POLITECNICO DI TORINO
Repository ISTITUZIONALE

A Deep Reinforcement Learning Approach for Efficient, Safe and Comfortable Driving

Original
A Deep Reinforcement Learning Approach for Efficient, Safe and Comfortable Driving / Selvaraj, D.C., Hegde, S.S., Amati, N., Deflorio, F.P., Chiasserini, C.F.. - In: APPLIED SCIENCES. - ISSN 2076-3417. - STAMPA. - 13:9(2023). [10.3390/app13095272]

Availability:
This version is available at: 11583/2978096 since: 2023-05-12T06:53:55Z

Publisher:
MDPI

Published
DOI:10.3390/app13095272

Terms of use:

This article is made available under terms and conditions as specified in the corresponding bibliographic description in the repository

Publisher copyright

(Article begins on next page)

10 October 2026

applied sciences

MDPI

Article

A Deep Reinforcement Learning Approach for Efficient, Safe and Comfortable Driving †

Dinesh Cyril Selvaraj *, Shailesh Hegde, Nicola Amati, Francesco Deflorio and Carla Fabiana Chiasserini

CARS@Polito, Politecnico di Torino, 10129 Turin, Italy; shailesh.hegde@polito.it (S.H.); nicola.amati@polito.it (N.A.); francesco.deflorio@polito.it (F.D.); carla.chiasserini@polito.it (C.F.C.)
* Correspondence: dinesh.selvaraj@polito.it
† Presented at the 24th EURO Working Group on Transportation Meeting, Aveiro, Portugal, 8–10 September 2021.

Abstract: Sensing, computing, and communication advancements allow vehicles to generate and collect massive amounts of data on their state and surroundings. Such richness of information fosters data-driven decision-making model development that considers the vehicle's environmental context. We propose a data-centric application of Adaptive Cruise Control employing Deep Reinforcement Learning (DRL). Our DRL approach considers multiple objectives, including safety, passengers' comfort, and efficient road capacity usage. We compare the proposed framework's performance to traditional ACC approaches by incorporating such schemes into the CoMoVe framework, which realistically models communication, traffic, and vehicle dynamics. Our solution offers excellent performance concerning stability, comfort, and efficient traffic flow in diverse real-world driving conditions. Notably, our DRL scheme can meet the desired values of road usage efficiency most of the time during the lead vehicle's speed-variation phases, with less than 40% surpassing the desirable headway. In contrast, its alternatives increase headway during such transient phases, exceeding the desired range 85% of the time, thus degrading performance by over 300% and potentially contributing to traffic instability. Furthermore, our results emphasize the importance of vehicle connectivity in collecting more data to enhance the ACC's performance.

Keywords: connected vehicles; vehicle dynamics; adaptive cruise control; traffic simulation; machine learning

check for updates

Citation: Selvaraj, D.C.; Hegde, S.; Amati, N.; Deflorio, F.; Chiasserini, C.F. A Deep Reinforcement Learning Approach for Efficient, Safe and Comfortable Driving. *Appl. Sci.* **2023**, *13*, 5272. https://doi.org/10.3390/app13095272

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

Received: 24 March 2023
Revised: 15 April 2023
Accepted: 20 April 2023
Published: 23 April 2023

Copyright: © 2023 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (https://creativecommons.org/licenses/by/4.0/).

1. Introduction

According to a study conducted by the World Health Organization (WHO) [1], 1.35 million people annually succumb to road accidents and 20–50 million people experience non-fatal injuries. Road accidents can lead to traffic disruption and have a significant economic impact, with some countries losing up to 3% of their gross domestic product. In this context, the implementation of robust Advanced Driver Assistance Systems (ADAS) applications, such as Adaptive Cruise Control and Automatic Emergency Braking, can help drivers navigate safely, providing an automatic braking response to potential hazards, thereby enhancing safety and optimizing traffic efficiency. To envision such systems, vehicles and roadways are currently, and will progressively be, outfitted with sensory, processing and wireless communication capabilities. As a result, they are becoming data sources that generate massive amounts of information, advancing the growth of data-driven models for context-aware decision-making.

In this context, machine learning (ML) techniques play a crucial role, as they have proved to be very effective in prediction tasks and in supporting accurate decision-making. Importantly, given the ability of vehicles to exchange data with each other through vehicle-to-vehicle (V2V) communications as well as with the infrastructure through vehicle-to-infrastructure (V2I) communications, distributed ML approaches in vehicular networks are attracting a significant deal of interest.

Adaptive Cruise Control (ACC) is emerging as one of the most popular and relevant applications of ADAS that can benefit from ML techniques. Indeed, especially in adverse road conditions, standard ACC increases the mental and temporal workload of the drivers [2], often leading to human errors. On the other hand, connected autonomous vehicles are required to implement sophisticated control of the vehicle's behavior in terms of efficiency, safety, and comfort under all road conditions. The headway control between two consecutive vehicles can increase road capacity and stabilize traffic flow [3–5], thereby improving traffic efficiency. The safety term addresses issues pertaining to vehicle stability and the Time-To-Collision between two vehicles. The comfort term regulates the time derivative of the vehicle's acceleration to provide passengers with a jerk-free travel experience.

In particular, we are focusing on improving the safety of vehicles by accounting for vehicle stability in different road conditions, an aspect that has been overlooked in the literature. Generally, the literature on ACC [6,7] assesses the safety of a vehicle only based on the inter-vehicular distance between the ego vehicle and the lead vehicle. However, road conditions, such as wet or icy surfaces, can impact vehicle stability, affecting passenger safety. Therefore, in our study, we have incorporated longitudinal slip as a vehicle stability indicator to ensure the safety of the passengers, a factor other related works have failed to tackle. The proposed methodology relies on the longitudinal wheel slip and tire–road friction coefficient to prevent loss of traction with the road surface under diverse conditions, effectively addressing the drawback. However, the accurate representation of the vehicle states is difficult in real-world conditions, for instance, GNSS positioning errors. Even though we assumed the availability of such information from simulation models, several research works [8–10] have been proposed to estimate them in real-life situations.

In this work, we address the challenges posed by the ACC in connected autonomous vehicles by utilizing Reinforcement Learning (RL), a particular ML technique, to address the main problems with the standard ACC system. The choice of RL is due to its ability to cope with highly dynamic environments and non-linear systems for near-optimal decision-making and control. In RL, an agent learns to map environmental states onto actions and subsequently receives a numerical reward for the adopted action. It aims to learn the optimal state–action mapping policy yielding maximum cumulative numerical reward.

With the aim of dealing with real-world operational conditions, we focus on Deep Reinforcement Learning (DRL), which incorporates deep neural networks in the RL approach to effectively represent the optimal policy mapping despite the high dimensionality of the states usually observed in real-world situations. In particular, we use the Deep Deterministic Policy Gradient (DDPG) [11] technique, which is a model-free off-policy DRL algorithm, and we develop a DRL-based ACC application (depicted in Figure 1) that can quickly adapt to the surroundings to ensure safe, comfortable, and efficient driving.

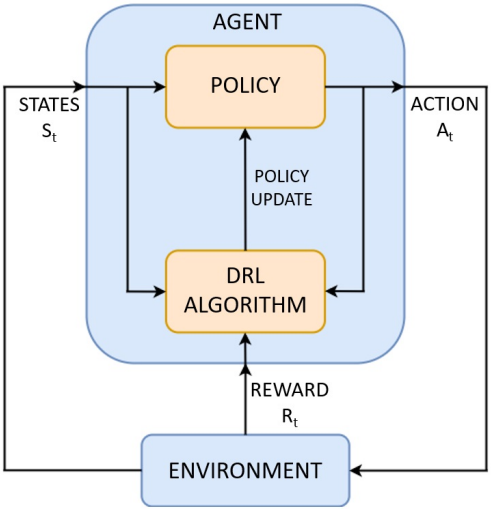

Figure 1. Architecture of the proposed DRL framework.

The key contributions of this work are summarized as follows:

(i) We design a DRL framework that takes into account and appropriately weights the various environmental factors influencing ACC, including vehicle stability. To the best of our knowledge, we are the first to comprehensively and successfully address all relevant issues, as existing studies focusing on ACC have not focused on such a crucial issue as vehicle stability.
(ii) We assess the performance of our DRL framework by incorporating it into the CoMoVe framework [12], which offers a realistic representation of traffic mobility, vehicle communication, and dynamics. By utilizing such a fully fledged simulation tool, we derive performance results regarding vehicle stability, comfort, and traffic flow efficiency under diverse traffic conditions and road circumstances.
(iii) We compare the DRL framework results against traditional ACC and cooperative ACC (CACC) algorithms and demonstrate the benefits of utilizing the information obtained through V2X communications in the learning process of the DRL agent, especially concerning the algorithm convergence time.

The remainder of this paper is organized as follows. Section 2 discusses the related literature works and emphasizes our contributions. Section 3 introduces some preliminaries on RL and presents the proposed DRL model and its integration with the CoMoVe framework. Section 4 presents the simulation scenarios and the DRL agent's performance, while Section 5 concludes the paper and outlines future research directions.

2. Related Work

ML techniques have been widely used in autonomous and automated vehicles to address a wide range of applications requiring a decision-making process [13–15]. Below, we discuss the works that are most relevant to ours, as they have focused on ACC and addressed the drawbacks that affect traditional ACC approaches.

In this context, among the various possible ML approaches, (D)RL has been often applied, because of its effectiveness in dealing with control problems in dynamic and partially observable environments. Further, the survey in [15] validates the effectiveness of (deep) reinforcement learning models in vehicle longitudinal control systems. (D)RL models provide an edge over their counterparts, which suffer from the lower variety of scenarios recorded in complex road environments. The reward function of a (D)RL framework is a critical component in the DRL agent's learning process, as it allows the agent to assess its performance in different states and learn the optimal strategy. In particular, ref. [16] shows that the usage of sparse rewards can result in instability and policy convergence to non-optimal solutions. We remark that, in contrast to earlier work [17], we have defined a continuous reward function in our model in order to address the aforementioned issues and successfully determine the ideal trade-off between efficiency, safety, and driving comfort.

The study in [7] presents one of the earliest research works leveraging DRL for CACC. The framework proposed therein leverages the information acquired from the RADAR sensor and through V2V communications to preserve the time headway with the leading vehicle. The solution, however, limits the action space with discrete values where the vehicle can accelerate/decelerate only with predetermined values, which may result in oscillating behavior of the vehicle control and may not accurately represent the vehicle's response in real-world conditions. Further, ref. [7] considers the headway as the only safety objective of the DRL, thus neglecting comfort and stability. This is a also major drawback of traditional ACC solutions, which do not account for road conditions that can lead to a dangerous situation while controlling the vehicle acceleration.

These issues are partially addressed by the DRL-based framework proposed in [6], which deals with a continuous action space and reward function, accounting for multiple criteria concerning time-to-collision, headway, and jerk. The major shortcoming of the framework in [6] resides in the representation of the vehicle dynamics or the lack thereof. Without considering vehicle dynamics, the framework's vehicle response is inconsistent with the vehicle's actual performance. In addition, the representation of the RADAR

sensor outputs is generated using the vehicle positions, as opposed to simulating a realistic RADAR sensor. Other works proposing multi-objective functions to deal with safety and comfort can be found in [18–20]. Even though vehicle stability under different road conditions is an essential factor in passenger safety, none of them consider it an objective.

In the context of comfort, ref. [21] presents a comprehensive empirical analysis of the factors affecting the comfort of the passenger when designing an automated driving vehicle. In our study, we evaluate the influence of situational driving conditions, such as potential collisions, on passenger comfort. A work that on the contrary focuses solely on stability is the scheme introduced in [22]. The solution therein controls the torque vectoring of an electric vehicle to improve vehicle stability. Ref. [22] use a discrete action space to determine the optimal torque vectoring ratio by exploiting yaw rate and steering angle as states.

Finally, ref. [23] presents a performance comparison between the Model Predictive Control (MPC) and DRL for ACC. This study highlights that, being a model-based system, the MPC requires online optimization, hence exacting a high toll in terms of computing resource consumption for real-time applications. DRL, on the other hand, can be model-free and produce results in a timely and efficient manner. In addition, the DRL framework inherently captures the surrounding environment, contributing to improved performance under control delays and sensor measurement errors. As a drawback, DRL suffers from generalization issues. We emphasize that, based on our preliminary findings, DRL is capable of overcoming such issues with sufficient training in various scenarios. We have indeed tested the ACC performance of the DRL approach under different environmental conditions and scenarios and always found that it could cope well with any of these different situations.

Usually, (D)RL frameworks exploit well-known simulators as training environments and validation tools to evaluate their agent's performance; an example is the study in [17], which utilizes VISSIM, a commercial traffic simulator. Within the scope of our study, we utilize the capabilities of the CoMoVe simulation scheme [12], both as an environment and a validation instrument for (D)RL agents. Compared to other virtual validation tools [24,25], the CoMoVe simulation framework accounts for detailed vehicle dynamics and communication models to ensure realistic simulations.

Novelty: To the best of our knowledge, we are the first to account for all relevant factors in ACC, including the various environmental conditions that may be present in real-world scenarios. In particular, unlike previous work, we designed our DRL framework to use a continuous action space and consider time headway as well as the comfort and safety of the driver and passengers, including vehicle stability. Regarding the latter, the longitudinal slip of the vehicle has been one of our objectives to ensure vehicle stability on various types of road surfaces. We have employed the CoMoVe framework [12] to evaluate the performance of our DRL, which provides a realistic representation of vehicle dynamics, communication, and traffic mobility. Further, it uses a RADAR and Vision-based sensor array to represent the real-world sensor system.

3. Design and Implementation of the DRL Framework

In this section, we first introduce the DRL model we developed (Section 3.1); then, we describe how the DRL model is integrated into our CoMoVe framework for its assessment in real-world scenarios (Section 3.2).

3.1. The DRL Model

We now comprehensively present the DRL model, beginning with an overview of DRL and then presenting the solution we created.

3.1.1. Preliminaries

In general, Reinforcement Learning (RL) is a sequential decision-making problem in which an agent observes certain states ($s(t)$) as a representation of the environment

and chooses an action ($a(t)$) based on the policy (π) at a given time step t. Given the action, the environment will transition to a new state (s') and receive a numerical value ($r(s(t), a(t))$) as a reward. In RL, the interaction between the agent and the environment is typically formulated as a discounted Markov Decision Process (MDP). The discounted MDP can be formally defined as a tuple $(\mathcal{S}, \mathcal{A}, \mathcal{P}, \mathcal{R}, \gamma)$ where $\mathcal{S}$ is the state space, $\mathcal{A}$ is the action space, $\mathcal{P}$ is the state transition probability matrix ($\mathcal{P}^a_{ss'} = \mathbb{P}[s(t+1) = s' | s(t) = s, a(t) = a]$) conditioned to the action taken by policy π, $\mathcal{R}$ is the reward function ($r(s(t), a(t)) = \mathbb{E}[r(s(t), a(t)) | s(t) = s, a(t) = a]$) and $\gamma \in [0, 1]$ is a discount factor that defines the importance of future rewards relative to the instant rewards. Policy π defines the decision-making strategy through which an RL agent chooses an action based on the previous set of observations. The RL agent aims to choose actions that maximize the expected discounted reward $\mathbb{E}[\sum_{k=0}^{T} \gamma^k r(s(t+k), a(t+k))]$ where γ is the discount rate and $r(s(t+k), a(t+k))$ is the immediate reward received at time step $t+k$.

An RL agent can utilize either model-based or model-free techniques to find the optimal solution to a given problem. Model-based approaches properly model the environment's transitions, where Dynamic Programming (DP)-based algorithms can identify the optimal solution. We resort to model-free solutions because, in most situations, the state transitions of the environment are challenging to model due to the intrinsic nature of the transitions. Such methods do not require a comprehensive understanding of transitions; instead, they rely on experience samples from the environment where Monte-Carlo Temporal-Difference (TD) Learning-based methods are applied. Q-learning [26] and SARSA [27] are the two most popular TD model-free algorithms used to solve RL problems. However, tabular approaches such as Q-learning become increasingly difficult to implement when there are a vast number of states and actions, as the $Q(s, a)$ values must be stored for all state–action pairs. Therefore, with deep neural networks as function approximators, $Q(s, a|\theta)$ can be used to represent the state and action value pairs, with θ signifying the neural network's weights. The integration of deep neural networks and reinforcement learning techniques in Deep Reinforcement Learning facilitates the training of a DRL agent to reach the ideal solution for complex real-world problems. In this study, the Deep Deterministic Policy Gradient (DDPG) [11], a model-free Deep Reinforcement Learning algorithm, is used to learn the optimal policy for the DRL-based ACC application. It is worth noting that the DDPG algorithm is equipped to handle control action space in a continuous domain and uses an actor-critic framework to learn a deterministic policy.

3.1.2. DRL-Based Acc Application

Figure 1 presents the high-level view of the framework we propose. The framework is considered to be deployed in the ego vehicle, where data about the system state are collected from local sensors as well as from neighboring vehicles, and a decision on the desired acceleration is made.

State Space: The state space at any given time slot t comprises: (i) the lead vehicle acceleration $\alpha(t)$, (ii) the headway $\vartheta(t)$, (iii) the headway derivative $\Delta\vartheta(t)$, (iv) the longitudinal slip $\xi(t)$, and (v) the friction coefficient $\mu(t)$. In the state space, V2X communication is used to acquire the lead vehicle acceleration ($\alpha(t)$), and it is assumed that the ego vehicle is equipped with an estimation mechanism to ascertain the tire–road friction coefficient ($\mu(t)$). The rest of the state space parameters are expressed in the following equations:

$$\vartheta(t) = \frac{\Delta P_{lead}(t)}{\mathcal{V}_{ego}(t)} \tag{1}$$

$$\Delta\vartheta(t) = \vartheta(t) - \vartheta(t-1) \tag{2}$$

$$\xi(t) = \begin{cases} \frac{\mathcal{V}^R_{ego}(t) - \mathcal{V}^W_{ego}(t)}{\mathcal{V}^R_{ego}(t)}, & \ddot{x}(t) >= 0 \\ \frac{\mathcal{V}^R_{ego}(t) - \mathcal{V}^W_{ego}(t)}{\mathcal{V}^W_{ego}(t)}, & \text{otherwise} \end{cases} \tag{3}$$

where $\Delta P_{lead}(t)$ represents the inter-vehicular distance between two car following vehicles, $\mathcal{V}_{ego}(t)$ indicates the ego vehicle's velocity, $\vartheta(t-1)$ and $\vartheta(t)$ are the headway values at time $t-1$ and t (respectively), $\mathcal{V}^{R}_{ego}(t)$ and $\mathcal{V}^{W}_{ego}(t)$ are the ego vehicle's equivalent rotational velocity and longitudinal axle velocity (respectively) [28], and $\ddot{x}(t)$ is the ego vehicle's acceleration at time t. Compared to [6,7], our model takes into account not just the wheel longitudinal slip ratio but also the tire–road friction coefficient accounting for the vehicle stability.

Action Space: The action space $a(t) \in \mathcal{A}$, which denotes the acceleration to be adopted by the ego vehicle, encompasses values within the range $[-2, 1.47]$ to ensure a comfortable ride for the driver and passengers [29]. The sampling period (τ) of our framework is set at 100 ms, with state observation and action decisions taken at each such interval.

Reward Components: We express the reward, i.e., the numerical value received by the DRL Agent from the environment as a direct response to the DRL Agent's action, as a multi-objective function. Specifically, it includes three components: Headway (signifying traffic flow efficiency), Comfort (for ride quality), and Stability (denoting safety), each component taking values in the $[-1,1]$ range. More formally, given state $s(t)$ and action $a(t)$, the reward is given by

$$r(s(t), a(t)) = x_{hw} \cdot r_{hw}(s(t), a(t)) + x_{cmf} \cdot r_c(s(t), a(t)) + x_{stb} \cdot r_s(s(t), a(t)) \quad (4)$$

where x_{hw}, x_{cmf}, and x_{stb} are weighting coefficients that, as detailed later, are set dynamically over time, and $r_{hw}(s(t), a(t))$, $r_c(s(t), a(t))$, and $r_s(s(t), a(t))$ are, respectively, the headway, comfort, and vehicle stability reward component at time step t. The reward components are described in detail below.

Headway reward component: Time headway can be used as an alternative way to calculate the distance between the lead and ego vehicle, and it is expressed as in Equation (1). Although the term gap is widely used to define this time interval between consecutive vehicles, in this study, we will use the term "headway", according to the recent related papers on ACC [30,31]. The desired time headway between two vehicles is set at 1.3 s, and a headway value under 0.5 s indicates a high risk of collision between the two vehicles [7]. However, the headway value becomes ∞ when the ego vehicle approaches a stand-still situation, i.e., zero velocity. This causes undesirable effects in DRL states, rewards, and the agent's learning progress. To overcome the issue, we have saturated the ego vehicle velocity to 2.16 m/s with the secure stand-still distance between two successive vehicles as 2.81 m [32], and the optimal headway of 1.3 s, as reported in Equation (5).

$$\mathcal{V}_{ego} = \begin{cases} \mathcal{V}_{ego}, \mathcal{V}_{ego} > 2.16 \\ \mathcal{V}^{sat}_{ego}, \mathcal{V}_{ego} \leq 2.16 \\ \mathcal{V}^{sat}_{ego} = \frac{2.81\,m}{1.3\,s} \Rightarrow 2.16\,m/s \end{cases} \quad (5)$$

A Log-Normal distribution function is used to model the headway reward component ($r_{hw}(s(t), a(t))$), which provides a maximum reward of +1 for a headway of 1.3 s and a minimum of -1 for 0.5 s. It is formulated as

$$r_{hw}(s(t), a(t)) = 2 \cdot (F_{hw} - 0.5), \quad \text{with} \quad (6)$$

$$F_{hw} = 0.4944 \cdot f_{lognorm}(\vartheta(t) | \epsilon = 0.285, \sigma = 0.15), \quad (7)$$

$$f_{lognorm}(x | \epsilon, \sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp \frac{(-(lnx - \epsilon)^2)}{2\sigma^2} \quad (8)$$

Comfort reward component: Jerk $j(t)$, i.e., the time derivative of the vehicle's acceleration, measures the comfort of the driver and passengers. In this work, the comfort reward component is designed using the absolute values of the jerk with values less than 0.9 m/s^3 corresponding to the best comfort, while values above 1.3 m/s^3 fall under aggressive driving behavior [29]. When $j(t)$ increases from 0.6 m/s^3 to 2 m/s^3, the comfort reward value gradually drops until it reaches -1, the lower reward limit. The desired reward trend for jerk behavior has been formulated through the polynomial curve fitting method. However, in critical situations, the safety of passengers is of more importance than comfort. Thus, we consider the Time-to-Collision (TTC), which indicates the time until the potential

collision occurs, as a safety indicator and discount comfort, instead giving priority to safety in dangerous situations. According to European regulations, TTC ≤ 4 s indicates a critical situation. Thus, we write:

$$r_c(s(t), a(t)) = w_{cmf} \cdot f(jerk), \quad \text{with} \tag{9}$$

$$f(jerk) = polyfit(j(t), 9), \qquad w_{cmf} = \begin{cases} 1, TTC(t) > 4 \\ 0, TTC(t) \leq 4\,. \end{cases} \tag{10}$$

where w_{cmf} is the weight indicator to disregard comfort in critical situations.

Stability reward component: The longitudinal stability is valued in terms of the maximum tractive effort of a tire on a road contact area. According to the experimental data, a longitudinal slip lower than 0.2 is regarded as a stable condition [33]. In this case, a tanh function is used to represent the stability reward component. We write:

$$r_s(s(t), a(t)) = 2.0099 \cdot (F_{stb} + 1) \quad \text{with} \quad F_{stb} = \tanh(-3 \cdot \xi(t))\,. \tag{11}$$

Dynamic weight coefficients: The dynamic coefficients x_{hw}, x_{stb}, and x_{cmf} play an important role in weighing each reward component; their values depend on the operating region, with the ideal regions being defined as $1.25 \leq \vartheta(t) \leq 1.35$ for headway, $0 \leq |\xi(t)| \leq 0.2$ for stability, and $0 \leq |j(t)| \leq 0.9$ for comfort. To amplify the negative rewards, those reward components taking values outside the ideal region are given more importance than others. However, since the coefficients must always sum to 1, the maximum (minimum) total reward a DRL agent can receive in time step t is still $+1$ (-1). In particular, we set the dynamic coefficients in such a way that, if all reward components are outside or inside the ideal regions, then each of them will obtain $1/3$ as their coefficient. In other cases, with one of the reward components (e.g., comfort) being outside the ideal region, the dynamic coefficient values will be $2/3$ for the comfort reward component and $1/6$ for each of the other components. Therefore, the dynamic coefficient values will sum to $+1$ for all combinations, and the resulting reward will be between $[-1, +1]$.

3.2. Integrating the Drl Model in the Comove Framework

A realistic simulation environment is vital for a DRL agent to learn the desired behavior, which includes the movements of surrounding vehicles and environmental circumstances. We realize such a simulation environment through CoMoVe, our sophisticated simulation framework. The CoMoVe framework [12], shown in Figure 2, integrates widely known simulators from Mobility, Communication, and Vehicle Dynamics domains. It combines (i) the ns-3 simulator, with the LENA module, simulating LTE-based V2X communications; (ii) the SUMO simulator for vehicle mobility; (iii) the MathWorks module, which models the vehicle on-board sensors and vehicle dynamics; and (iv) the Python engine, which serves as an interface to facilitate the exchange of data between the modules, and it also features a DRL agent to control the vehicle's movements.

The Python engine used a combination of Python libraries to envision efficient interactions with other simulators. It includes TraCI, MATLAB Engine, and ns3 Python bindings to interact with SUMO, extract vehicle-related information, and simulate vehicular communication, respectively. Concerning the DRL state components, the ns3 V2X communication model facilitates the reception of lead vehicle acceleration value ($\alpha(t)$), the headway ($\vartheta(t)$) and headway derivative ($\Delta\vartheta(t)$) values are computed using the velocity and distance measurements acquired from the vehicle sensor models, and the Simulink Vehicle Dynamic model provides the longitudinal slip ($\xi(t)$) and friction coefficient ($\mu(t)$) values. The Python Engine gathers these values and provides them to the DRL framework as observed state components. The Vehicle Dynamics Model utilizes the action (desired acceleration) of the DRL framework as a reference signal for the lower-level controller of the ego vehicle. The vehicle is modeled using Simulink with fourteen Degree-of-Freedom. The key aspects of the vehicle model are an eight-speed automatic transmission, rear-wheel driven, and a Spark Ignition engine.

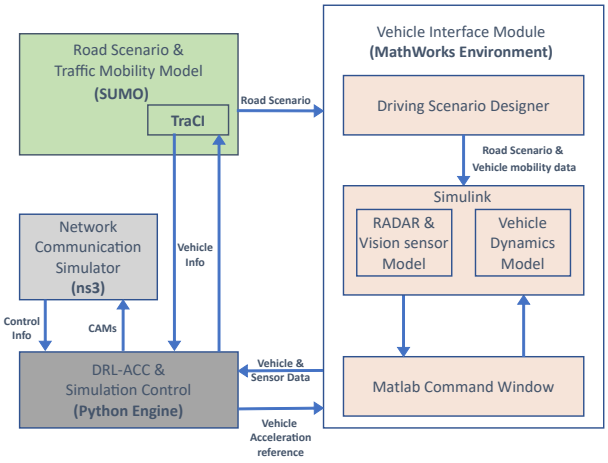

Figure 2. CoMoVe framework architecture.

4. Performance Results

Using the CoMoVe framework, we now demonstrate how our DRL-based ACC application improves driving safety, comfort, and efficiency under various road conditions, and compare its performance to that of traditional proportional controllers for ACC and CACC as in [12]. In addition to the proportional controller of ACC, CACC includes a sum of the lead vehicle's acceleration as a feedforward signal to make the vehicle more reactive.

4.1. Reference Scenarios

We draw the reference scenarios based on the work of [18,20], which we have enhanced by introducing the road friction coefficient, as it has a significant impact on vehicle stability. We have divided our scenarios into two distinct traffic conditions: highway and urban. In highway-related scenarios, the ego vehicle follows the lead vehicle on a straight road under two conditions: (1) slippery road, with a road friction coefficient of 0.35, where the lead vehicle accelerates/decelerates smoothly, and (2) sharp lead vehicle deceleration, with a road friction coefficient of 0.55 and the lead vehicle decelerating by nearly $-7\,\mathrm{m/s^2}$. The low road friction coefficient is intended to affect only the car's left front and rear wheels, simulating conditions such as a wet puddle or oil leak along the roadside. Figure 3 depicts the highway scenarios and test cases.

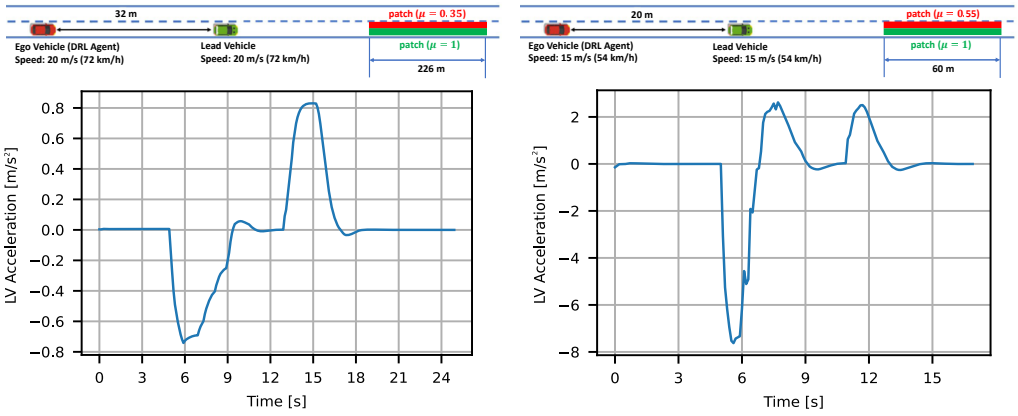

Figure 3. Highway scenarios: Road condition (**top**) and lead vehicle acceleration (**bottom**) in the slippery road scenario (**left**), and the sharp lead vehicle deceleration scenario (**right**).

In the urban environment, traffic queuing is a frequent situation that happens in day-to-day life under different circumstances, such as crossing an intersection or sluggish traffic movements. It is also an intriguing scenario to analyze, given its direct impact on efficient traffic flow, and is undoubtedly a challenging scenario for traditional ADAS

controllers. In our case, we simulate a traffic queuing scenario in which a lead vehicle approaches a slow-moving traffic situation. The profile of the lead vehicle is designed to drive at a rate consistent with the traffic flow; specifically, it decelerates to a lower speed of about 1 m/s, and later it gradually accelerates to match the traffic movements. Figure 4 illustrates the traffic queuing scenario and acceleration trend of the lead vehicle. We adopted the same hyperparameter values as in our prior paper [34] for the DDPG training process, with changes to the Replay buffer and mini-batch size, which were set to 25,000 and 48 (slippery road scenario)/64 (Sharp lead vehicle deceleration and Traffic queuing scenario,) respectively.

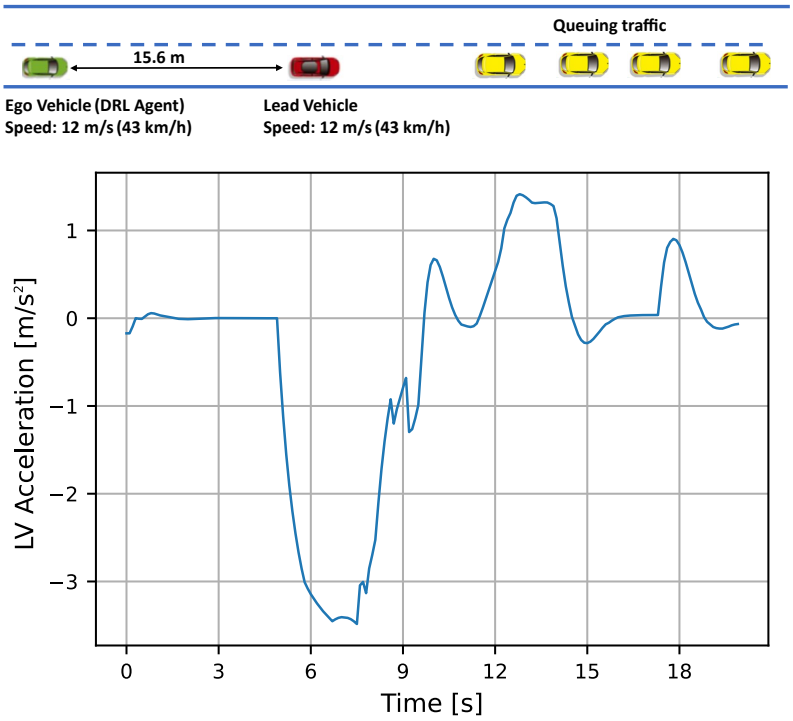

Figure 4. Urban Scenario: Road network (**top**) and lead vehicle acceleration (**bottom**) in the traffic queuing scenario.

4.2. Results

We first assess in Figure 5 the system performance under the slippery road scenario. Figure 5 shows the velocity (top left) and acceleration (top right) trend of the ego vehicle and lead vehicle. The corresponding ego vehicle's headway (left) and jerk (right) trend is shown in the bottom plots of Figure 5, concerning the adopted DRL, CACC, and ACC algorithms. The plot also highlights the desired range within which the headway should remain (black lines), for the ego vehicle to follow the lead vehicle and improve road usage efficiency. It is evident that the DRL approach helps the ego vehicle to maintain the headway in the desirable range for a longer period of time compared to its alternatives (97% as opposed to 32%). Furthermore, it is essential to maintain the desired headway during the acceleration and deceleration phases of the lead vehicle to prevent a potential phantom traffic jam situation. Under such conditions, our DRL framework can preserve the headway in the preferred range at all times, while the existing alternatives do so only 30% of the time. In particular, the headway for the traditional ACC and CACC algorithms lies outside the optimal range over 60% of the time. It must be highlighted that the time-to-collision (TTC), a safety indicator, remains above the critical four-second threshold, regardless of whether the headway is within the desired range or not.

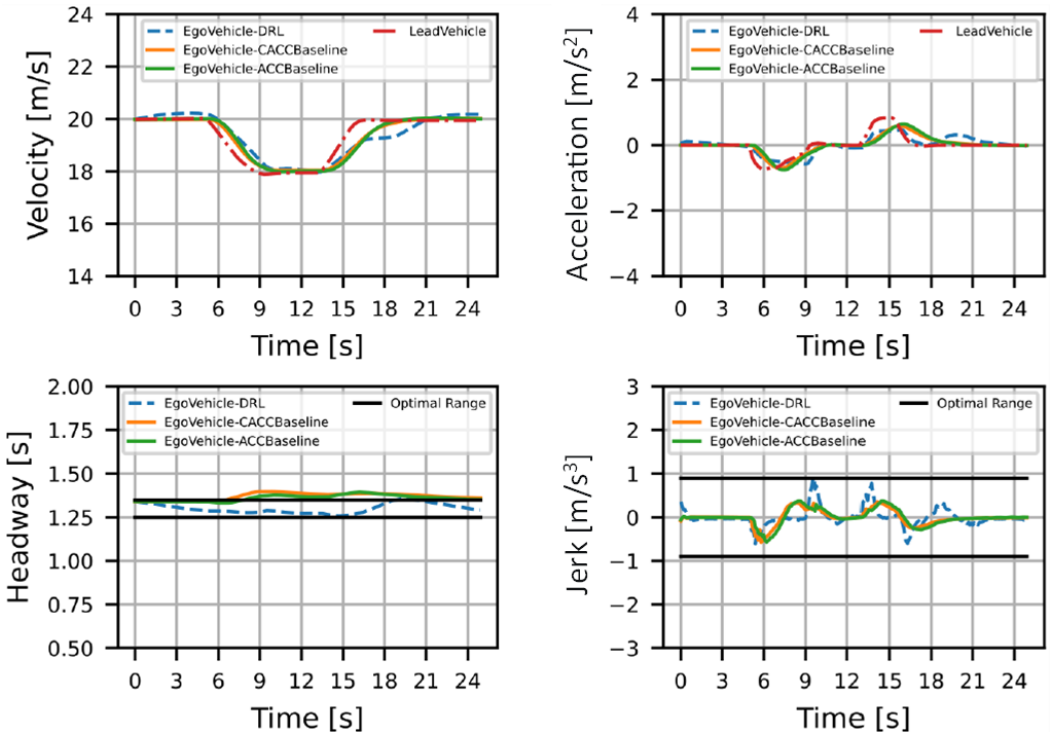

Figure 5. Slippery road scenario. Top: velocity (**left**) and acceleration (**right**). Bottom: headway (**left**) and jerk (**right**), under DRL, ACC, and CACC.

The driving comfort is depicted in the right bottom plot of Figure 5. In the slippery road scenario, all algorithms consistently produce jerk metric values that fall within the desirable range, resulting in a pleasant driving experience (the latter being highlighted with black lines). To further investigate the vehicle's stability, Figure 6 depicts the trend of the longitudinal wheel slip. As shown in Figure 6, the longitudinal slip remains within the safe operating range ($< \pm 0.2$), guaranteeing the vehicle's stability. As one can see by looking at the plots, the better performance of the proposed DRL solution in terms of road usage efficiency does not come at the cost of any degradation in terms of the vehicle's stability.

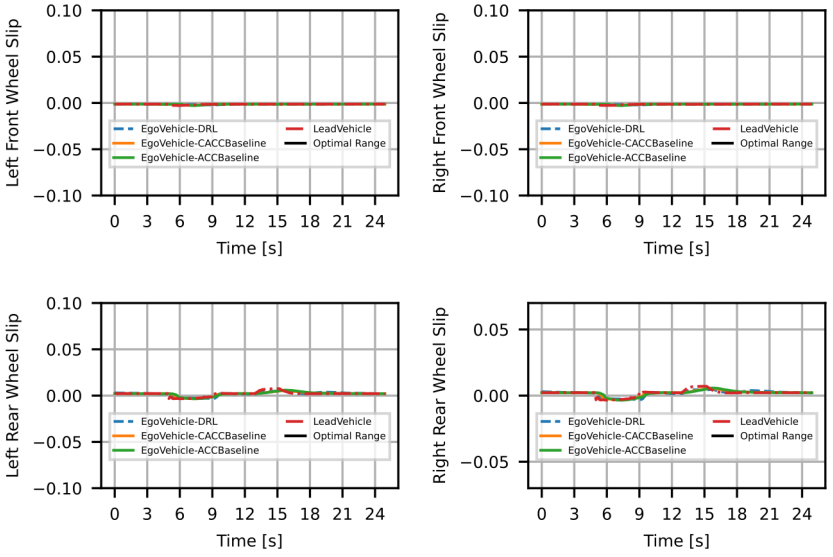

Figure 6. Slippery road scenario: Wheel longitudinal slip with respect to DRL, ACC, and CACC. The top (bottom) left and right plots represent the front (rear) left and right wheels (respectively).

Next, we focus on the sharp lead vehicle deceleration scenario. As we can notice from the lead vehicle's acceleration profile in Figure 3, this scenario reflects an emergency situation where the lead vehicle has to brake heavily under an unfavorable road or traffic condition. Figure 7 reveals that the lead vehicle velocity (top left) dropped from 15 m/s to 7 m/s in a span of 1.5 s. As a consequence, the ego vehicle has to brake suddenly and heavily, to avoid a collision. The performance of all three algorithms under this critical scenario are presented in the bottom plots of Figure 7 as well as in Figure 8. The headway performance (bottom left plot of Figure 7) reveals that the proposed DRL framework provides a dramatic improvement with respect to the traditional ACC and CACC algorithms.

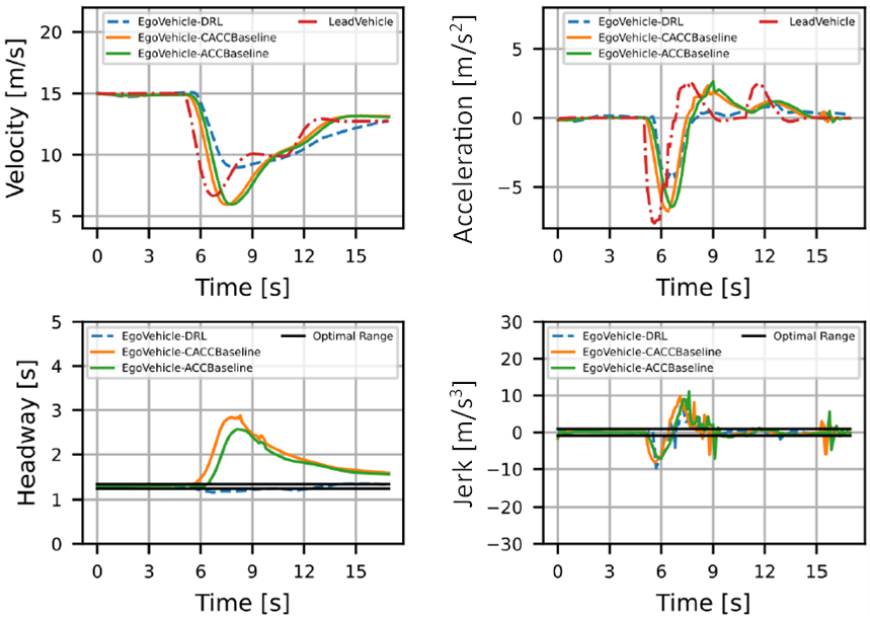

Figure 7. Sharp lead vehicle deceleration scenario. Top: velocity (**left**) and acceleration (**right**). Bottom: headway (**left**) and jerk (**right**) under DRL, ACC, and CACC.

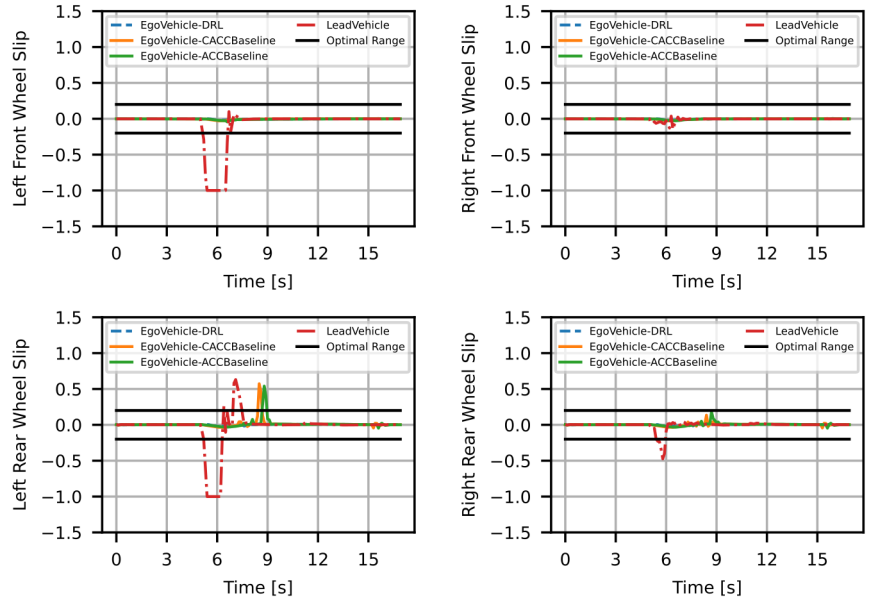

Figure 8. Sharp lead vehicle deceleration scenario: Wheel longitudinal slip with respect to DRL, ACC, and CACC. The top (bottom) left and right plots represent the front (rear) left and right wheels (respectively).

The simulation results show that DRL can maintain the preferred headway for 55% of the time, which represents a 19% improvement with respect to traditional algorithms. Similarly, DRL outperforms the (C)ACC algorithms during the transient phase, keeping the headway in its desired range 28% of the time—a considerable improvement over the 13% achieved by traditional algorithms. Furthermore, the DRL's headway trend persists around the desired region for a longer period of time than its alternatives. In terms of comfort (bottom right plot of Figure 7), the jerk values occasionally exceed the preferred region for all algorithms, whereas the DRL approach can significantly reduce the intensity and duration of jerk spikes. We observe that, in such a critical scenario, the TTC briefly falls below 4 s (between 5.9 s and 7.0 s) when the velocity of the lead vehicle decreases abruptly for the DRL-based scheme; however, the fact that the headway remains above 1 s indicates that the situation is not particularly dire. In addition, the DRL model is intended to prioritize safety over comfort, as the jerk is only considered for TTCs greater than 4 s (as demonstrated by Equation (10)).

In terms of vehicle stability, Figure 8 shows that traditional ACC-based approaches have longitudinal slip values outside the desired range, resulting in the ego vehicle's poor performance and compromising the driver's and passengers' safety. On the contrary, our DRL-based solution achieves the desired performance level, demonstrating that it can provide an excellent trade-off between the objectives and achieve better road usage efficiency and comfort.

In the traffic queuing scenario, as shown by the lead vehicle acceleration trend in Figure 4, the lead vehicle slows down at a deceleration rate of $3\,m/s^2$ in order to align with the velocity of the urban traffic. This situation requires a similar response from the ego vehicle to avert a potential collision and keeps the traffic flow smooth. Therefore, the DRL agent that runs on the following vehicle (i.e., the ego vehicle) has to learn the behavior of the lead vehicle to avoid the collision while sustaining a good and nearly constant headway to ensure traffic efficiency. Since we are dealing with a low-velocity situation, we calculate the headway based on the saturated ego vehicle velocity, as reported in Equation (5).

The top plots of Figure 9 depict the acceleration and velocity trend: the lead vehicle's velocity profile demonstrates a rapid deceleration from 12 m/s to 1 m/s in just 5 s, with subsequent changes in velocity to keep up with the traffic. The headway plot (bottom left plot of Figure 9) reveals that DRL is able to sustain the headway near the desired region more often than the traditional algorithms: the conventional approaches indeed leave a wider space between the vehicles, resulting in a decrease in traffic flow efficiency at lower speeds. In terms of traffic flow efficiency, the DRL framework outperforms other approaches, as it can sustain the preferred headway for a longer amount of time, while (C)ACC algorithms can only sustain it briefly when the lead vehicle is cruising at a steady speed. Additionally, during the transient phase, the DRL can sustain the desirable headway 50% of the time, which is substantially better than the alternatives' 4%. The Root Mean Square Error (RMSE) comparison, which is presented in Table 1, further highlights the (C)ACC algorithm's significant deviation from the desired region.

In terms of comfort (bottom right plot of Figure 9), jerk values are out of the ideal range occasionally for all algorithms. However, given the gravity of the situation, achieving the desired headway and ensuring safety are critical, surpassing the need for passengers' comfort. In the DRL framework, the maximum value of jerk is observed as the vehicle approaches zero velocity. This is due to the usage of the saturated ego vehicle velocity in the headway calculation, as stated in Equation (5), resulting in a lesser headway, which activates the heavy braking action. Using the saturated velocity, however, is a crucial step in controlling the vehicle at lower speeds. In terms of TTC, DRL provides a lower TTC ($\leq$4 s) between the time frame of the 7th to 11th second; however, given the observed speed and headway, we can conclude that the ego vehicle is not on an imminent collision course with the lead vehicle. Furthermore, since this scenario involves dry road conditions, Figure 10 shows that the vehicle proves to remain stable under any of the considered schemes.

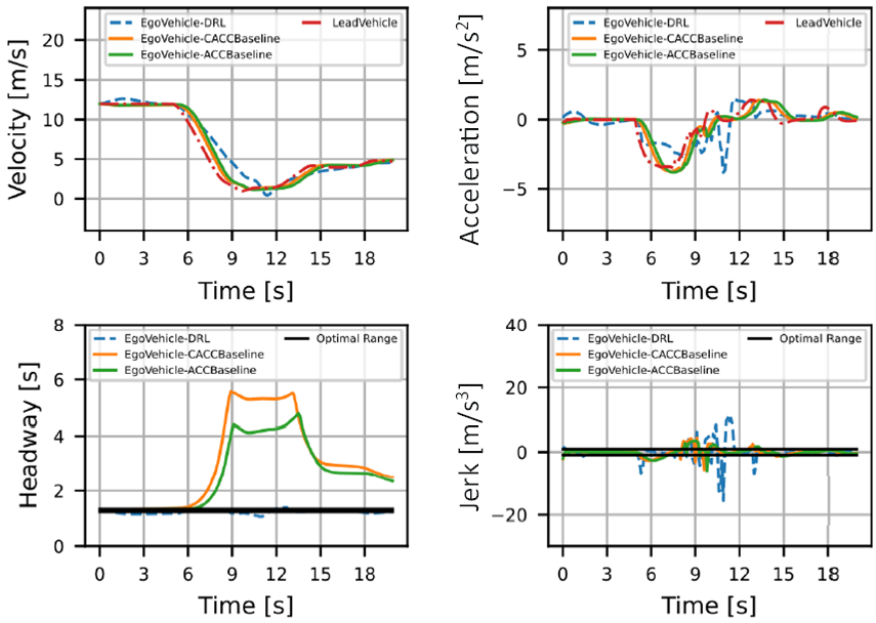

Figure 9. Traffic queuing scenario. Top: velocity (**left**) and acceleration (**right**). Bottom: headway (**left**) and jerk (**right**), under DRL, ACC, and CACC.

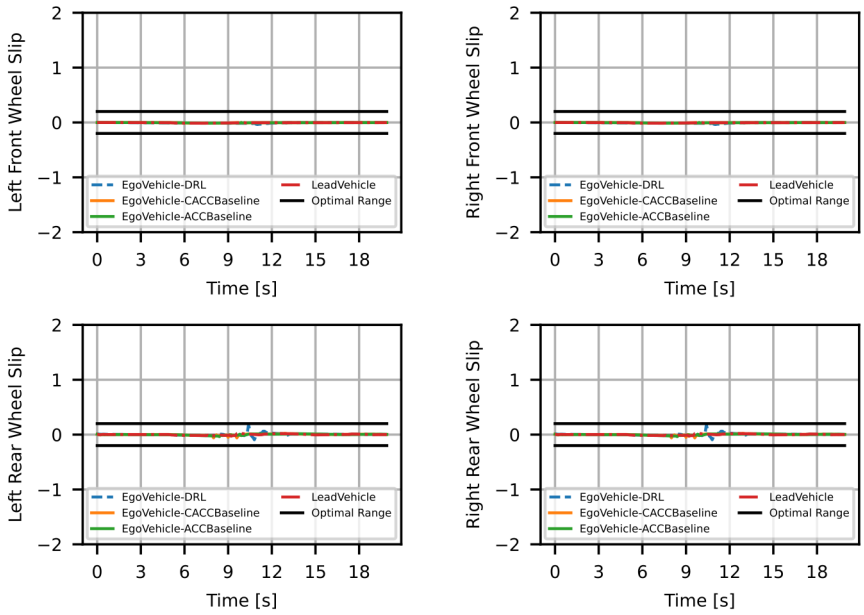

Figure 10. Traffic queuing scenario: Wheel longitudinal slip with respect to DRL, ACC, and CACC. The top (bottom) left and right plots represent the front (rear) left and right wheels, (resp.).

Table 1 provides the obtained values of RMSE as a performance index, comparing the DRL framework against the CACC and ACC algorithms. The DRL framework substantially outperforms the traditional approaches across all scenarios regarding traffic flow efficiency and stability. However, in terms of comfort, the (C)ACC algorithms exhibit lower RMSE values of jerk, and this is especially evident in the traffic queuing scenario. Nevertheless, the higher RMSE value in the traffic queuing scenario is mostly due to the jerk when the vehicle is at lower velocities. We can therefore observe that the proposed DRL framework can achieve an excellent balance between safety, efficiency, and comfort, ultimately providing an excellent experience for both drivers and passengers.

Table 1. Comparison of Root Mean Square Error (RMSE) values.

		RMSE		
Metrics	**Scenarios**	**DRL**	**ACC**	**CACC**
Headway (Ideal = 1.3)	Normal	0.0278	0.0638	0.073
	Sharp Deceleration	0.0629	0.5298	0.6561
	Traffic queuing	0.0839	1.7594	2.2911
Jerk (Ideal = 0)	Normal	0.2111	0.1804	0.1693
	Sharp Deceleration	1.846	2.471	2.5839
	Traffic queuing	2.8181	1.1773	1.2699
Slip (Ideal = 0)	Normal	0.0028	0.0029	0.0028
	Sharp Deceleration	0.0071	0.0581	0.0594
	Traffic queuing	0.0202	0.01	0.0115

Finally, to assess the importance of V2V communication, we remove the lead vehicle acceleration $a(t)$ from the DRL framework states and train the DDPG policy for the traffic queuing scenario. While pre-training the model, the DRL, without the lead vehicle's acceleration, converged to a stabilized control policy only after 1750 iterations and attained a maximum reward of 0.71. On the contrary, the DRL framework with lead vehicle acceleration was able to converge on the optimal control policy in 1250 iterations during pre-training, achieving a maximum reward of 0.76. This strongly suggests that the lead vehicle acceleration information is a vital component of the DRL states and that it considerably aids the DDPG algorithm learning and converging to the optimal control policy efficiently.

5. Conclusions

We proposed a deep reinforcement learning (DRL) approach to enhance the adaptive cruise control system in connected autonomous vehicles. The proposed strategy aims to achieve traffic efficiency, safety (including vehicle stability), and comfort by integrating and appropriately weighting headway, longitudinal slip, and jerk. In contrast to traditional ACC and cooperative ACC schemes, the proposed method offers much better overall performance. Importantly, the DRL approach outperforms the traditional CACC and ACC algorithms in headway performance by 36% in totality and by 47% during the lead vehicle's speed variation phases, thus resulting in higher traffic flow efficiency under both highway and urban conditions. In addition, the RMSE comparison reveals that the proposed method can achieve a good balance between safety, comfort, and efficiency, maximizing traffic efficiency and enhancing the overall driving experience. Importantly, these results have been obtained under a realistic model of vehicle dynamics and various difficult scenarios. Finally, we demonstrated that the information gathered via V2X communication concerning lead vehicle acceleration is a crucial component of the DRL-based ACC application, which yields significant performance improvements. As future work, we will focus on examining the effect of the vehicle state's estimation uncertainty on the performance of the DRL. Furthermore, an interesting research direction consists of incorporating our DRL algorithm in other frameworks aimed at maximizing fuel efficiency and reducing vehicles' carbon footprint.

Author Contributions: Conceptualization, C.F.C., N.A. and F.D.; Data curation, D.C.S. and S.H.; Methodology, D.C.S., S.H., C.F.C., N.A. and F.D.; Project administration, C.F.C., N.A. and F.D.; Supervision, C.F.C., N.A. and F.D.; Validation, D.C.S. and S.H.; Writing—original draft, D.C.S.; Writing—review and editing, D.C.S., S.H., C.F.C., N.A. and F.D. 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: Data are not available due to restrictions.

Acknowledgments: This work was developed in the framework of the activities of the Interdepartmental Center for Automotive Research and Sustainable Mobility (CARS) at Politecnico di Torino (www.cars.polito.it, accessed on 19 April 2023).

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

Abbreviations

The following abbreviations are used in this manuscript:

ACC	Adaptive Cruise Control
DRL	Deep Reinforcement Learning
ML	Machine Learning
V2V	Vehicle-to-Vehicle Communication
V2X	Vehicle-to-Everything Communication
V2I	Vehicle-to-Infrastructure Communication
ADAS	Advanced Driver Assistance Systems
GNNS	Global navigation satellite system
RL	Reinforcement Learning
DDPG	Deep Deterministic Policy Gradient
CACC	Cooperative Adaptive Cruise Control
MPC	Model Predictive Control
CoMoVe	Communication, Mobility, and Vehicle dynamics
MDP	Markov Decision Process
DP	Dynamic Programming
TD	Temporal-Difference
TTC	Time-to-Collision
RMSE	Root Mean Square Error

References

1. World Health Organization. *Global Status Report on Road Safety 2018*; WHO: Geneva, Switzerland, 2018. Available online: https://www.who.int/publications/i/item/9789241565684 (accessed on 19 April 2023).
2. Koglbauer, I.; Holzinger, J.; Eichberger, A.; Lex, C. Drivers' Interaction with Adaptive Cruise Control on Dry and Snowy Roads with Various Tire-Road Grip Potentials. *J. Adv. Transp.* **2017**, *2017*, 5496837. [CrossRef]
3. Das, S.; Maurya, A.K. Time Headway Analysis for Four-Lane and Two-Lane Roads. *Transp. Dev. Econ.* **2017**, *3*, 9. [CrossRef]
4. Li, Y.; Lu, H.; Yu, X.; Sui, Y.G. Traffic Flow Headway Distribution and Capacity Analysis Using Urban Arterial Road Data. In Proceedings of the International Conference on Electric Technology and Civil Engineering (ICETCE), Lushan, China, 22–24 April 2011; pp. 1821–1824. [CrossRef]
5. Shrivastava, A.; Li, P. Traffic Flow Stability Induced By Constant Time Headway Policy For Adaptive Cruise Control (ACC) Vehicles. In Proceedings of the 2000 American Control Conference. ACC (IEEE Cat. No.00CH36334), Chicago, IL, USA, 28–30 June 2000; Volume 3, pp. 1503–1508. [CrossRef]
6. Zhu, M.; Wang, Y.; Pu, Z.; Hu, J.; Wang, X.; Ke, R. Safe, efficient, and comfortable velocity control based on reinforcement learning for autonomous driving. *Transp. Res. Part C Emerg. Technol.* **2020**, *117*, 102662. [CrossRef]
7. Desjardins, C.; Chaib-draa, B. Cooperative Adaptive Cruise Control: A Reinforcement Learning Approach. *IEEE Trans. Intell. Transp. Syst.* **2011**, *12*, 1248–1260. [CrossRef]
8. Liu, W.; Xia, X.; Xiong, L.; Lu, Y.; Gao, L.; Yu, Z. Automated Vehicle Sideslip Angle Estimation Considering Signal Measurement Characteristic. *IEEE Sens. J.* **2021**, *21*, 21675–21687. [CrossRef]
9. Gao, L.; Xiong, L.; Xia, X.; Lu, Y.; Yu, Z.; Khajepour, A. Improved Vehicle Localization Using On-Board Sensors and Vehicle Lateral Velocity. *IEEE Sens. J.* **2022**, *22*, 6818–6831. [CrossRef]

10. Xia, X.; Hashemi, E.; Xiong, L.; Khajepour, A. Autonomous Vehicle Kinematics and Dynamics Synthesis for Sideslip Angle Estimation Based on Consensus Kalman Filter. *IEEE Trans. Control Syst. Technol.* **2023**, *31*, 179–192. [CrossRef]
11. Lillicrap, T.P.; Hunt, J.J.; Pritzel, A.; Heess, N.; Erez, T.; Tassa, Y.; Silver, D.; Wierstra, D. Continuous control with deep reinforcement learning. *arXiv* **2015**, arXiv:1509.02971. [CrossRef]
12. Selvaraj, D.C.; Hegde, S.; Chiasserini, C.F.; Amati, N.; Deflorio, F.; Zennaro, G. A Full-fledge Simulation Framework for the Assessment of Connected Cars. *Transp. Res. Procedia* **2021**, *52*, 315–322. [CrossRef]
13. Ma, Y.; Wang, Z.; Yang, H.; Yang, L. Artificial intelligence applications in the development of autonomous vehicles: A survey. *IEEE/CAA J. Autom. Sin.* **2020**, *7*, 315–329. [CrossRef]
14. Badue, C.; Guidolini, R.; Carneiro, R.V.; Azevedo, P.; Cardoso, V.B.; Forechi, A.; Jesus, L.; Berriel, R.; Paixão, T.M.; Mutz, F.; et al. Self-driving cars: A survey. *Expert Syst. Appl.* **2021**, *165*, 113816. [CrossRef]
15. Kuutti, S.; Bowden, R.; Jin, Y.; Barber, P.; Fallah, S. A Survey of Deep Learning Applications to Autonomous Vehicle Control. *IEEE Trans. Intell. Transp. Syst.* **2021**, *22*, 712–733. [CrossRef]
16. Matheron, G.; Perrin, N.; Sigaud, O. Understanding Failures of Deterministic Actor-Critic with Continuous Action Spaces and Sparse Rewards. In *Artificial Neural Networks and Machine Learning—ICANN 2020*; Farkaš, I., Masulli, P., Wermter, S., Eds.; Springer International Publishing: Cham, Switzerland, 2020; Volume 12397, pp. 308–320. [CrossRef]
17. Ye, Y.; Zhang, X.; Sun, J. Automated vehicle's behavior decision making using deep reinforcement learning and high-fidelity simulation environment. *Transp. Res. Part C Emerg. Technol.* **2019**, *107*, 155–170. [CrossRef]
18. Fu, Y.; Li, C.; Yu, F.R.; Luan, T.H.; Zhang, Y. A Decision-Making Strategy for Vehicle Autonomous Braking in Emergency via Deep Reinforcement Learning. *IEEE Trans. Veh. Technol.* **2020**, *69*, 5876–5888. [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. [CrossRef]
20. Pan, F.; Bao, H. Reinforcement Learning Model with a Reward Function Based on Human Driving Characteristics. In Proceedings of the 2019 15th International Conference on Computational Intelligence and Security (CIS), Macao, Macao, 13–16 December 2019; pp. 225–229. [CrossRef]
21. Dettmann, A.; Hartwich, F.; Roßner, P.; Beggiato, M.; Felbel, K.; Krems, J.; Bullinger, A.C. Comfort or Not? Automated Driving Style and User Characteristics Causing Human Discomfort in Automated Driving. *Int. J. Hum.–Comput. Interact.* **2021**, *37*, 331–339. [CrossRef]
22. Amaral, J.R.; Göllinger, H.; Fiorentin, T.A. Improvement of Vehicle Stability Using Reinforcement Learning. In Proceedings of the Anais do XV Encontro Nacional de Inteligência Artificial e Computacional (ENIAC 2018), São Paulo, Brazil, 22–25 October 2018; Sociedade Brasileira de Computação—SBC: São Paulo, Brazil, 2018; pp. 240–251. [CrossRef]
23. Lin, Y.; McPhee, J.; Azad, N.L. Comparison of Deep Reinforcement Learning and Model Predictive Control for Adaptive Cruise Control. *IEEE Trans. Intell. Veh.* **2021**, *6*, 221–231. [CrossRef]
24. Urra, O.; Ilarri, S. MAVSIM: Testing VANET applications based on mobile agents. In *Cognitive Vehicular Networks*; CRC Press: Boca Raton, FL, USA, 2016; pp. 199–224. [CrossRef]
25. Stevic, S.; Krunic, M.; Dragojevic, M.; Kaprocki, N. Development and Validation of ADAS Perception Application in ROS Environment Integrated with CARLA Simulator. In Proceedings of the 2019 27th Telecommunications Forum (TELFOR), Belgrade, Serbia, 26–27 November 2019; pp. 1–4. [CrossRef]
26. Watkins, C.J.C.H.; Dayan, P. Q-learning. *Mach. Learn.* **1992**, *8*, 279–292. [CrossRef]
27. Sutton, R.S.; Barto, A.G. *Reinforcement Learning: An Introduction*, 2nd ed.; The MIT Press: Cambridge, MA, USA, 2018; p. 526.
28. Rajamani, R. *Vehicle Dynamics and Control*; Mechanical Engineering Series; Springer US: Boston, MA, USA, 2012. [CrossRef]
29. Bae, I.; Moon, J.; Jhung, J.; Suk, H.; Kim, T.; Park, H.; Cha, J.; Kim, J.; Kim, D.; Kim, S. Self-Driving like a Human driver instead of a Robocar: Personalized comfortable driving experience for autonomous vehicles. *arXiv* **2020**, arXiv:2001.03908. [CrossRef]
30. Makridis, M.; Mattas, K.; Ciuffo, B. Response Time and Time Headway of an Adaptive Cruise Control. An Empirical Characterization and Potential Impacts on Road Capacity. *IEEE Trans. Intell. Transp. Syst.* **2020**, *21*, 1677–1686. [CrossRef]
31. Benalie, N.; Pananurak, W.; Thanok, S.; Parnichkun, M. Improvement of adaptive cruise control system based on speed characteristics and time headway. In Proceedings of the 2009 IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10–15 October 2009; pp. 2403–2408. [CrossRef]
32. Houchin, A.; Dong, J.; Hawkins, N.; Knickerbocker, S. Measurement and Analysis of Heterogenous Vehicle Following Behavior on Urban Freeways: Time Headways and Standstill Distances. In Proceedings of the 2015 IEEE 18th International Conference on Intelligent Transportation Systems, Gran Canaria, Spain, 15–18 September 2015; pp. 888–893. [CrossRef]
33. Wong, J.Y. *Theory of Ground Vehicles*, 3rd ed.; John Wiley: New York, NY, USA, 2001.
34. Selvaraj, D.C.; Hegde, S.; Amati, N.; Deflorio, F.; Chiasserini, C.F. An ML-Aided Reinforcement Learning Approach for Challenging Vehicle Maneuvers. *IEEE Trans. Intell. Veh.* **2023**, *8*, 1686–1698. [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.