automation

Article

Decision-Making Policy for Autonomous Vehicles on Highways Using Deep Reinforcement Learning (DRL) Method

Ali Rizehvandi [1], Shahram Azadi [1] and Arno Eichberger [2],*

[1] Faculty of Mechanical Engineering, K. N. Toosi University of Technology, Tehran 15418-49611, Iran; ali.rizehvandi@email.kntu.ac.ir (A.R.); azadi@kntu.ac.ir (S.A.)
[2] Institute of Automotive Engineering, Graz University of Technology, 8010 Graz, Austria
* Correspondence: arno.eichberger@tugraz.at

Abstract: Automated driving (AD) is a new technology that aims to mitigate traffic accidents and enhance driving efficiency. This study presents a deep reinforcement learning (DRL) method for autonomous vehicles that can safely and efficiently handle highway overtaking scenarios. The first step is to create a highway traffic environment where the agent can be guided safely through surrounding vehicles. A hierarchical control framework is then provided to manage high-level driving decisions and low-level control commands, such as speed and acceleration. Next, a special DRL-based method called deep deterministic policy gradient (DDPG) is used to derive decision strategies for use on the highway. The performance of the DDPG algorithm is compared with that of the DQN and PPO algorithms, and the results are evaluated. The simulation results show that the DDPG algorithm can effectively and safely handle highway traffic tasks.

Keywords: autonomous vehicles; decision-making; deep reinforcement learning (DRL) method; overtaking; DDPG algorithm

check for updates

Citation: Rizehvandi, A.; Azadi, S.; Eichberger, A. Decision-Making Policy for Autonomous Vehicles on Highways Using Deep Reinforcement Learning (DRL) Method. *Automation* **2024**, *5*, 564–577. https://doi.org/10.3390/automation5040032

Academic Editor: Umberto Montanaro

Received: 26 August 2024
Revised: 15 October 2024
Accepted: 25 October 2024
Published: 8 November 2024

1. Introduction

Autonomous driving (AD) allows vehicles to navigate through diverse driving situations without requiring human input [1,2]. Thanks to the enormous potential of artificial intelligence (AI), self-driving cars have become a central topic in global research [3]. Numerous companies, including Toyota, Tesla, Ford, Audi, Waymo, Mercedes-Benz, and General Motors, are developing their self-driving vehicles and have made great progress in this field. Automotive researchers are also closely following the progress of self-driving car design [4]. The efficacy of self-driving cars includes four critical components: perception, decision-making, planning, and control [5].

Perception is the process of a self-driving car detecting its surroundings through the use of sensors, which include lidar, radar, cameras, GPS, and others. The decision module controls the driving behaviors of the vehicle, such as acceleration, braking, lane changes, and staying in lane. The planning module helps the self-driving car to determine the best route from one point to another [6]. Lastly, the control module directs the power transmission system's components to execute maneuvers accurately and follow the planned route. Based on the level of intelligence demonstrated by these modules, self-driving cars are classified into six levels, ranging from Level 0 to Level 5.

The strategy used in decision-making for self-driving vehicles is significant and often compared to the human brain. This strategy is generally formulated using rules derived from human driving experiences or modeled by utilizing supervised learning approaches. For instance, Song and colleagues employed a continuous Markov chain to predict the movements of nearby vehicles [7]. They then used a partially observable Markov decision process (POMDP) to develop the overall decision-making framework. In addition, decision-making capabilities were enhanced for urban road traffic scenarios [8]. The decision-making policy outlined in this study considers multiple criteria to assist city vehicles in making practical and logical choices amid various traffic conditions. Lane change decision strategies for connected automobiles were investigated in [9]. Moreover, the authors of [10] emphasized the idea of a driving system that mimics human behavior and is capable of adapting driving decisions by taking into account the needs and preferences of human drivers.

The deep reinforcement learning (DRL) method is a powerful tool for addressing long sequential decision-making problems. In recent years, many studies have explored the application of deep reinforcement learning in the field of automated driving. For example, Duane and colleagues proposed a hierarchical structure for learning decision policies using reinforcement learning (RL) methods. Additionally, researchers in [11,12] have used deep reinforcement learning (DRL) techniques to address challenges related to collision avoidance and trail sequencing in self-driving cars. The findings indicate that the deep reinforcement learning (DRL) approach outperforms traditional reinforcement learning (RL) methods for both challenges.

In addition to route planning, researchers have also considered fuel consumption for self-driving vehicles in [13,14]. They have developed an algorithm called Deep Learning Q (DQL) which has proven to be efficient in executing driving tasks. Han and his team used the DQL algorithm to determine lane changes or lane keeping for connected autonomous vehicles, incorporating feedback from nearby vehicles as network-informed knowledge [15]. This policy enhancement helped improve traffic flow and driving comfort. However, conventional deep reinforcement learning (DRL) techniques face difficulty in addressing highway overtaking challenges due to the continuous operational space and extensive range of possible scenarios [16].

In [17], the author examined the vehicle lane change process, which consists of two stages: the lane change decision and lane change movement. The author proposed a double-layer deep reinforcement learning structure in which the upper structure deep Q network (DQN) controls the decision-making process and sends lane change information to the lower deep deterministic policy gradient (DDPG) for vehicle trajectory control. After the lane change process, the DQN undergoes cooperative optimization based on the feedback of vehicle position information before and after the lane change.

Moreover, in [18], reinforcement learning has been widely recommended for use in the field of unmanned driving; however, developing the stability of unmanned vehicles and satisfying the demands of path tracking and vehicle obstacle avoidance under various operating conditions remains a challenging issue. In this paper, a control strategy for unmanned vehicles, based on a DDPG algorithm, was proposed to address the functional requirements of path tracking and obstacle avoidance. The focus of the strategy is on preventing collisions for unmanned vehicles.

In [19], the authors proposed a DRL-based motion planning strategy for traffic management in highway conditions where AV is integrated into two-way traffic and realizes the lane change maneuver. The AD system incorporates the DRL model using the end-to-end learning approach. They have created an enhanced DRL algorithm utilizing the DDPG with clearly defined reward functions.

Moreover, a lane change-tracking control model was proposed based on the deep reinforcement learning algorithm and simulation experiments were carried out to solve the problem automatic driving vehicles carrying out safe lane changes on highways. A model of the vehicle lane change path was built by using a quintuple polynomial approach with error-tracking functions. A three-degrees-of-freedom vehicle dynamics model was fused with the deep reinforcement learning framework to build the lane change path tracking control model, which was updated using a deep deterministic policy gradient (DDPG) algorithm. The algorithm learned the steering angle required for an optimal lane change path tracking to control the vehicle to complete the lane change process [20].

The vulnerability of Deep Q-Network amplifier learning algorithms and the DDPG against black-box attacks is examined by the authors in [21]. They utilize zero-order optimization methods like Zo-Signs which enable effective attacks without gradient information, revealing vulnerabilities in existing systems. Their findings indicate that these attacks have the potential to significantly decrease AV performance and diminish rewards by 60% or more. Additionally, they explore hostile training as a defensive strategy to enhance the robustness of DRL algorithms, Figure 1, finding that the performance is affected, although there is a trade-off.

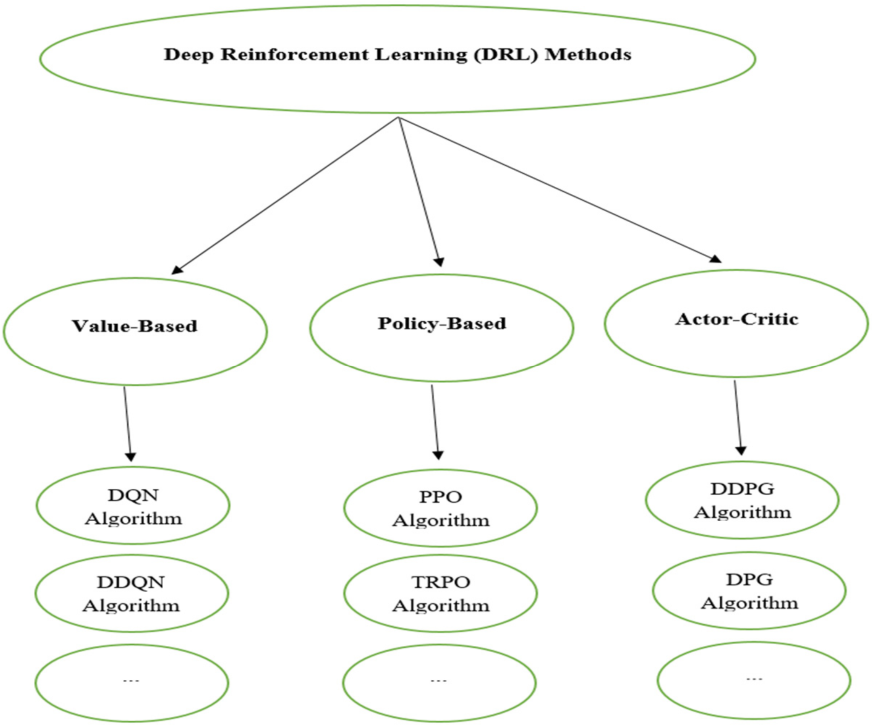


Figure 1. DRL algorithms.

In addition, ref. [22] suggests an advanced adaptive cruise control system with lane changing assistance (LCACC) for an articulated vehicle. This study employs a two-tier hierarchical control structure. The upper layer generates high-level commands, while the lower layer encompasses two modified DDPG networks, which control the steering and throttle/brake based on the commands from the upper layer. The lateral and longitudinal control of the vehicle are separated and managed by two modified DDPG networks. By appropriately designing the state and reward function, the actions of the articulated vehicle, such as steering and acceleration/braking, are more akin to human actions, thereby ensuring a comfortable ride.

The current paper presents a driving policy for self-driving cars using a deep reinforcement learning approach. The policy is designed for overtaking in highway traffic scenarios and ensures both safety and efficiency in complex environments.

The study starts by defining the driving scenario on a highway, to guide the agent safely and effectively. Then, a hierarchical control structure is introduced, which oversees both the lateral and longitudinal movements of the agent and other vehicles around it. Finally, the study uses the DDPG algorithm, a specialized deep reinforcement learning (DRL) technique, to develop a decision-making strategy specifically designed for highway traffic environments.

Finally, we evaluate and discuss the performance of the proposed simulated control framework. Figure 2 depicts the data learning methodology adopted in this study.

The study presents the following primary innovations and contributions:

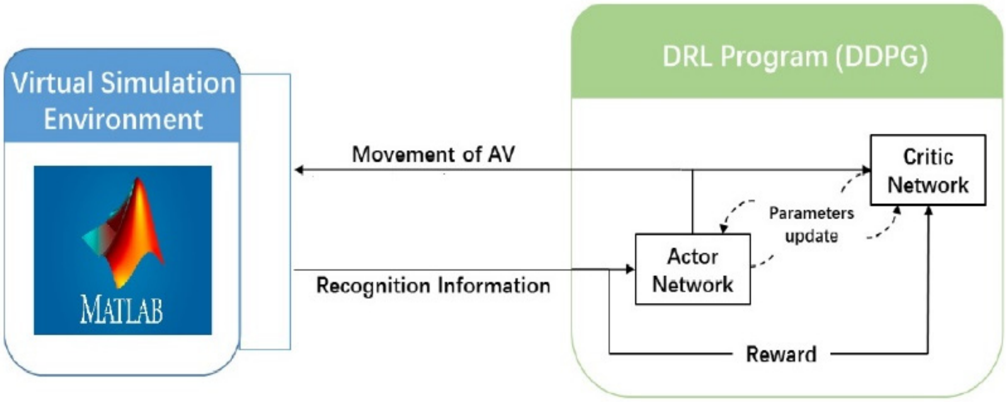


Figure 2. Training process.

The research introduces an optimal lane change strategy for self-driving cars within complex dynamic traffic. It is based on a deep reinforcement learning approach (DRL), where the decision-making phase is carried out using the DDPG algorithm. Here, lane change optimality is defined for vehicle safety and travel time. To the best of the author's knowledge, this is the first time that DDPG was used for this application.

The current study is organized into several sections. In Section 2, you will find an overview of the highway driving environment, which includes information on the operational control modules and surrounding vehicles. Section 3 focuses on the DDPG algorithm, providing a detailed discussion of the parameters of the reinforcement learning framework (RL). In Section 4, you will find the evaluation of the results. Finally, Section 5 concludes the study.

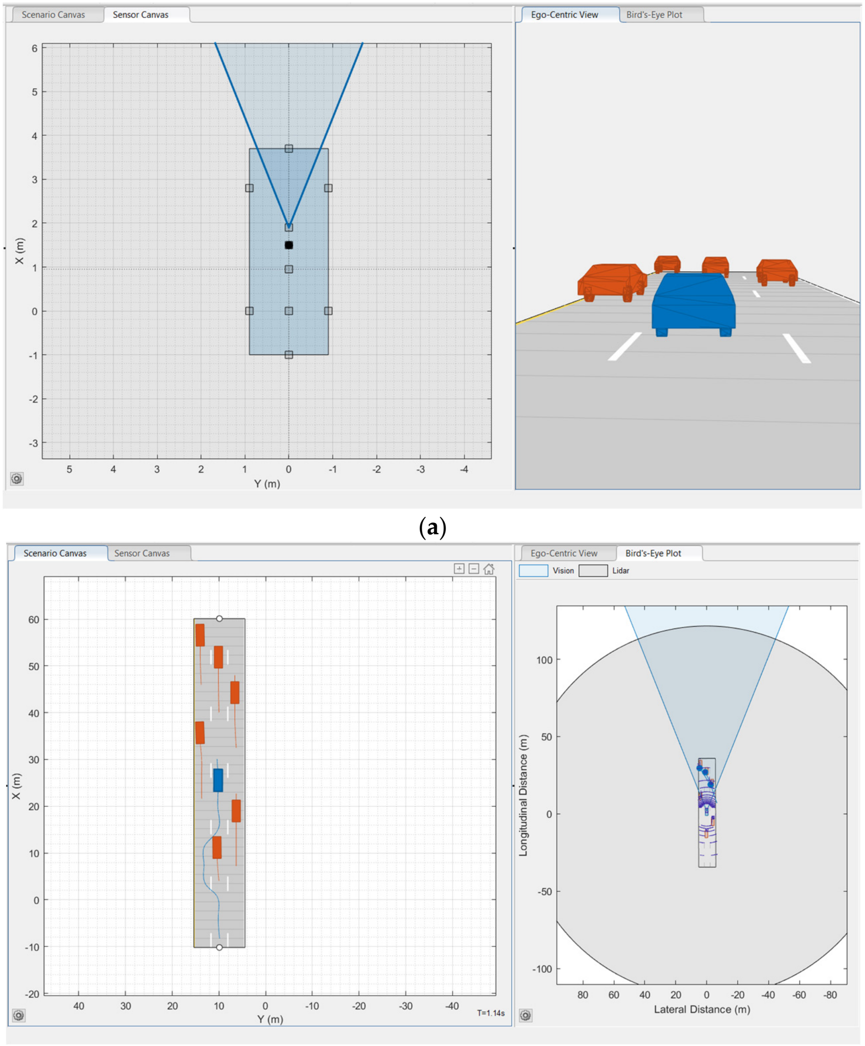
2. Driving Environment

The following section describes the driving scenario that was analyzed on the highway. To create this scenario, we used MATLAB software (version 2022) to construct a three-lane highway environment. After that, we designed the agent and the surrounding traffic environment. Furthermore, we introduced a hierarchical motion controller to monitor the lateral and longitudinal movements of both the agent and the surrounding vehicles.

When driving, we make decisions on the best way to get to our destination. This involves several behaviors such as changing lanes, keeping in our lane, accelerating, or braking. Our main goals are to avoid accidents and drive efficiently. Overtaking is a common behavior, which includes accelerating and passing other vehicles.

This passage discusses how self-driving cars make highway decisions, using the driving scenario shown in Figure 2. In the picture, the orange car is the self-driving car, while the green cars are the surrounding cars. The self-driving car starts driving at random speeds in the middle lane. Its goal is to drive as efficiently as possible while avoiding collisions with other cars. The decision-making algorithm is evaluated based on how well it balances these two goals. The speeds and starting positions of the surrounding cars are randomly selected to reflect the uncertainties of real traffic. At the beginning of the task, all the other cars are in front of the self-driving car, two per lane. From start to finish, the entire process is called an episode in this article. First, the three-lane highway is created. Then, the self-driving car and the surrounding cars are added to the environment. Finally, the self-driving car is equipped with sensors like cameras and LiDAR (as shown in Figure 3).

The subsequent section introduces a deep reinforcement learning (DRL) approach to facilitate the learning process and establish the highway decision policy.



(**a**)

(**b**)

Figure 3. Sensor modeling and highway traffic environment in MATLAB software. ((**a**): Camera and Lidar integration, (**b**): Traffic environment modeling).

3. Method

In the present research, deep reinforcement learning (DRL) techniques are applied to facilitate decision-making for an agent navigating within a highway environment.

3-1 Deep Reinforcement Learning (DRL) Method

Machine learning (ML) is a branch of artificial intelligence (AI) that concentrates on enhancing the efficiency of computational algorithms using data [19]. ML falls into three primary categories: supervised learning, unsupervised learning, and reinforcement learning (RL). In RL, an autonomous agent learns to perform tasks within an environment by seeking to maximize a predetermined reward function. The agent is rewarded when it takes the appropriate actions while interacting with its environment. On the other hand, if the chosen action is unfavorable, the agent is either penalized with negative rewards or punishments.

Supervised learning involves learning from labeled examples provided by experts. However, this method is not well-suited for solving interactive problems because accurately labeling interactions can be complex [20].

Unsupervised learning, on the other hand, focuses on discovering hidden structures within unlabeled data. While uncovering such structures can be advantageous, this approach cannot optimize rewards, which is a key objective of reinforcement learning (RL) [20].

Reinforcement learning (RL) addresses problems characterized by vast numbers of actions and states within an environment. Function approximators, such as Artificial Neural Networks (ANN), can be employed to handle the challenges posed by a large state spaces and action spaces. The utilization of a neural network as a function approximator in RL is referred to as deep reinforcement learning (DRL).

Markov decision processes (MDPs) typically structure RL problems, incorporating a set of states, a set of actions, a transition function (T) between states, and a reward function (R) [21], often expressed as the tuple (S, A, T, R). The likelihood of transitioning from states at time step t, after taking action a, resulting in a new state s + 1, is denoted as T(s_t, a_t, s_{t+1}) and ranges between 0 and 1. Immediate rewards from this transition are represented as R (s_t, a_t, s_{t+1}) and r_t, respectively. The graphic in Figure 4 offers a visual representation of the fundamental components of the RL model for autonomous vehicles.

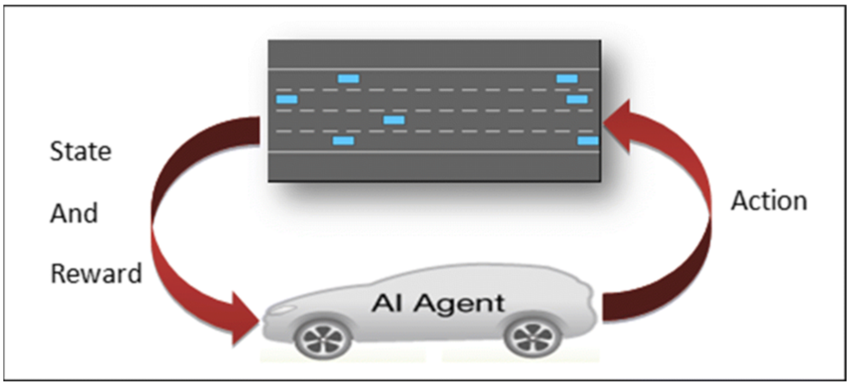


Figure 4. RL model for autonomous vehicle [23].

The expected discounted return R_t after time step t can be defined as:

$$R_t = \sum_t^{\infty} \gamma^t.r_t \tag{1}$$

where γ signifies a discount factor within the range [0, 1]. The discount factor γ falls within the range of 0 to 1. T's value can be finite or infinite (∞) depending on the specific problem. The assignment of action probabilities to states is referred to as a policy π (a | s). The value function v_π (s) represents the expected return under policy π from state s and is formulated as:

$$V^{\pi}(s_t) = E_{\pi}[R_t|S_t, \pi] \tag{2}$$

The action–value Q (s, a) function is as follow:

$$Q^{\pi}(s_t, a_t) = E_{\pi}[R_t|S_t, a_t, \pi] \tag{3}$$

which also contains the iterative Bellman equation:

$$Q^{\pi}(s_t, a_t) = E_{\pi}[r_t + \gamma \; maxQ^{\pi}(s_{t+1}, a_{t+1})] \tag{4}$$

However, some RL problems cannot be expressed as Markov decision processes (MDPs). In some cases, the states may not be fully visible or directly observable from the environment. In such situations, problems can be formulated as partially observable Markov decision processes (POMDPs). One approach to addressing these issues involves leveraging past knowledge and incorporating previous observations along with current ones to treat the problems as MDPs [20]. For instance, in Atari Games, observations can be derived from four consecutive images [3]. The primary goal of RL is to learn a policy that maximizes the expected returns. DDPG is an algorithm for continuous action domains that operate out-of-policy. It comprises two main components: learning a Q function through a critic network and learning a policy through a policy network. The Q learning aspect of the algorithm aims to approximate the optimal Q function Q∗(s, a) as shown in Equation (4) by minimizing Equation (5) With a critic network Q(s, a) with parameters φ and aggregated experiences d containing tuples (s, a, r, s′, a′), the Mean Squared Bellman Error (MSBE) can be characterized as:

$$(\varnothing, \theta) = E_D\left[\left(Q_{\varnothing}(s, a) - (r + \gamma\,(1 - d) max Q_{\varnothing}(\acute{s}, \acute{a}))\right)^2\right] \tag{5}$$

The policy network aims to learn a deterministic policy, u_{ϕ} (s), in order to select actions that maximize Q^*(s, a). To achieve this, we can utilize gradient ascent. However, to maintain stability during the learning process, the DDPG algorithm utilizes a replay buffer and target networks. These target networks comprise a target critic network and a policy target network. Although the structures of both networks mirror the originals, the parameters ϕ trail those of the originals. We update the target networks using Polyak averaging, which enables them to keep pace with the primary networks slowly and enhances stability. The target networks can be updated using Polyak averaging, enabling them to gradually align with the primary networks and enhance stability.

3-2 Parameter specification

To develop a DDPG decision-making approach, we need to create some variables that can simulate a driving scenario. In this simulation, the control mechanisms are the throttle and steering angle of the vehicle. We also need to consider the state variables, which include the relative distance and velocity differences between the agent vehicle and its surrounding counterparts, as outlined in relations (6) and (7).

$$\Delta S = \mid S_{ag} - S_{su} \mid \tag{6}$$

$$\Delta V = \mid V_{ag} - V_{su} \mid \tag{7}$$

In this particular context, ‘s’ and ‘v’ refer to the positional and speed data that are obtained from the vehicle’s dynamics. The indices ‘ag’ and ‘su’ are specifically used to denote the agent and the surrounding vehicles. It is essential to note that Equations (6) and (7) can also be interpreted as components of the P-transfer model, which is used within the framework of the reinforcement learning method (RL).

The study’s reward function takes into account three main factors: efficiency, safety, and driving objectives. The agent’s primary goal is to drive at the highest possible speed while staying in the correct lane and avoiding collisions with other vehicles on the road. At each time step (t), the reward is calculated using a formula (Equation (8)) that considers these factors.

$$R_t = -100.\text{collision} + 40(L - 1)^2 - 10(V_{ag} - V_{ag\,max})^2 \tag{8}$$

In this study, a decision-making strategy for autonomous vehicles is proposed, simulated, trained, and evaluated using MATLAB software. The environment consists of three lanes and six surrounding vehicles, where a collision is denoted by {0, 1} to indicate if the agent has encountered a collision or not. The lane count is represented by {1, 2, 3}, indicating the specific lane number on the highway. The training and evaluation process involves a value network with 128 layers and a total of 100 episodes, where the discount factor and learning rate are set to 0.8 and 0.2, respectively.

In this study, Simulink software (2022 version) was used to simulate the overtaking scenario, as shown in Figure 5.

In the following section, we will evaluate and confirm the effectiveness and validity of the proposed decision-making algorithm.

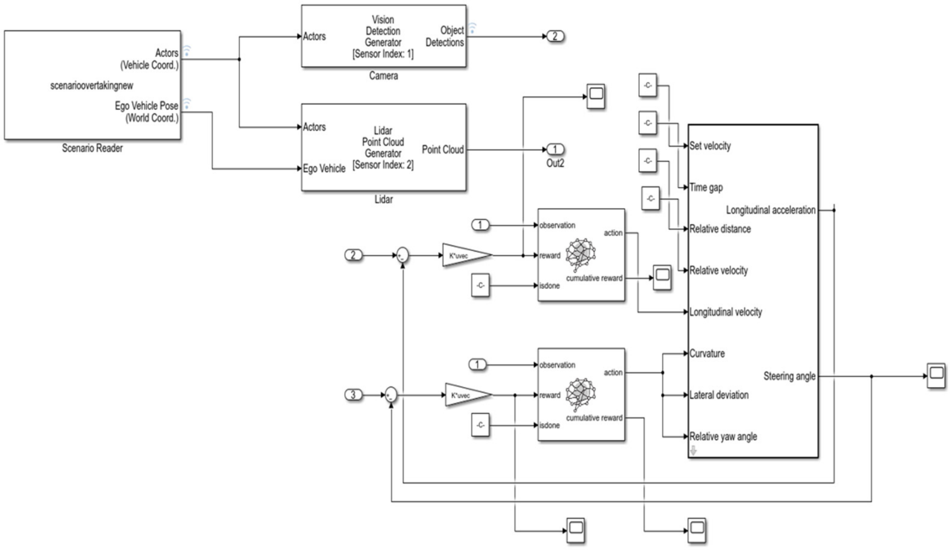
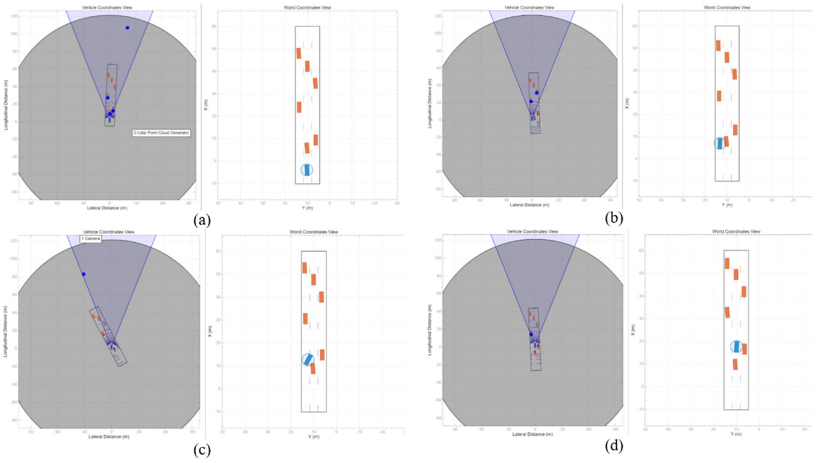


Figure 5. Block diagram.

4. Discussion

In this section, we assess and examine the control function of the DDPG algorithm proposed for the decision-making process of an agent in a highway traffic environment. Firstly, we compare and verify the effectiveness of the decision policy with another method in the evaluation. The simulation results indicate that the decision policy is optimal. Secondly, we prove the ability of the proposed DDPG algorithm to learn by analyzing the accumulated rewards. Figure 6 displays the overtaking exercise performed by the agent (blue car) in the traffic environment designed for the highway.



(a)

(b)

(c)

(d)

Figure 6. Overtaking maneuver in the highway environment. ((**a**): preparing for overtaking; (**b**): executing overtaking; (**c**): executing overtaking; (**d**): terminating overtaking).

4-1 Effectiveness of the DDPG algorithm

This section introduces three methods for decision-making in a highway traffic environment. Firstly, we evaluate the deep Q-learning algorithm (DQN) and proximal policy optimization (PPO), followed by an examination of the proposed DDPG algorithm. We also demonstrate the advantage of the DDPG algorithm over the DQN and PPO algorithms in this section. It is worth noting that we consider the parameters of the three deep learning algorithms, DDPG, DQN, and PPO, to be the same.

The performance of the control policy in the deep reinforcement learning method (DRL) is indicated by the total reward earned in each episode. Figure 7 shows the average reward in the three deep learning methods, DDPG, DQN, and PPO, over 25 episodes.

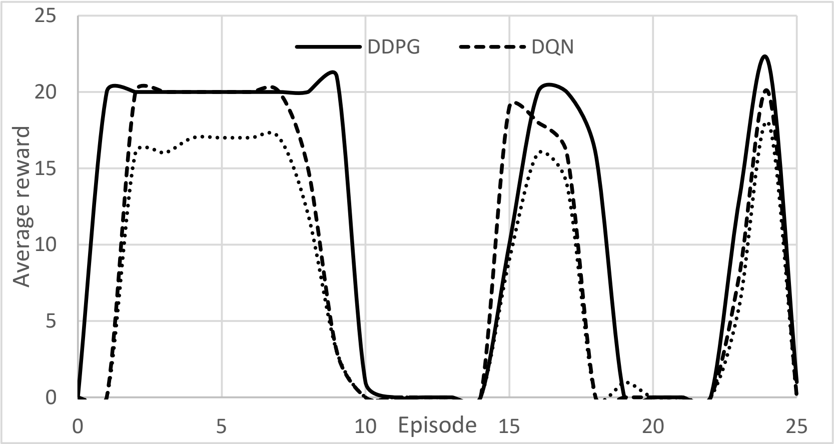


Figure 7. Average reward; DDPG, DQN, and PPO methods.

Based on Figure 7, the curves show the incremental progress of the agent's performance in interacting with the environment. However, the curve may decrease due to the complexity of the traffic environment designed on the highway, which can cause the agent to collide with surrounding vehicles or cross the lines on the highway during the overtaking exercise. According to Figure 6, the DDPG algorithm has a better learning rate for executing the overtaking maneuver designed in the highway environment compared to the DQN and PPO algorithms.

The study focuses on using the vehicle's speed and distance as state variables. Figure 8 illustrates the operating distance values obtained using the DDPG, DQN, and PPO methods. Also, Figure 8 displays the longitudinal speed values of the agent over 25 s, using the DDPG, DQN, and PPO methods.

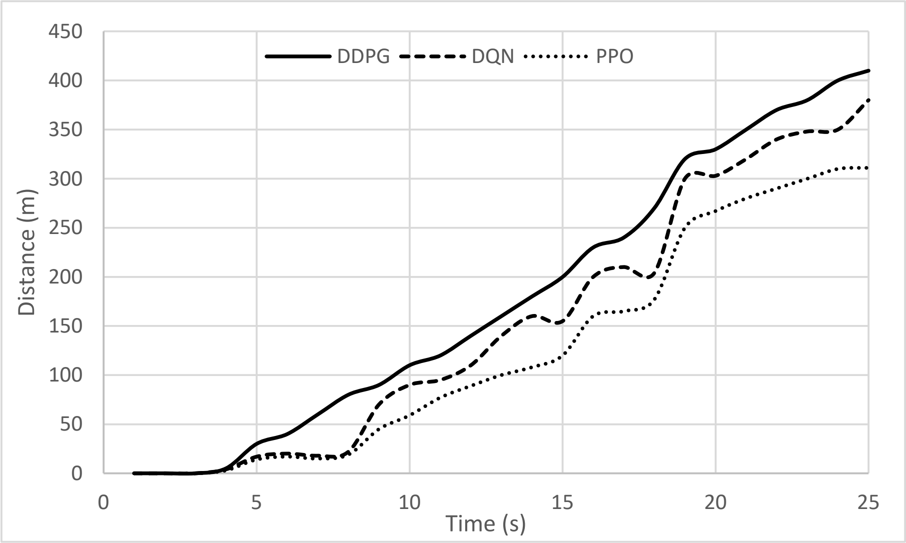


Figure 8. Agent distance; DDPG, DQN, and PPO methods.

Based on Figure 8, it appears that the control measures implemented by the DDPG agent enabled the vehicle to travel further and avoid collisions, as indicated by the larger displacement.

According to Figure 9, a higher speed leads to greater rewards, indicating that the DDPG approach is more effective than the DQN and PPO algorithms in the highway environment. On the other hand, based on the simulation results shown in Figures 6–8, the DDPG technique is more effective in achieving safety and efficiency goals in the highway traffic environment compared to the DQN and PPO algorithms.

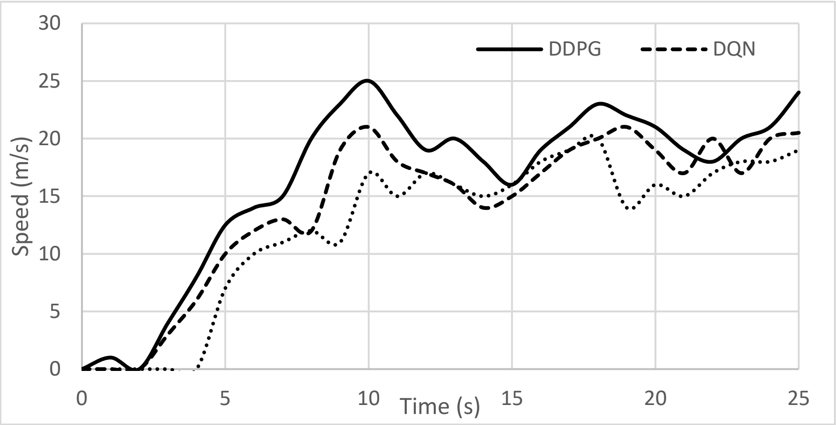


Figure 9. Agent Speed; DDPG, DQN, and PPO methods.

4-2 learning rate of the DDPG algorithm

This section is dedicated to discussing the learning rate and convergence rate of the DDPG algorithm. As mentioned before, the main goal of deep reinforcement learning algorithms is to update the Q-action value function Q (s, a) using different methods. Figure 10 shows a graph of longitudinal acceleration using DDPG, DQN, and PPO algorithms for 25 s.

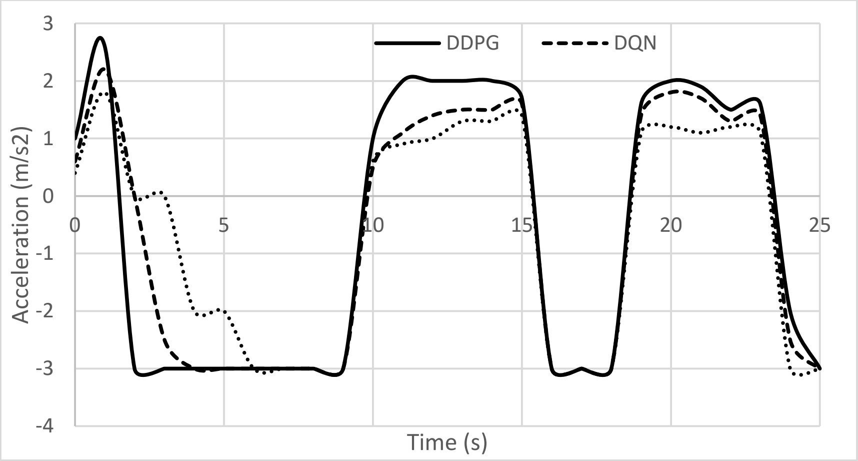


Figure 10. Agent acceleration; DDPG, DQN, and PPO methods.

Based on the analysis presented in Figure 10, it is evident that the agent in the DDPG algorithm is more familiar with the driving environment when compared to the agent in the DQN algorithm. This implies that the DDPG algorithm has a faster convergence rate for the highway decision-making problem, as compared to the DQN algorithm. Moreover, in the DDPG algorithm, the acceleration value of the agent is higher than that of the DQN algorithm, indicating that the agent is more inclined towards maintaining speed, which reflects the higher efficiency of the DDPG algorithm in the highway environment.

To compare the learning rates of the DDPG and DQN algorithms, Figure 10 shows the cumulative rewards obtained by the two methods.

According to the data presented in Figure 11, the DDPG algorithm exhibits consistently higher cumulative reward values compared to the DQN algorithm. This indicates that the control policy employed by the DDPG algorithm is superior and that the DDPG method can better comprehend the driving environment. Essentially, the agent in the DDPG algorithm is capable of searching for the optimal control policy at a faster rate.

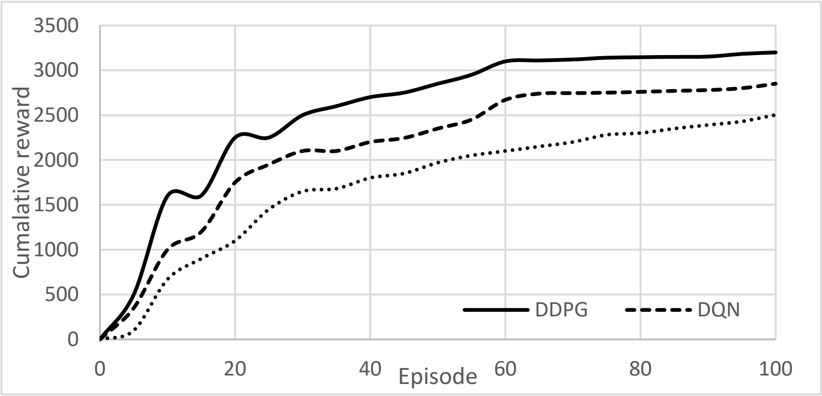


Figure 11. Agent cumulative reward; DDPG, DQN, and PPO methods.

Table 1 presents a comparison of the three mentioned algorithms.

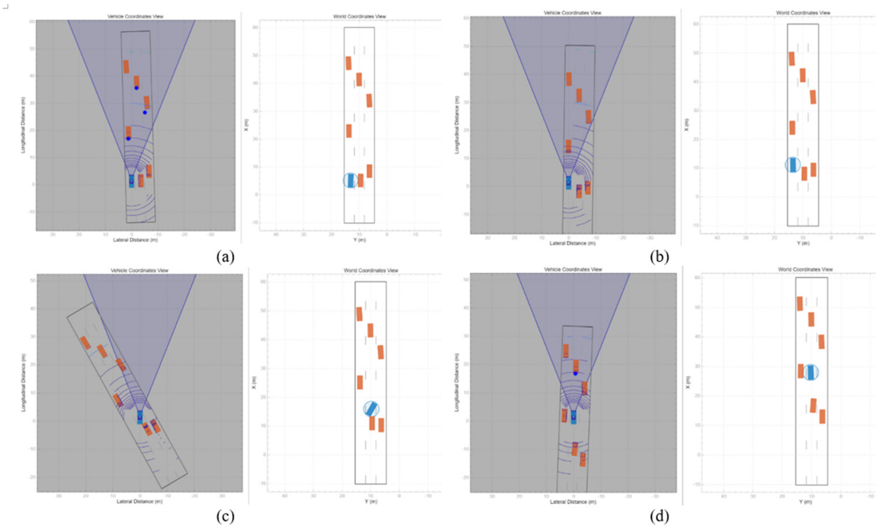
Table 1. Comparison of DDPG, DQN, and PPO algorithms.

Parameter	DDPG	DQN	PPO
Max distance (m)	410	380	311
Max speed (m/s)	25	21	20
Max acceleration (m/s^2)	2.6	2.2	1.8
Max average reward	22	20	18

The DDPG algorithm's adaptability to changing parameters in driving scenarios will be assessed through two new scenarios, with their performances being evaluated and explained.

In the first scenario, the overtaking agent was positioned next to the orange car to perform the overtaking maneuver. The results indicate that the maneuver was executed successfully without any collisions. However, due to the inherently risky nature of overtaking, the agent received only a minimal reward.

The low reward in the overtaking exercise can be attributed to two factors. Firstly, the agent crossed the lane lines during the maneuver. Secondly, there is a high probability of an accident during the same maneuver. Figure 12 demonstrates how the agent executed the maneuver.



(a)

(b)

(c)

(d)

Figure 12. Dangerous overtaking maneuver in the highway environment. ((**a**): preparing for overtaking; (**b**): executing overtaking; (**c**): executing overtaking; (**d**): terminating overtaking).

In Figure 13, the reason for the agent receiving a lower reward while performing the maneuver is depicted.

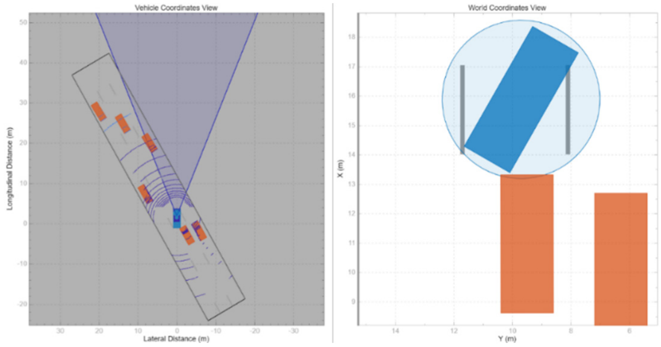
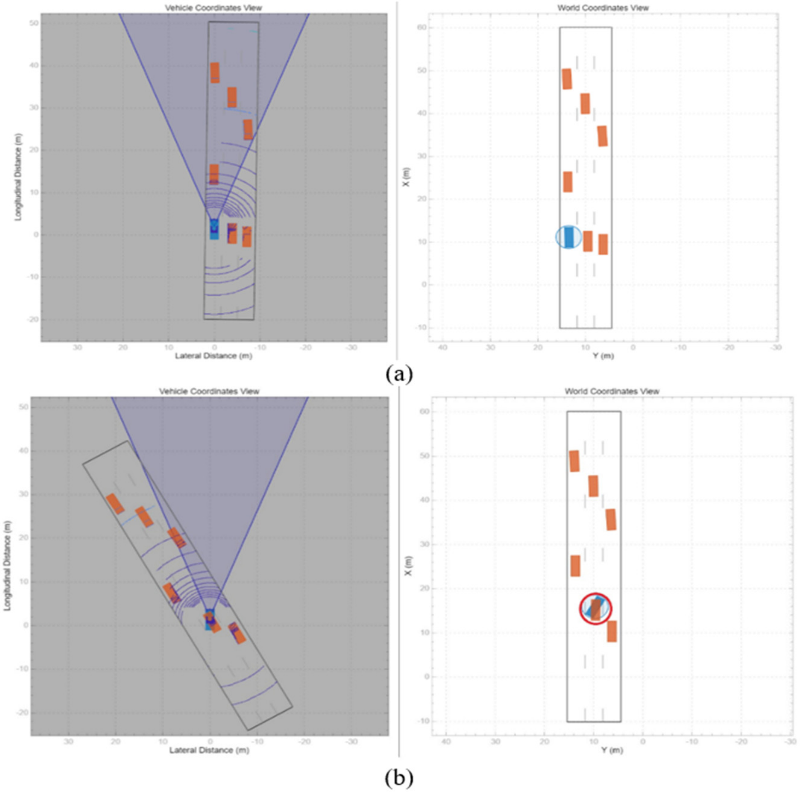


Figure 13. Dangerous action in the highway environment.

Figure 13 shows that the driver cut off the lane while overtaking and performed the maneuver in a situation where a collision was imminent. In scenario two, as shown in Figure 14, the speed of the car that was overtaken by the agent increased, resulting in a collision and a negative reward for the agent.



(a)

(b)

Figure 14. Collision in the overtaking maneuver in the highway environment ((**a**): preparing for overtaking; (**b**): collision).

Additionally, it could be beneficial to employ separate agents to learn specific scenarios. One agent could focus on the longitudinal movement of the car, while the other could specialize in the transverse movement. This division of training allows each agent to concentrate on specific tasks, thereby enhancing the overall training effectiveness.

5. Conclusions

This study presents a decision-making algorithm that is efficient and safe, relying on deep reinforcement learning (DRL), and is specifically designed for self-driving cars on highways. The algorithm used in this study is DDPG (deep deterministic policy gradient). The control framework developed can generalize across different driving scenarios, accounting for variations in the number of lanes and surrounding vehicles. Simulation results demonstrate that the proposed algorithm provides optimal performance and convergence rates. Moreover, our algorithm has a higher learning rate when compared to the DQN and PPO algorithms.

Our future work will involve exploring the application of our proposed decision-making system in real-time driving scenarios. We also plan to investigate the benefits of a connected environment, in which vehicles can share information. Additionally, we will utilize real-world driving data to evaluate the performance of our proposed decision-making method in actual driving environments. All of these efforts are aimed at enhancing the safety and efficiency of autonomous driving systems in real-world applications.

Author Contributions: Conceptualization, A.R. and S.A.; methodology, A.R. and S.A.; software, A.R. and S.A.; validation, A.R. and S.A.; formal analysis, A.R. and A.E.; investigation, A.R.; resources, A.R. and S.A.; data curation, S.A.; writing—original draft preparation, A.R.; writing—review and editing, A.E. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

Acknowledgments: We acknowledge the use of online translation tools to enhance the quality of our text.

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

References

1. Raj, A.; Kumar, J.A.; Bansal, P. A multicriteria decision making approach to study barriers to the adoption of autonomous vehicles. *Transp. Res. Part A Policy Pract.* **2020**, *133*, 122–137. [CrossRef]
2. Liu, T.; Tian, B.; Ai, Y.; Chen, L.; Liu, F.; Cao, D. Dynamic states prediction in autonomous vehicles: Comparison of three different methods. In Proceedings of the 2019 IEEE Intelligent Transportation Systems Conference (ITSC), Auckland, New Zealand, 27–30 October 2019; pp. 3750–3755.
3. Rasouli, A.; Tsotsos, J.K. Autonomous vehicles that interact with pedestrians: A survey of theory and practice. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 900–918. [CrossRef]
4. Gkartzonikas, C.; Gkritza, K. What have we learned? A review of stated preference and choice studies on autonomous vehicles. *Transp. Res. Part C Emerg. Technol.* **2019**, *98*, 323–337. [CrossRef]
5. Hoel, C.-J.; Driggs-Campbell, K.; Wolff, K.; Laine, L.; Kochenderfer, M.J. Combining planning and deep reinforcement learning in tactical decision making for autonomous driving. *IEEE Trans. Intell. Veh.* **2020**, *5*, 294–305. [CrossRef]
6. Hoel, C.-J.; Wolff, K.; Laine, L. Tactical decision-making in autonomous driving by reinforcement learning with uncertainty estimation. *arXiv* **2020**, arXiv:2004.10439. [CrossRef]
7. Song, W.; Xiong, G.; Chen, H. Intention-aware autonomous driving decision-making in an uncontrolled intersection. *Math. Probl. Eng.* **2016**, *2016*, 1025349. [CrossRef]
8. Furda, A.; Vlacic, L. Enabling safe autonomous driving in real-world city traffic using multiple criteria decision making. *IEEE Intell. Transp. Syst. Mag.* **2011**, *3*, 4–17. [CrossRef]
9. Nie, J.; Zhang, J.; Ding, W.; Wan, X.; Chen, X.; Ran, B. Decentralized cooperative lane-changing decision-making for connected autonomous vehicles. *IEEE Access* **2016**, *4*, 9413–9420. [CrossRef]
10. Li, L.; Ota, K.; Dong, M. Humanlike driving: Empirical decision-making system for autonomous vehicles. *IEEE Trans. Veh. Technol.* **2018**, *67*, 6814–6823. [CrossRef]
11. Yang, C.; You, S.; Wang, W.; Li, L.; Xiang, C. A Stochastic predictive energy management strategy for plug-in hybrid electric vehicles based on fast rolling optimization. *IEEE Trans. Ind. Electron.* **2020**, *67*, 9659–9670. [CrossRef]
12. Mnih, V.; Kavukcuoglu, K.; Silver, D.; Rusu, A.A.; Veness, J.; Bellemare, M.G.; Graves, A.; Riedmiller, M.; Fidjeland, A.K.; Ostrovski, G.; et al. Human-level control through deep reinforcement learning. *Nature* **2015**, *518*, 529–533. [CrossRef] [PubMed]
13. Duan, J.; Li, S.E.; Guan, Y.; Sun, Q.; Cheng, B. Hierarchical reinforcement learning for self-driving decision-making without reliance on labelled driving data. *IET Intell. Transp. Syst.* **2020**, *14*, 297–305. [CrossRef]
14. Kim, M.; Lee, S.; Lim, J.; Choi, J.; Kang, S.G. Unexpected collision avoidance driving strategy using deep reinforcement learning. *IEEE Access* **2020**, *8*, 17243–17252. [CrossRef]
15. Zhang, Q.; Lin, J.; Sha, Q.; He, B.; Li, G. Deep Interactive reinforcement learning for path following of autonomous underwater vehicle. *IEEE Access* **2020**, *8*, 24258–24268. [CrossRef]

16. Chen, C.; Jiang, J.; Lv, N.; Li, S. An intelligent path planning scheme of autonomous vehicles platoon using deep reinforcement learning on network edge. *IEEE Access* **2020**, *8*, 99059–99069. [CrossRef]
17. Yang, C.; Zha, M.; Wang, W.; Liu, K.; Xiang, C. Effcient energy management strategy for hybrid electric vehicles/plug-in hybrid electric vehicles: Review and recent advances under intelligent transportation system. *IET Intell. Transp. Syst.* **2020**, *14*, 702–711. [CrossRef]
18. Han, S.; Miao, F. Behavior planning for connected autonomous vehicles using feedback deep reinforcement learning. *arXiv* **2020**, arXiv:2003.04371. [CrossRef]
19. Nageshrao, S.; Tseng, H.E.; Filev, D. Autonomous highway driving using deep reinforcement learning. In Proceedings of the 2019 IEEE International Conference on Systems, Man and Cybernetics (SMC), Bari, Italy, 6–9 October 2019; pp. 2326–2331.
20. Shen, Y.; Peng, J.; Kang, C.; Zhang, S.; Yi, F.; Wan, J.; Wang, X. Collaborative optimisation of lane change decision and trajectory based on double-layer deep reinforcement learning. *Int. J. Veh. Des.* **2023**, *92*, 336–356. [CrossRef]
21. Lai, J.-P.; Li, H.; Shi, Y.; Xu, L.-M.; Yan, H. Anti Collision Control Strategy of Unmanned Vehicle Based on DDPG Algorithm. *Wuhan Ligong Daxue Xuebao J. Wuhan Univ. Technol.* **2021**, *43*, 68–76. [CrossRef]
22. Lv, K.; Pei, X.; Chen, C.; Xu, J. A Safe and Efficient Lane Change Decision-Making Strategy of Autonomous Driving Based on Deep Reinforcement Learning. *Mathematics* **2022**, *10*, 1551. [CrossRef]
23. Rais, M.S.; Boudour, R.; Zouaidia, K.; Bougueroua, L. Decision making for autonomous vehicles in highway scenarios using Harmonic SK Deep SARSA. *Appl. Intell.* **2022**, *53*, 2488–2505. [CrossRef]

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.