **2017**, *6*, 245–252. [CrossRef]
5. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
6. Schwarz, B. Lidar: Mapping the world in 3d. *Nat. Photonics* **2010**, *4*, 429. [CrossRef]
7. Mathew, T.V.; Ravishankar, K.V.R. Car-following behavior in traffic having mixed vehicle-types. *Transp. Lett. Int. J. Transp. Res.* **2011**, *3*, 113–126. [CrossRef]
8. Rajamani, R.; Zhu, C. Semi-Autonomous Adaptive Cruise Control Systems. *IEEE Trans. Veh. Technol.* **2002**, *51*, 1186–1192. [CrossRef]
9. David, L.C. Effect of adaptive cruise control systems on mixed traffic flow near an on ramp. *Phys. A Mech. Appl.* **2007**, *379*, 274–290.
10. Treiber, M.; Kesting, A. Car-Following Models Based on Driving Strategies. In *Traffic Flow Dynamics: Data, Models and Simulations*, 1st ed.; Spring: Berlin, Germany, 2013; pp. 181–204.
11. He, S.L.; Guo, X.; Ding, F.; Qi, Y.; Chen, T. Freeway Traffic Speed Estimation of Mixed Traffic Using Data from Connected and Autonomous Vehicles with a Low Penetration Rate. *J. Adv. Transp.* **2020**, *1178*, 1361583. [CrossRef]
12. Wymann, B.; Espi'e, E.; Guionneau, C.; Dimitrakakis, R.; Sumner, A. TORCS, the Open Racing Car Simulator. Available online: http://www.torcs.org (accessed on 12 March 2015).
13. Dosovitskiy, A.; Ros, G.; Codevilla, F.; Lopez, A.; Koltun, V. CARLA: An Open Urban Driving Simulator. In Proceedings of the 1st Annual Conference on Robot Learning, Mountain View, CA, USA, 13–15 November 2017; pp. 1–16.
14. Krajzewicz, D.; Erdmann, J.; Behrisch, M.; Bieker, L. Recent development and applications of sumo-simulation of urban mobility. *Int. J. Adv. Syst. Meas.* **2012**, *5*, 128–243.
15. Vinitsky, E.; Kreidieh, A.; Le Flem, L.; Kheterpal, N.; Jang, K.; Wu, F.; Liaw, R.; Liang, E.; Bayen, A.M. Benchmarks for Reinforcement Learning in Mixed-Autonomy Traffic. In Proceedings of the Conference on Robot Learning, Zürich, Switzerland, 29–31 October 2018.
16. Bai, Z.; Cai, B.; ShangGuan, W.; Chai, L. Deep Learning Based Motion Planning for Autonomous Vehicle Using Spatiotemporal LSTM Network. In Proceedings of the 2018 Chinese Automation Congress (CAC), Xi'an, China, 30 November–2 December 2018; pp. 1610–1614.
17. Bellman, R. A Markovian Decision Process. *J. Math. Mech.* **1957**, *6*, 679–684. [CrossRef]
18. Hubmann, C.; Schulz, J.; Becker, M.; Althoff, D.; Stiller, C. Automated driving in uncertain environments: Planning with interaction and uncertain maneuver prediction. *IEEE Trans. Intell. Veh.* **2018**, *3*, 5–17. [CrossRef]
19. Song, W.; Xiong, G.; Chen, H. Intention-aware autonomous driving decision-making in an uncontrolled intersection. *Math. Probl. Eng.* **2016**, *2016*, 1025349. [CrossRef]
20. Tan, K.L.; Poddar, S.; Sarkar, S.; Sharma, A. Deep Reinforcement Learning for Adaptive Traffic Signal Control. In Proceedings of the ASME 2019 Dynamic Systems and Control Conference (DSCC), Park City, UT, USA, 9–11 October 2019.
21. Gu, J.; Fang, Y.; Sheng, Z.; Wen, P. Double Deep Q-Network with a Dual-Agent for Traffic Signal Control. *Appl. Sci.* **2020**, *10*, 1622. [CrossRef]

22. Shu, K.; Yu, H.; Chen, X.; Wang, Q.; Li, L.; Cao, D. Autonomous Driving at Intersections: A Critical-Turning-Point Approach for Left Turns. In Proceedings of the 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC), Rhodes, Greece, 20–23 September 2020; pp. 1–6.
23. Liu, T.; Mu, X.; Huang, B.; Tang, X.; Zhao, F.; Wang, X.; Cao, D. Decision-making at Unsignalized Intersection for Autonomous Vehicles: Left-turn Maneuver with Deep Reinforcement Learning. *arXiv* **2020**, arXiv:2008.06595.
24. Tran, Q.D.; Bae, S.H. Improved Responsibility-Sensitive Safety Algorithm Through a Partially Observable Markov Decision Process Framework for Automated Driving Behavior at Non-Signalized Intersection. *Int. J. Automot. Technol.* **2021**, *22*, 301–314. [CrossRef]
25. Tran, Q.D.; Bae, S.H. Proximal Policy Optimization Through a Deep Reinforcement Learning Framework for Multiple Autonomous Vehicles at a Non-Signalized Intersection. *Appl. Sci.* **2020**, *10*, 5722. [CrossRef]
26. Tran, Q.D.; Bae, S.H. An Efficiency Enhancing Methodology for Multiple Autonomous Vehicles in an Urban Network Adopting Deep Reinforcement Learning. *Appl. Sci.* **2021**, *11*, 1514. [CrossRef]
27. Milanes, V.; Shladover, S.E. Modeling cooperative and autonomous adaptive cruise control dynamic response using experimental data. *Transp. Res. Part C Emerg. Technol.* **2014**, *48*, 285–300. [CrossRef]
28. Rumelhart, D.; Hinton, G.; Williams, R. Learning representations by back-propagating errors. *Nature* **1986**, *323*, 533–536. [CrossRef]
29. Schulman, J.; Wolski, F.; Dhariwal, P.; Radford, A.; Klimov, O. Proximal policy optimization algorithms. *arXiv* **2017**, arXiv:1707.06347.