Investigating the Effects of Cooperative Driving for CAVs in Different Driving Scenarios Using Multi-Driver Simulator Experiments

Lishengsa Yue, Ph.D.
Postdoctoral Researcher
Department
Dept. of Civil, Environmenta
& Construction Engineering

Mohamed Abdel-Aty, Ph.D.
Trustee Chair
Pegasus Professor and Chair
Dept. of Civil, Environmental
& Construction Engineering

Zijin Wang, B.E.
Graduate Research Assistant
Dept. of Civil, Environmental
& Construction Engineering

Investigating the Effects of Cooperative Driving for CAVs in Different Driving Scenarios Using Multi-Driver Simulator Experiments

Lishengsa Yue, Ph.D., PI
Postdoctoral Researcher
Department
Dept. of Civil, Environmental &
Construction Engineering
https://orcid.org/0000-0002-0864-0075

https://orcid.org/0000-0002-4838-1573

Zijin Wang,
Graduate Research Assistant
Dept. of Civil, Environmental &
Construction Engineering
https://orcid.org/0000-0002-3285-433X

Mohamed Abdel-Aty, Ph.D., Co-PI
Trustee Chair
Pegasus Professor and Chair
Dept. of Civil, Environmental &
Construction Engineering

A Report on Research Sponsored by

SAFER-SIM University Transportation Center

Federal Grant No: 69A3551747131

Aug 2022

DISCLAIMER

The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. The report is funded, partially or entirely, by a grant from the U.S. Department of Transportation's University Transportation Centers Program. However, the U.S. government assumes no liability for the contents or use thereof.

TECHNICAL REPORT DOCUMENTATION PAGE

General instructions: To add text, click inside the form field below (will appear as a blue highlighted or outlined box) and begin typing. The instructions will be replaced by the new text. If no text needs to be added, remove the form field and its instructions by clicking inside the field, then pressing the Delete key twice.

Please remove this field before completing form.

<table>
<tr><td>1. Report No.
[SAFER-SIM admin at UI to complete]</td><td>2. Government Accession No.</td><td>3. Recipient’s Catalog No.</td></tr>
<tr><td colspan="2" rowspan="2">4. Title and Subtitle
Investigating the Effects of Cooperative Driving for CAVs in Different Driving Scenarios Using Multi-Driver Simulator Experiments</td><td>5. Report Date
Aug 29, 2022</td></tr>
<tr><td>6. Performing Organization Code
Enter any/all unique numbers assigned to the performing organization, if applicable.</td></tr>
<tr><td colspan="2">7. Author(s)
Lishengsa Yue, Ph.D. https://orcid.org/0000-0002-0864-0075
Mohamed Abdel-Aty, Ph.D. https://orcid.org/0000-0002-4838-1573
Zijin Wang, https://orcid.org/0000-0002-3285-433X</td><td>8. Performing Organization Report No.
Enter any/all unique alphanumeric report numbers assigned by the performing organization, if applicable.</td></tr>
<tr><td colspan="2" rowspan="2">9. Performing Organization Name and Address
Enter the name and address of the organization(s) performing the research.</td><td>10. Work Unit No.</td></tr>
<tr><td>11. Contract or Grant No.
Safety Research Using Simulation (SAFER-SIM) University Transportation Center
(Federal Grant #: 69A3551747131)</td></tr>
<tr><td colspan="2" rowspan="2">12. Sponsoring Agency Name and Address
Safety Research Using Simulation University Transportation Center
Office of the Secretary of Transportation (OST)
U.S. Department of Transportation (US DOT)</td><td>13. Type of Report and Period Covered
Final Research Report (Month YYYY – Month YYYY)</td></tr>
<tr><td>14. Sponsoring Agency Code</td></tr>
<tr><td colspan="3">15. Supplementary Notes
This project was funded by Safety Research Using Simulation (SAFER-SIM) University Transportation Center, a grant from the U.S. Department of Transportation – Office of the Assistant Secretary for Research and Technology, University Transportation Centers Program.
The contents of this report reflect the views of the authors, who are responsible for the facts and the accuracy of the information presented herein. This document is disseminated in the interest of information exchange. The report is funded, partially or entirely, by a grant from the U.S. Department of Transportation’s University Transportation Centers Program. However, the U.S. government assumes no liability for the contents or use thereof.</td></tr>
</table>

16. Abstract

Enter a brief factual summary of the most significant information, including the purpose, methods, results, and conclusions of the work. When appropriate, the abstract should include advice on how the results of the research can be used. For guidance, please see ANSI/NISO Z39.14-1997 (R2015) Guidelines for Abstracts (http://www.niso.org/apps/group_public/project/details.php?project_id=124).

Cooperative driving powered by connected vehicle (CV) technology is expected to improve traffic safety and efficiency, especially at locations with dense vehicle interactions. Although lots of research have developed their cooperative driving algorithms for different locations, the effects of human drivers in the loop and multi-agent driving decision making are less studied.

In this project, three tasks were investigated: (1) developing cooperative driving strategies (CDS) for non-signalized intersections in a mixed traffic environment, and testing its effects in different automation level and market penetration rate; (2) proposing human-machine-interfaces (HMI) for non-signalized intersection cooperative driving, and evaluating the performance of different HMIs in various traffic conditions; (3) training a cooperative decision-making strategy for cooperative diverging at freeway off-ramp based on multi-agent reinforcement learning. UCF-SST self-developed human-in-the-loop co-simulation platform was used to complete the tasks.

For task 1, an efficiency-oriented CDS was developed for mixed traffic cases, and tested on different CV and CAV market penetration rates. The experiment results showed that the proposed CDS reduced up to 53.8%, 66.4%, and 73.7% of travel time in CV-HDV (human-driven vehicle), CV-CAV, and CAV environments, respectively. For task 2, a driver-centered CDS was developed by modifying the algorithm in task 1, and then three different cooperative driving HMIs were evaluated by simulators. The results suggest a graphic-based HMI is better at displaying minor speed change requirements to the drivers, and it can guide the drivers approaching an intersection with better precision. For task 3, a multi-agent deep-Q network (MADQN) was trained for decision-making on freeway off-ramp diverging driving scenarios. The trained model significantly outperformed the baseline model in terms of efficiency and safety while ensuring decent successful diverging rate.

17. Key Words

Cooperative driving, Non-signalized intersection, Human-machine interface, Off-ramp, Co-simulation, Reinforcement learning

18. Distribution Statement

No restrictions. This document is available through the SAFER-SIM website, as well as the National Transportation Library

19. Security Classif. (of this report)	20. Security Classif. (of this page)	21. No. of Pages	22. Price
Unclassified	Unclassified	Enter the total number of pages in the report, including both sides of all pages and the front and back covers.	

Table of Contents

List of Figures

List of Tables

Abstract

Cooperative driving powered by connected vehicle (CV) technology is expected to improve traffic safety and efficiency, especially at locations with dense vehicle interactions. Although lots of research have developed their cooperative driving algorithms for different locations, the effects of human drivers in the loop and multi-agent driving decision making are less studied.

In this project, three tasks were investigated: (1) developing cooperative driving strategies (CDS) for non-signalized intersections in a mixed traffic environment, and testing its effects in different automation level and market penetration rate; (2) proposing human-machine-interfaces (HMI) for non-signalized intersection cooperative driving, and evaluating the performance of different HMIs in various traffic conditions; (3) training a cooperative decision-making strategy for cooperative diverging at freeway off-ramp based on multi-agent reinforcement learning. UCF-SST self-developed human-in-the-loop co-simulation platform is used to complete the tasks.

For task 1, an efficiency-oriented CDS was developed for mixed traffic cases, and tested on different CV and CAV market penetration rates. The experiment results showed that the proposed CDS can reduce up to 53.8%, 66.4%, and 73.7% of travel time in CV-HDV (human-driven vehicle), CV-CAV, and CAV environments, respectively. For task 2, a driver-centered CDS was developed by modifying the algorithm in task 1, and then three different cooperative driving HMIs were designed and evaluated on the simulators. The results suggest a graphic-based HMI is better at displaying minor speed change requirements to the drivers, and it can guide the drivers approaching an intersection with better precision. For task 3, a multi-agent deep-Q network (MADQN) was trained for decision-making on freeway off-ramp diverging driving scenarios. The trained model significantly outperformed the baseline model in terms of efficiency and safety while ensuring decent successful diverging rate.

1 Introduction

Traffic conflicts and accidents appear more frequently in road sections with intensive vehicle interactions. Minor driver misjudgments or improper decisions due to the incomplete perception of vehicle interactions may lead to severe traffic conflicts or even accidents. According to a National Highway Traffic Safety Administration (NHTSA) study, driver error led to 94% of the crashes [1]. Furthermore, the U.S. General Services Administration (GSA) reported that human error causes 98% of crashes [2].

Freeway off-ramps and non-signalized intersections are generally considered as locations that involve plenty of vehicle interactions, especially during peak hours. Near the freeway exit, if a vehicle traveling in the inner lane wants to leave the freeway, it has to traverse more than two lanes to reach the freeway exit on the right side. During this lane change process, intensive vehicle interaction will occur between diverging vehicles and through vehicles, particularly when the traffic is heavy. In addition, if a driver misses the best timing to drive to the outside lane, the driver may be forced to change the lane in an abrupt way, which raises crash risks and deteriorates safety.

Similar to the freeway exit, non-signalized intersections are also regarded as hotpots with sophisticated driving behavior and plenty of vehicle interactions. At no-signalized intersections, vehicles from different approaches may enter the intersection simultaneously and strongly interact with each other. In a traditional human-driven vehicle (HDV) environment, drivers approaching the intersection are not aware of the position and speed of other vehicles until they become visible or close enough. Typically, a driver would wait at the stop sign first and then determine whether he/she has the priority to drive through. However, the right-of-way is not always clear, making drivers

hesitate whether to continue driving or to stop. In some cases, the involved drivers may enter the intersection simultaneously and some of them may have to initiate a sudden brake later to yield to others, which not only significantly deteriorates traffic efficiency and driving comfort, but also increases the risks of vehicle conflicts or even crash.

The recent advancement of connected and automated vehicle (CAV) technology brought unprecedented opportunities to develop and implement cooperative and automated driving strategies, and diminish human driving errors and enhance traffic safety at abovementioned two locations. The cooperative driving strategy (CDS) is expected to mitigate or even avoid intensive vehicle interactions in complicated driving environments by enabling vehicle dynamics (position, speed, acceleration, headway, etc.) and driving intention (lane change, driving priority, etc.) to be broadcasted to the involved vehicles, and thus enable collaborative decision making and vehicle control to improve driving safety and efficiency.

Recently, CDS of all sorts of control mechanisms have been proposed. However, there are a few limitations that they did not address. First, most of the existing CDS are developed in a connected and automated vehicle (CAV) environment that assumes autonomous vehicle control with absolute precision. However, this is still far from application as no such CAV environment is available on road up to date. Instead, cooperative driving between CVs could be realized in the foreseeable future as onboard unit (OBU) with communication capability becomes more and more prevalent. Therefore, CDS for CV that considers driver acceptance and driving imperfection should be developed.

Second, the cooperative driving suggestions (e.g., desired speed) are delivered to the CV drivers through in-vehicle human-machine interface (HMI), such as head-up display (HUD). The design of the HMI is expected to affect the drivers' acceptance and

response to the driving guidance, and further impacts the cooperation between vehicles. However, despite some studies focused on the design of HMI, none of them is designed for cooperative driving that provides CV drivers with real-time speed guidance. In the non-signalized cooperative driving scenario, the CV drivers may be requested to take acceleration or deceleration action to avoid a potential conflict when approaching the intersection, however the exact action and scale of such action required is difficult to interpret and deliver to the CV drivers. Due to this reason, the HMI design should consider the drivers' acceptance and adaptability to the real-time speed suggestion, as the HMI is supposed to display the driving guidance effectively with minimum cognitive load for the CV drivers.

Third, most of the existing CDS algorithms are model-based algorithms (e.g. model prediction methods, rule-based models), which might be outperformed by state-of-the-art model-free methods like deep learning and reinforcement learning. Although there has already been some machine learning-based research focused on vehicle cooperation at many locations such as the on-ramp merging, the algorithms for some other driving environments like off-ramp diverging remain largely unexplored. Furthermore, the domain of multi-vehicle cooperation is less studied, which is crucial in heavy traffic circumstance as the computing demand increases exponentially as the number of vehicles increases.

To address the abovementioned issues and fill the research gaps, this study aims at proposing CDS algorithms for both the non-signalized intersection and off-ramp locations, and testing the CDS through microscopic simulation driving simulator experiments. There are three tasks to be fulfilled in this research:

Task 1: Propose CDS for non-signalized intersection

To develop an ad-hoc negotiation-based cooperative driving algorithm for CV at the non-signalized intersection, and test the performance of CDS in a mixed traffic environment with different CV and CAV market penetration rates (MPR) by conducting multi-driver simulation experiments.

Task 2: Design cooperative driving HMI

To design different in-vehicle HMIs for CDS at non-signalized intersections that provide CV drivers with real-time driving speed suggestions in an understandable way. The effects of different HMIs are to be investigated through multi-driver-in-the-loop co-simulation experiments.

Task 3: Reinforcement learning at the off-ramp

To propose a multi-agent reinforcement learning algorithm that cooperates off-ramp diverging vehicles and through vehicles by learning an optimal policy for lane change and vehicle following decision making. Microscopic traffic simulation will be used to train the model and test its performance in various traffic demand conditions.

2 Literature Review

2.1 Cooperative Driving

The concept of cooperative driving was first described by the Association of Electronic Technology for Automobile Traffic and Driving in Japan in the early 1990s [3]. Its key idea is to organize and coordinate the movements of neighboring vehicles [4], [5]. Its implementation depends on fast and reliable wireless V2V communication. Research concerning this issue can be found in [6], [7]. There are various topics related to cooperative driving, ranging from developing CDSs for different scenarios to VANET design, to CDS evaluation considering multiple factors including V2V communication effects, road geometry, and vehicle kinematic information, etc.

2.1.1 Intersections

Cooperative driving at intersections has been widely studied. For signalized intersections, the prevalent topic is the advanced traffic signal control via vehicle-to-infrastructure (V2I) communication that is based on the assumption that vehicles' movements will only be controlled by traffic lights. The key element of this topic is the prediction of accurate arriving rates of traffic flows in the next few minutes so that the best traffic light timing that maximizes traffic efficiency can be calculated [8]–[10]. Xie et al. [11], He [12] and Gaur and Mirchandani [13] explored platoon-based traffic control. The idea is to group vehicles into several platoons before they arrive at the vicinity of the intersection.

Cooperative driving for non-signalized intersections is another research hotspot. Two different control methods of CDS have been widely studied: "ad hoc negotiation based" and "planning based". Efficiency improvement is the prior objective for their CDS. The first method makes vehicles roughly follow first-come-first-served order to pass the intersection, while the second one considers vehicles arriving within a certain spatial scope and formulates relatively long-term driving plans for vehicles. Tachet et al. [14] and Levin et al. [15] studied the first method, which is relatively simple but still practical. The planning-based CDS is expected to achieve better overall performance, and its effectiveness is tested by Zhang et al. [16] and Wu et al. [17]. The results indicate that cooperative control methods improve efficiency significantly compared to traditional traffic control methods, but when traffic load increases the performance deteriorated. Meng et al. [18] compared the two methods, and found out the planning-based methods yield much better performance in a high traffic demand scenario. Yang et al. [19] applied games theory to cooperative driving development for non-signalized intersections. Lee and Park [20] designed algorithms that eliminate the potential overlaps of vehicular trajectories coming from all conflicting approaches at the intersection. De Campos et al. [21] proposed

a velocity-based negotiation approach. In general, their CDS improves traffic efficiency by reducing travel time or traffic delay from 30% to 80%. Also, Lin et al. [22] developed CDS that considers mixed traffic scenarios. However, there is only coordination for CAV but no CAV-HDV coordination in the paper. Furthermore, the abovementioned CDSs are designed for a CAV environment, not for CV. As stated previously, it is necessary to develop CDS for CV to accelerate the implementation of cooperative driving.

2.1.2 Off-ramp

Although cooperative driving at off-ramps is less studied compared to on-ramp merging and weaving segment driving, there are several research that focus on collaborative decision-making or control mechanism at off-ramps. Dong et al. [23] used cooperative adaptive cruise control (CACC) to control vehicle following behavior to investigate the safety and efficiency benefits of CAV at the off-ramp. However, they did not consider lane change maneuver cooperatively but trained a human-like lane change decision model from ground truth trajectory data at the off-ramp. The paper by Wang et al. [24] did consider cooperative lane change by considering the lane utility and car-following models of the following vehicles in adjacent lanes. Zheng et al. [25] proposed a cooperative algorithm based on the coordination of behaviors between the diverging vehicle and its cooperative vehicle on the target lane. The algorithm was developed by modifying Minimizing Overall Braking Induced by Lane Changes Model (MOBIL) and Intelligent Driver Model (IDM).

Apart from the model-based algorithms, reinforcement learning (RL) has received increasingly attention in ramp-related optimization problems. Most of the RL algorithms are proposed to coordinate vehicles for on-ramp merging, nevertheless they share certain similarities with the off-ramp diverging problem and can provide insights for the design of off-ramp RL algorithm. The work done in [26] introduced a deep multi-agent RL algorithm

that applies safety masks to merging and through vehicles in a mixed traffic environment. A twin-delay Deep Deterministic Policy Gradient (DDPG) RL method is presented in [27] for decision-making at on-ramps, and the model showed great performance by reducing training time by 25%. Yu et al. [28] showed a distributed multiagent coordinated learning for CAV based on dynamic coordination graphs, which provides insights for the design of communication patterns for multi-vehicles on highways. Graph neural network has also been applied to represents the surrounding vehicle states in a mixed traffic environment, and it was used as input for a Deep Q-learning (DQN) algorithm for off-ramp diverging problem [29].

For the multi-agent domain, an independent MARL framework, called IQL, is proposed in [30], allowing each agent to learn independently and simultaneously while viewing other agents as part of the environment. More recently, parameter sharing method is widely applied with homogeneous agents [31], which bootstraps single-agent RL methods and learns an identical policy for each agent, and thus enables the handling of changes in the number of participating agents. A parameter-sharing A2C (MA2C) algorithm is proposed in their work to solve the fleet management problem and experimental results are given to confirm the performance. In [32], several state-of-the-art single RL algorithms are extended to the MARL with parameter sharing denoted as MAPPO and MAACKTR.

2.2 HMI Design

There has been a boom in a variety of Advanced Driving Assistance Systems (ADAS) in terms of driving safety, eco-driving, and travel time saving. Different in-vehicle HMIs have also been developed to display the ADAS functions to the drivers. Visual-based and audio-based HMIs are the most widely adopted modalities for in-vehicle ADAS. New

modalities including haptic-based and AR-based HMI have also been proposed recently. Based on different functionalities, the HMI can be divided into two types: the notification HMI and the guidance HMI. The first type aims at providing drivers with event-based notification or alert, while the second type displays and updates driving guidance (e.g., suggested speed) in real-time. Most HMIs for safety-related ADAS belong to the first type, and multiple HMIs for safety warnings have been developed that aim at alerting drivers the presence of pedestrians or surrounding vehicles [33], [34]. The second HMI type is usually for achieving functions of eco-driving or cooperative driving. This type of HMI aims at consistently delivering the driving guidance to drivers, such as real-time suggested speed or throttle usage. As drivers are receiving information consistently, they could easily be cognitively overloaded if the HMI fails to display the information in a clear and understandable way. Various visual and haptic-based eco-driving HMIs have been proposed and tested through driving simulator experiments or field tests. It is Azzi et al. [35] who first studied the effect of visual and haptic in-vehicle assistance on eco-driving. A more comprehensive design of the visual guidance on throttle usage is presented in Jamson et al. work [33]. They introduced three types of dash-based guidance methods: dot system, gauge system, and foot system; each system is built on different images with different colors representing current state (blue, green, and red corresponds to insufficient, appropriate, and excessive pedal pressure, respectively). The haptic guidance system is also investigated in their research, and it is recommended to use a strong force feedback system. Masola et al. [34] discussed the design rules and requirements for in-vehicle HMI from the perspective of how to design a suitable graphic interface for various driving tasks. They also presented an HMI design that combined multiple guidance functions in a dash-based display system, where the suggested speed and the current speed are displayed in sperate areas on the dashboard.

Cooperative driving at non-signalized intersections is aiming at changing vehicle speed to avoid unnecessary vehicle interaction at the intersection, and the cooperative driving HMI shall display and update the suggested speed to the drivers in real-time. As the optimal driving speed calculated by CDS may change consistently to adapt to the driving environment, the HMI needs to present the speed suggestion that quickly responds to the driving environment changes with high precision. However, the design of cooperative driving HMI that considers these requirements remains largely unexplored. The limited research on this cooperative driving HMI are mostly focused on assisting cooperative merging, which displays merging availability rather than speed guidance [36], [37]. For speed guidance HMI, the naive design that directly displays suggested speed is still widely adopted. As mentioned in [34], HMI with appropriate graphic design is expected to increase drivers' acceptance and reduce cognitive load. Therefore, it is worth investigating and develop a graphic HMI for cooperative driving.

2.3 Driving Simulator

Driving simulators, which have been widely used for studying driving behavior and evaluating emerging vehicular technologies, could be an effective tool to test the performance of HMI designs. Generally, driving simulators are largely used for driving behavior analysis from measures including traffic safety [38], [39], traffic efficiency [39], [40], and driving comfort [41], [42]. For HMI design, most of the research also relies on driving simulators [33], [43]–[45]. However, single driving simulator may not be capable for testing the HMI under cooperative driving scenarios, as it does not allow multiple drivers to cooperate and thus multi-driver simulator is needed. Over the years, limited multi-driver simulators have been developed and used for research [46]–[48]. Moreover, the advent of co-simulation between driving simulator and traffic microsimulation allows more complicated traffic of high-fidelity to be produced in the driving simulators [48], and

it could be used to study the driving performance in a mixed traffic environment. Hence, it is beneficial to conduct a co-simulation experiment to investigate the effects of the in-vehicle HMI designs in a mixed traffic environment.

3 Methodology

3.1 Apparatus

The schematic diagram of our self-developed simulation platform is shown in Figure 3.1. The simulation platform consists of two major modules: the multi-driver driving simulator module and the microscopic traffic simulation module. In the first module, the driver GUI clients are connected to the CARLA server through a network bridge, and the CDS is embedded in the clients via API. CARLA is an open-source autonomous driving simulator. The second module runs microscopic traffic simulation in SUMO, which is an open-source microscopic traffic simulator. The road network and driving behavior parameters in SUMO are calibrated using real-world trajectory data extracted from drone video. The real-world map is used to digital twin the simulation models in both modules. In the co-simulation mode, the two modules will run simultaneously and all the traffic participants are controlled synchronously. Data of the experiments is collected from both modules, and the driving behavior data and traffic flow data are matched according to vehicle ID and simulation timestep.

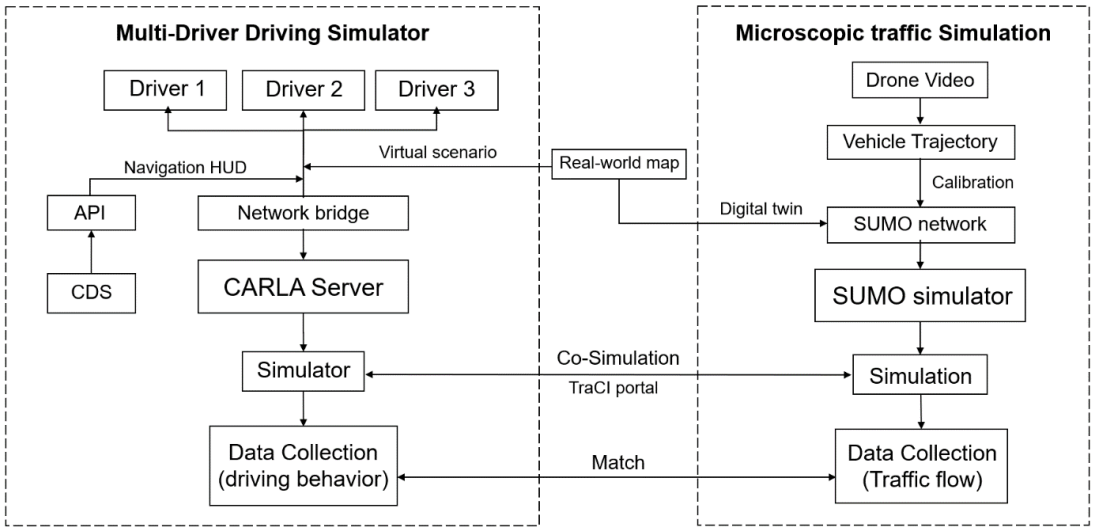


Figure 3.1 Schematic diagram of the simulation platform

3.1.1 Microscopic Traffic Simulation

The study area of this research is a 3-way stop control intersection inside the campus of the University of Central Florida in Orlando, FL, USA. The intersection is at the merge point of Aquarius Agora Drive, Greek Park Drive, and Gemini Boulevard (Figure. 3.2). It is a relatively busy intersection among stop control intersections, especially during commute hours. Queues are frequently observed during peak hours.

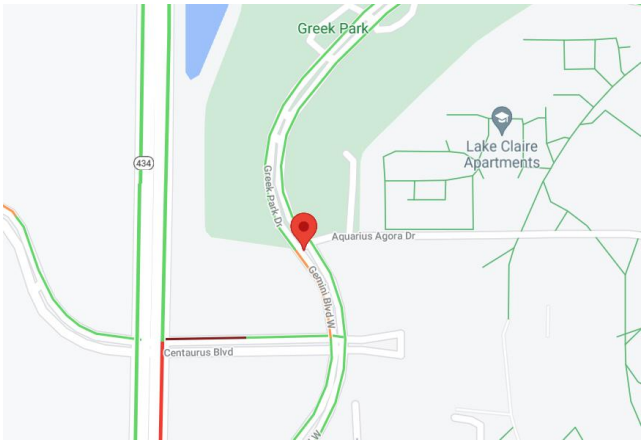


Figure 3.2 The object intersection

To reproduce real-world traffic flow and high-fidelity driving behaviors in simulation, we collected drone video data to extract vehicle trajectories for microscopic traffic simulation

model calibration. Drone videos were recorded at the intersection during peak hours in May 2020. Vehicle trajectories were extracted from the video, as shown in Figure. 3.3. Overall, more than 300 valid trajectories were collected. Afterwards, the parameters that are considered to affect driving behaviors at stop-control intersections are calculated and used for the simulation model calibration.

Figure 3.3 Vehicle tracking and trajectory extraction

The simulation model of the study area was built in SUMO (Figure 3.4). The road network in SUMO was generated from the Openstreetmap file. As shown in Figure 3.4, there are 3 entrances and exits of the intersection, whose start points or end points are marked as 1 to 6. The routes for entering the intersection are marked as R3, R1, and R5. To identify different routes of the intersection, each route is named after the combination of its start point and end point. For instance, the route starting from point 3 and ending at point 2 is marked as r32. These naming rules will be kept in the following paragraphs. The traffic flow and driving behavior parameters in the model were calibrated using trajectory data. The parameter values are shown in Table 3.1, including the trajectory-calibrated values and the SUMO default values. The parameters in the SUMO model were all default values except for those in Table 3.1.

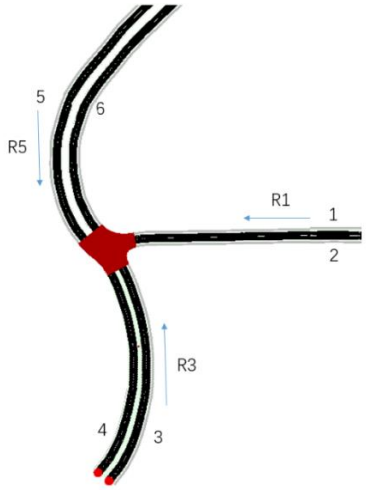


Figure 3.4 SUMO model

Table 3.1 SUMO model parameters

Parameters		**Description**	**Trajectory-calibrated value**	**SUMO default value**
minGap (m)		Empty space after leader	3.19	2.50
accel ($\mathrm{m/s^2}$)		The acceleration ability of vehicles	1.46	2.60
decel ($\mathrm{m/s^2}$)		The deceleration ability of vehicles	1.88	4.50
speedFactor		The vehicles expected multiplicator for lane speed limits	0.10	0.10
speedDev		The deviation of the speedFactor	0.10	0.10
departSpeed (m/s)	R3	The speed with which the vehicle shall enter the network	10.36	0.00
	R1		9.56	0.00
	R5		8.96	0.00
Probability (flows)	r32	Probability for emitting a vehicle each second	0.01	*NA*
	r36		0.05	*NA*
	r52		0.03	*NA*
	r54		0.06	*NA*
	r16		0.01	*NA*
	r14		0.01	*NA*

3.1.2 Multi-driver driving simulator

In order to create a real-world-like virtual scene and provide a simulation platform that enables the interaction between subject vehicles, we developed our own multi-driver driving simulator (Figure. 3.5). The system was developed based on the Open-source autonomous driving simulator CARLA. The driving simulator system connects up to 3 drivers to the CARLA server through a network bridge, and thus the drivers can simultaneously operate their vehicles in the same simulation environment. The data collection module was created to store the drivers' driving behavior variables. Also, we have developed a driver navigation HUD module, which sends the navigation information and driving instructions calculated by the CDS to the drivers.

Figure 3.5 UCF SST Multi-driver driving simulator

The virtual scene in the simulator was created based on Openstreetmap and GIS data using CARLA-supported map creator RoadRunner. The map was duplicated in a way similar to the real-world geometry and surrounding environment to provide a verisimilar driving scene. The virtual scene is shown in Figure. 3.6.

Figure 3.6 Virtual scene in CARLA

3.1.3 Co-simulation

CARLA simulator and SUMO simulation were connected through the TraCI portal provided by SUMO, and real-time co-simulation is achieved. In the co-simulation mode, the road network in SUMO and the map in CARLA were matched, and all the vehicles and other traffic participants were synchronized in both software. When conducting driving simulator experiments, the vehicle generated by SUMO acted as background traffic which simulates the real-world traffic flow, and the vehicles spawned by CARLA were operated by the drivers. Hence, the drivers were interacting with vehicles in the real-world traffic flow.

3.2 Cooperative Driving Strategy

3.2.1 Task 1

For task 1, we developed an “ad hoc negotiation based” cooperative driving strategy, which is based on the idea of first-come-first-served passing order. The CDS was developed based on the following assumptions:

(1) Roadside units (RSU) are built at the intersection. It is the calculation center of the CDS. The RSU has a data collection and transmission range of 150 m from the boundary of the intersection, which meets the standard of the Dedicated Short-Range Communications (DSRC) Standards in the United States [49]. The area within the 150 m communication range is named the Control Zone in the rest of the paper (Figure. 3.7).

(2) The RSU receives information CV and CAV, including destination, position, speed and lane information collected when they are inside the control zone. Also, the same information of HDV can be collected by RSU through camera or Lidar tracking. The error for data transmission and vehicle perception is ignored.

(3) Once the RSU calculate the optimal passing schedule and form driving instruction, it is sent to the CV and CAV. CV drivers will receive messages for arrival time and speed suggestions, and CAV motion strictly follows the instruction.

(4) With the CDS embedded, CV and CAV are not required to stop at the stop sign but to drive pass the intersection, if no conflict is expected. The vehicle under cooperation is not allowed to make overtaking maneuvers in the control zone.

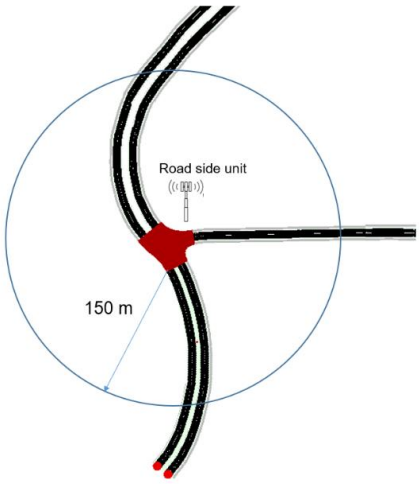


Figure 3.7 Control Zone

The CDS was designed to find an optimal vehicle passing schedule and travel speed for the upstream traffic of the intersection, and to achieve maximum throughput efficiency in a safe driving manner. To describe the objective of optimizing the passing schedule with minimum travel time (the time it takes from entering the control zone to the start point of the intersection), the following mathematical expression is used:

$$min\ T = \sum_{i=1}^{m} t(CV_i) + \sum_{j=1}^{n} t(CAV_j) \tag{3.1}$$

where T is the sum of the estimated travel times of the CV and CAV in the control zone, t is the estimated travel time for a single vehicle, and m and n refers to the number of CV and CAV, respectively. The travel time of HDVs is not considered in the equation, this is because they are unable to receive messages and will thus fail to follow the CDS. To achieve maximum traffic throughput efficiency, the vehicles should be instructed to drive as fast as possible under the speed limit. Also, driving at the instructed speed is expected to lead the CV/CAV into the intersection without causing a conflict. That is, there must be a sufficient time gap between the arrival times of two consecutively arriving vehicles at the conflict point. In the words, the interval of estimated arrival time for the first arriving vehicle should not overlap with the arriving time interval of the second arriving vehicle. Considering the geometry of the object intersection, we set the maximum driving speed to 55 kph. Vehicles that are under cooperation will be driving at the maximum speed unless instructed by the CDS to slow down. To better express the idea of saving travel time, we use the term arrival time to replace travel time, which refers to the time duration that the vehicle spends traveling from the current position to the start of the intersection or the conflict point of two routes. The arrival time is estimated based on the vehicle's current position and speed, the driving parameters (trajectory calibrated), and random effects. The following expressions describe the optimization problem of forming an optimal vehicle passing schedule:

$$min\, T = \sum_{i}^{n_{veh}} t_i^{arr}(veh_i) \quad \text{for } veh\text{=CV/CAV} \tag{3.2}$$

$$\text{s.t. } [t_{ij} - t_{gap} + \alpha_i,\ t_{ij} + t_{gap} + \alpha_i] \cap [t_{i-1,j} - t_{gap} + \alpha_{i-1},\ t_{i-1,j} + t_{gap} + \alpha_{i-1}] = \varnothing$$

$$\text{s.t. } v_i < v_{\max}$$

where t_i^{arr} is the arrival time for vehicle i to reach conflict point j, t_{gap} is the time gap for safety consideration (including gap for HDV, CV, and CAV, marked as t_{gap_HDV}, t_{gap_CV}, t_{gap_CAV}), α is the term of a random variable to simulate the stop-sign stop time for HDV that follows $N(1,1)$, and v_i is the speed of vehicle i at the current CDS calculation step, $v_{\max} = 55$ kph is the maximum driving speed. The term $[t_{ij} - t_{gap} + \alpha_i,\ t_{ij} + t_{gap} + \alpha_i]$ refers to the time interval of the estimated arrival time to a conflict of the vehicle, marked as arrival time interval (ATI). In this research, t_{gap_HDV}, t_{gap_CV}, t_{gap_CAV} are set to 2 s, 1.5 s, and 1.5 s respectively. The time gaps of HDV are larger than CV/CAV, because HDV is not driving under CDS instruction, therefore the arrival time for HDV is harder to estimate and needs a larger gap to cancel HDV drivers' driving randomness.

The key element of arranging a vehicle passing schedule is to precisely estimate the arrival time. Considering the differences in the driving behaviors between HDV, CV and CAV, we use different methods to estimate their arrival time. For HDV, we use a three-stage calculation: constant speed stage, deceleration stage, and acceleration stage. The three stages refer to the state of (1) driving at approximately constant speed before arriving at the intersection; (2) about to reach the intersection, the vehicle starts to decelerate and prepare to stop; (3) accelerating after stopping at the stop sign. The acceleration and deceleration process are treated as uniform acceleration/deceleration motion, which is the default by SUMO and provides good simulation results. Therefore, the arrival time of HDV can be calculated by:

$$t^{arr} = t_{con} + t_{dec} + t_{acc} \tag{3.3}$$

$$t_{con} = (d - d_{dec}) / v \tag{3.4}$$

$$t_{dec} = v / veh_{dec} \tag{3.5}$$

$$t_{acc} = \sqrt{2d_{acc} / veh_{acc}} \tag{3.6}$$

$$d_{dec} = v^2 / (2veh_{dec}) \tag{3.7}$$

where t_{con}, t_{dec}, t_{acc} are the arrival time for the three stages respectively, *d* is the vehicle's distance to the start point of the intersection, d_{dec} is the deceleration distance, d_{acc} is the acceleration distance (which is the same as the distance from the start point of the intersection to the conflict point), veh_{dec} and veh_{acc} are the vehicles' average deceleration and acceleration value (trajectory calibrated). For HDV, veh_{dec} =1.88 $\mathrm{m/s^2}$, and veh_{acc} =1.46 $\mathrm{m/s^2}$ (Table 3.1).

The CV and CAV do not have to stop at the intersection, if traffic state at the intersection allowed, thus it is easier to calculate their arrival time:

$$t^{arr} = (d + d_{acc}) / v \tag{3.8}$$

where *d* and d_{acc} are the distance to the intersection and the distance between the start of the intersection and conflict point. If the vehicle is expected to make a turn at the intersection, then it has to decelerate to the maximum allowed speed for turning $v_{turn} = 36$ kph to make the corner. The time duration of the deceleration and acceleration process can be calculated in the same ways as calculating HDV's.

There are totally 6 conflict points in the objective intersection, as shown in Fig. 3.8. For each vehicle, the arrival times at conflict points that it may encounter are calculated.

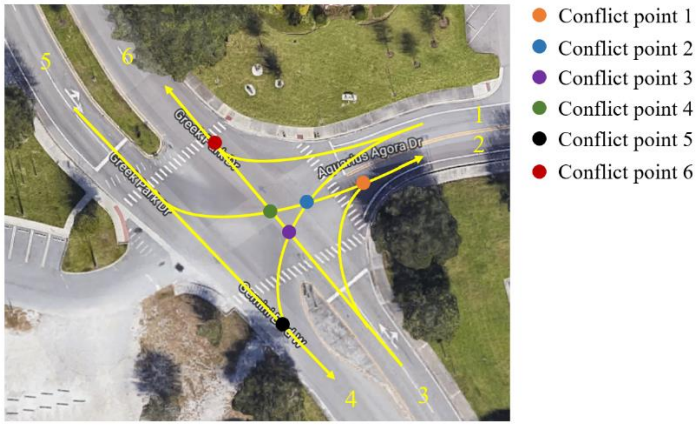


Figure 3.8 Conflict points

Once the arrival times for all vehicles are estimated, the cooperation process begins. The basic idea of the cooperation is: the vehicles change their speed to adjust their arrival time so that overlaps between consecutive arrival time intervals (ATI) are canceled. Only CV and CAV can achieve V2V/V2I communication, thus HDV will not participate in the cooperation process and will not adjust its speed. It is easy to presume that the original arrival time without CDS optimization is not the best and sometimes may even lead to potential conflicts if vehicles keep driving at current speed. Hence, the schedule needs to be optimized and the driving speed needs to be adjusted for both CV and CAV. For each conflict point, the arrival times of all vehicles that encounter the conflict point are sorted by descending order. Vehicles with smaller arrival times have the priority of passing first. If there is no overlap between the arrival time intervals (ATI) of two consecutively arriving vehicles, then the second vehicle does not have to yield to the first vehicle, and the pass schedule needs no change. However, if overlap is found, the second vehicle must decelerate to arrive later compared to the original pass schedule. The mathematical expression of the control mechanism is shown as follow:

$$t_i^{arr} = \max(t_{ij}^{arr} - t_{ij}^{add}) \quad i \in n_{veh}, j \in n_{CP} \tag{3.9}$$

$$t_{ij}^{arr} = \begin{cases} t_{ij}^{arr} & \text{if} \quad t_{ij}^{arr} < t_{i-1,j}^{arr} + 2t_{gap} + \alpha_{i-1} - \alpha_i \\ t_{i-1,j}^{arr} + 2t_{gap} + \alpha_{i-1} - \alpha_i & \text{else} \end{cases} \tag{3.10}$$

where t_{ij}^{arr}, t_{gap}, α are the same as previously mentioned, and t_{ij}^{add} is the estimated time that the vehicle spends on d_{acc} (the gap between intersection start point and conflict point *j*). The time estimation method is the same as shown in formula 10.

For a single vehicle, the optimized arrival time for the vehicle is the maximum arrival time among all the conflict points. Also, there are two situations that the CDS will stop providing messages to CV drivers: (1) If the CV is following a slow-driving HDV in the same lane inside the control zone as overtake is not allowed in the control zone; (2) although the estimated arrival time for HDV is larger than the CVs' at the common conflict point, the HDV arrived first at the intersection and stopped at the stop sign. In these cases, the CV is expected to be affected by and interact with HDV, and giving CDS instruction is inappropriate.

It should be noted that only CV and CAV can adjust their arrival time. The arrival time for HDV is fixed based on the estimation because they cannot receive CDS messages and follow the driving instructions. For CV, the CDS messages are given to the drivers with the suggested arrival time and driving speed. CAV's motion will be controlled directly by the CDS, including speed, acceleration, and path. The CAV speed planning uses a simple method: For each CDS execution step, the algorithm judges whether its current speed is equal to the desired speed. If not, the CAV accelerates or decelerates to approach the desired speed with a constant acceleration or deceleration value. For CV/CAV, veh_{acc} and veh_{dec} are set to $2\ \mathrm{m/s^2}$ and $4\ \mathrm{m/s^2}$, respectively. The parameters used for the CDS are shown in Table 3.2.

Table 3.2 Parameters of the CDS

Parameter	Description	Value
v_{max}	Maximum allowed speed for CV and CAV	55 kph
v_{turn}	Maximum turning speed for CV and CAV	36 kph
veh_{acc} (HDV)	Vehicle acceleration for HDV	1.46 $\mathrm{m/s^2}$
veh_{dec} (HDV)	Vehicle deceleration for HDV	1.88 $\mathrm{m/s^2}$
veh_{acc} (CV/CAV)	Vehicle acceleration for CV/CAV	2 $\mathrm{m/s^2}$
veh_{dec} (CV/CAV)	Vehicle deceleration for CV/CAV	4 $\mathrm{m/s^2}$
t_{gap} (HDV)	Safe time gap for HDV	2 s
t_{gap} (CV/CAV)	Safe time gap for CV/CAV	1.5 s
α	Random stop time of the HDV	*N* (1,1)

3.2.2 Task 2

The CDS introduced in task 1 is proposed to save driving travel time and improve traffic throughput efficiency, which makes the drivers drive at maximum speed without causing a conflict. It may request the drivers to accelerate to a much higher speed once entering the CDS control zone. However, whether the drivers can properly follow the CDS suggested speed is in doubt. In order to alleviate the driving load of fast acceleration, an automatic speed adaptive CDS is proposed based on the algorithm in task 1.

Different to efficiency oriented CDS in task 1 which maximize vehicle speed, the automatic speed adaptive CDS aims at cooperating vehicles to drive pass the non-signalized intersection by making the least speed adjustment without causing any conflict. The algorithm takes the current speed and distance to the intersection of all vehicles in the control zone as input, and outputs the closet speed to current speed that can avoid conflict with oncoming vehicles. Hence, the drivers can drive pass the intersection safely with the minimum speed adjustment. It significantly reduces the driving load of acceleration or deceleration and simplifies the task of following the CDS

speed guidance. The optimization objective of the automatic speed adaptive CDS is expressed as:

$$\min \Delta T = \sum_{i=1}^{n} (|\Delta v_i|) = \sum_{i=1}^{n} (|v_i^{current} - v_i^{optimal}|),\ i \in CV \tag{3.11}$$

where Δv is the speed difference of vehicle i, which is the absolute value of current speed minus the optimal speed. ΔT is the sum of speed difference for all CV vehicles, n is the total number of CV vehicles, and *CV* is the set of all connected vehicles.

At each timestep, the algorithm executes once and calculates the optimal speed for all the vehicles in the control zone. To prevent generating an extreme speed difference for the controlled vehicles and ensure driving safety, vehicle dynamic constraints and safety constraints are implemented as show in below.

$$|v_i^{current} - v_i^{optimal}| < \Delta v_{\max}, i \in CV \tag{3.12}$$

$$|t_i - t_j| > \Delta t_{\min},\ i \neq j; i, j \in V \tag{3.13}$$

$$dec_{\max} < a_i < acc_{\max},\ i \in CV \tag{3.14}$$

where t_i is the expected time that vehicle i arrives at the CDS control line (30 meters ahead of the intersection stop line), $dec_{\max}$ and $acc_{\max}$ is the maximum deceleration and acceleration ability of the vehicle, $\Delta v_{\max}$ is the maximum speed change, $\Delta t_{\min}$ is the minimum safety gap, and *V* is the set of all vehicles in the control zone including CV and HDV. $\Delta v_{\max}$ is 5 m/s and $\Delta t_{\min}$ is 3 seconds. $dec_{\max}$ and $acc_{\max}$ is set to 4.5 m/s^2 and 2.6 m/s^2 respectively, which align with the Sumo default settings. The estimation of t_i is based on vehicle's current speed and position, and the calculation method is the same as in section 3.2.1. The reason for choosing 30 meters ahead of intersection stop line as the control line is that at this position most of the vehicles start to decelerate based on

the drone video, which means the drivers finished cruising or car following period and therefore do not need speed guidance.

The first constraint limits the speed adjustment range by preventing the optimal speed greater or smaller than 5 m/s compared with current speed. The second constraint ensure driving safety by ensuring the arrival time at the control line for any two vehicles greater than 3 seconds, which shares the same idea of avoiding a conflict with a post encroachment time (PET) greater 3 seconds. The last constraint corresponds to the vehicle's acceleration and deceleration ability. Sequential Least Squares Programming (SLSQP) algorithm is used to find the optimal solution, and the calculation is done by using the Python Scipy library.

3.2.3 Task 3

The goal of task 3 is to train a decision-making model for CAVs to drive cooperatively so that freeway exiting vehicles can diverge to the off-ramp successfully and cause minimum turbulence to the traffic flow. To achieve this goal, a multi-agent reinforcement learning approach is introduced in this section.

For a conventional reinforcement learning setting, at each timestep t, the agent observes the state s_t, and selects an action a_t. Then the environment advances to the next state s_{t+1}, and the agent receives a reward r from the environment. The goal is to learn an optimal policy π^* that helps the agent to make the best decision based on the current state, which maximizes the accumulated reward $R_t = \sum_{k=0}^{T} \gamma^k r_{t+k}$. r_{t+k} is the reward at timestep $t + k$ and $\gamma \in (0, 1]$ is the discount factor, which quantifies the importance of future rewards.

The state-action value function (also called *Q*-function) under policy π^*, denoted by $Q_\pi(s_t, a_t)$ is the expected return at state s_t with action a_t given that making decision

following policy π^* afterwards. The optimal Q-function can be characterized by the Bellman equation:

$$Q_\pi(s_t, a_t) = E[\sum_{t\geq 0} \gamma^t r(s_t, a_t, s_{t+1}) | s_t = s, a_t = t] \quad (3.15)$$

The state value function (expressed as V^*) is the expected return at state s_t considering all available actions at the current step and then follow the optimal policy π^*. The state value function is calculated by:

$$V_\pi(s_t) = E[\sum_{t\geq 0} \gamma^t r(s_t, a_t, s_{t+1}) | s_t = s] \quad (3.16)$$

The action space A_i of agent i is defined as the set of five high-level control decisions: (1) change lane to the left, (2) change lane to the right, (3) add left side virtual leader, (4) add right side virtual leader, (5) no action. The action of adding a left or right-side virtual leader is designed to assist the leader in the adjacent lane to change lanes in the condition that gap between the left or right-side leader and the ego vehicle is not sufficient. Once the action is taken by an agent, it will regard the left or right leader as its actual leader and change its speed according to the car following model. For instance, as shown in Figure 3.9, vehicle 2 cannot change lanes to the right because of an insufficient gap to the right follower. By adding vehicle 2 as the virtual leader, vehicle 1 will back off and secure a safe gap to the virtual leader. The overall action space of the system is the joint actions from n vehicles are $A = A_1 \times A_2 \times \cdots \times A_n$.

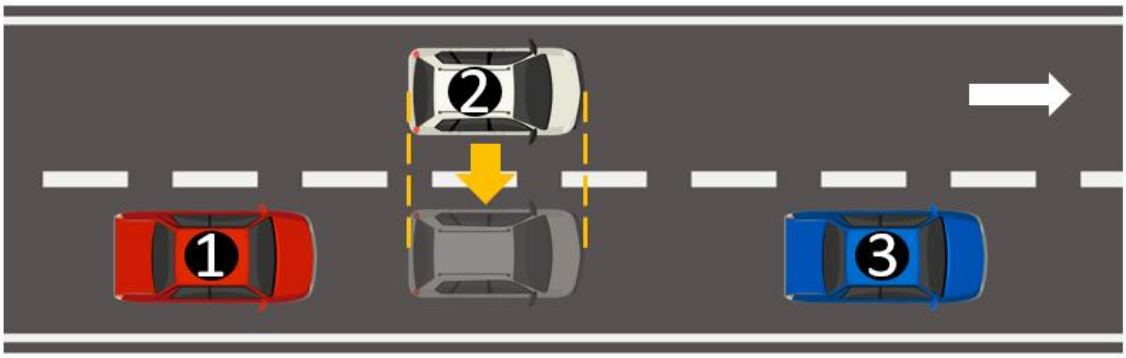


Figure 3.9 Adding a virtual leader

The state of agent i, s_i, is defined as a matrix of dimension $N \times W$, where N is the number of vehicles being controlled and W is the number of features used to represent the state of a vehicle. For each agent, the state consists of two parts: the features of the ego vehicle and the features of the closet neighbors. For ego states S_{ego}, a set of features are selected as shown below.

$$S_{ego} = \{isDiverge, l, d, v\} \tag{3.17}$$

where $isDiverge$ indicates whether the ego vehicle is diverging to the off-ramp or not, l is the lane id that starts from the rightmost lane as 0, d is the remaining distance to the diverging point of the ramp, and v is the current speed of the ego vehicle.

An agent's neighbor state includes the relative speed and relative position of the closet leaders and followers in the current lane or adjacent lanes, which is expressed by:

$$\begin{aligned} &S_{neighbor} = \{isPresent_i, t_i\}, i \in V; \\ &V \in \{leader,\ follower,\ left\,leader,\ left\ follower,\ right\,leader, right\ follower\} \end{aligned} \tag{3.18}$$

where $isPresent$ represents if the current or adjacent leader or follower exits, 1 is exits and 0 is not exits; t_i is the time headway of the current or adjacent leader or follower. The state of an agent is the union of its ego state and neighbor state, $S_{agent} = S_{ego} \cup S_{neighbor}$.

The reward function is crucial to train the agents so that it follows desired behaviors. The total reward at each step consists of three components: speed difference reward $r_{\Delta v}$, time headway rewards $r_{headway}$, and diverging reward $r_{diverge}$. The speed difference reward is designed to reflect the traffic turbulence by comparing the agent's actual speed and desired speed. For each agent, its desired speed is assigned to a fixed value with random noise added and the desired speed does not change during each episode. $r_{\Delta v}$ is expressed as:

$$r_{\Delta v} = |v - v_{desired}| \tag{3.19}$$

where v is the current speed and $v_{desired}$ is the desired speed of the agent. The time headway reward penalizes a car-following behavior with a small time headway to the leader, and it prevents aggressive lane changing behavior cut into an insufficient gap. The calculation of $r_{headway}$ is expressed as:

$$r_{headway} = \log(\frac{t_i}{t_{base}}) \tag{3.20}$$

where $t_{headway}$ is the current time headway to its leading vehicle. $r_{headway}$is set to 0 if there is no leading vehicle. t_{base} refers to the desired time headway, which indicates the minimum time headway to keep safety following maneuver. The value of t_{base} is set to 1.2 second that align with Sumo default.

Furthermore, if the diverging vehicle is traveling on the rightmost lane, a reward will be given as expressed below:

$$r_{diverge} = w\left(\frac{d-L}{L}\right)^2 \tag{3.21}$$

Where d is the remaining distance to the diverging point, L=500 m is the upstream length of the diverging point, and w=10 is the coefficient.

The total reward of agent i R at time step t is the sum of the three types of rewards as follows:

$$r_{i,t} = w_1 r_{\Delta t} + w_2 r_{headway} + w_3 r_{diverge} \tag{3.22}$$

where $w_i, i \in \{1,2,3\}$ are the weights of the rewards. The global reward R at time step t is expressed as the average reward among all the vehicles in the environment, which can be expressed by:

$$R_t = \frac{1}{n}\sum_{i=1}^{n} r_{i,t} \tag{3.23}$$

where n is the number of the vehicles.

In this research, the cooperative setting is adopted for the multi-agent reinforcement learning problem that the agents collaborate to optimize a common long-term return. All the agents share the same reward function and Q-function, which enables the single agent RL to be applied, if all agents are coordinated as one decision maker. Deep Q Network (DQN) is adopted as the training algorithm, where the actor network and critic network are approximated by two neural networks with the same structure and parameters. The neural network uses three fully connected layers with 128 neurons, taking the global state as input and outputs state-value function. For the actor network, the state-value function pass through the Softmax activator and generates the probability of taking each action.

To prevent choosing invalid actions, action masking is applied. There are two types of actions that are considered invalid: unsafe lane change and secondary lane change. An unsafe lane change is defined as a vehicle taking the action of lane change when it still overlaps in the longitudinal direction with vehicles in the adjacent lane. Secondary lane change masking prevents an agent from choosing the action of lane change when the vehicle is currently conducting a lane change maneuver. The action masking works by setting the Softmax output probability of the invalid action to 0 and thus the agent can only choose the actions from the available action list. Afterwards, the vehicle that is closer to the off-ramp is considered with higher priority, and it takes an action and executes first and then advances to the next agent with less priority.

3.3 HMI Design

The speed guidance generated by the CDS algorithm is to be broadcasted to the CV drivers, and the drivers adjust their driving speed accordingly. In a sophisticated driving environment where CV and HDV are mixed, there are great chances that unexpected behavior of HDV occurs, which could bring significant turbulence to the traffic and impact

the optimal speed calculated by the CDS. In such a case, the recommended speed broadcast to the driver may fluctuate over time, and it cause troubles for the CV drivers to follow the suggested speed. Hence, appropriate in-vehicle HMI should be designed that do not only provide CV drivers clear information but also cause minimum cognitive difficulty of obtaining the information to enhance their performance.

In order to achieve this goal, three types of in-vehicle HMI are designed for the CDS algorithm in section 3.2.1, which are (1) Δv interface; (2) Δt interface; and (3) Δv graphic interface. Δv is the difference between the vehicle's optimal speed and CDS current speed while Δt is the difference between the estimated arrival time to the CDS control line and the optimal arrival time that is shown below:

$$\Delta t = \frac{d}{v + \Delta v} - \frac{d}{v} \tag{3.24}$$

where d is the distance to the CDS control line and v is the vehicle's current speed. A negative Δv indicates the driver needs to slow down while a negative Δt represents a speed up command.

The Δv or Δt display method is designed to deliver the exact value of the time or speed difference calculated by the CDS with a brief literal explanation. The Δv or Δt value will be displayed on the vehicle's head-up display system (HUD), with the font color of red indicating slow down and green representing speed up. The value of Δv and Δt is rounded to prevent fractions to be displayed which brings extra reading load. For Δv display, the driver speed guidance display is shown below:

- "Speed up Δv mph": when Δv>0 and the driver need to speed up;
- "Slow down $|\Delta v|$ mph": when Δv<0 and the driver need to slow down;
- "Keep speed": when Δv=0 and no need to change speed;
- No display: when the CDS failed to find an optimal solution in a sophisticated environment.

For Δt display, the speed guidance display is expressed as:

- "Δt s ahead": when Δt >0 and the driver need to slow down;
- "Δt s behind": when Δt <0 and the driver need to speed up;
- "Keep speed": when Δt=0 and no need to change speed;
- No display: when the CDS failed to find an optimal solution in a sophisticated environment.

In addition to the Δv and Δt display method, a Δv graphic display interface is also introduced. It converts the speed difference Δv to a real-time updating graphic navigation panel that is displayed on the HUD. The interface consists of a vehicle icon and a baseline map, where multiple lines with different colors are drawn on the baseline map as shown in Figure 3.10. The central green line is the baseline that indicates the optimal state (Δv=0), and the yellow lines and red lines imply the vehicle icon around these lines needs to adjust its speed. The space between two consecutive lines represents a speed difference of 1m/s, which means the top or bottom yellow lines indicates $\Delta v = \pm 5$ m/s. The top and bottom red lines are the boundaries that the vehicle icon will never move beyond this region. Given that Δv is updating step by step following the CDS, the position of the vehicle icon moves in a real-time manner. When Δv<0, the vehicle icon moves up from the center line, and the vehicle icon goes down when Δv>0. The drivers can obtain the key information of whether to acceleration or deceleration maneuver action from the guidance display, which is expected to alleviate the CV drivers' load of reading the numbers. The three display methods are summarized in Table 3.3.

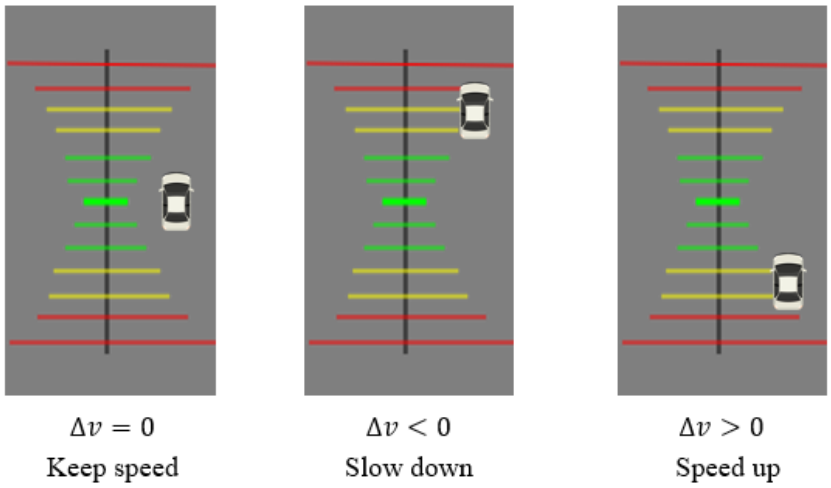
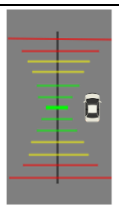
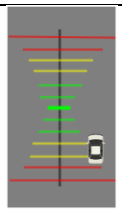
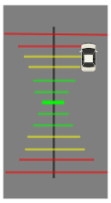


Figure 3.10 Δv graphic display

Table 3.3 Summary of proposed in-vehicle guidance interface

	Keep speed (Δv=0)	**Speed up (Δv >0)**	**Slow down (Δv <0)**	**No solution for CDS**
Δv	Keep Speed	Slow down 5 mph	Speed up 3 mph	/
Δt	Keep Speed	5 s ahead	3 s behind	/
Δv-graphic				/

4 Experiment Design

4.1 Task 1

To investigate the effectiveness of the CDS in a mixed traffic state, driving simulator and microscopic traffic simulation are designed. There are totally five layers of experiments, which correspond to different traffic mixture state, shown in Table 4.1.

Table 4.1 Experiment design

Layer index	Layer type	Experiment type	Simulation software	Vehicle involved
1	100% HDV	Microscopic traffic simulation	SUMO	HDV (SUMO generated)
2	HDV-CV mixture	Microscopic traffic simulation + multi-driver driving simulator	SUMO + CARLA	HDV (SUMO generated) CV (driven by participants)
3	100% CV	Multi-driver driving simulation	CARLA	CV (driven by participants)
4	CV-CAV mixture	Microscopic traffic simulation + multi-driver driving simulator	SUMO+ CARLA	CV (driven by participants) CAV (SUMO generated)
5	100% CAV	Microscopic traffic simulation	SUMO	CAV (SUMO generated)

SUMO was used to generate HDV and CAV, while CARLA was used as the testbed for CV. SUMO and CARLA co-simulated layer 2 and layer 4, where HDV/CAV was controlled by SUMO, and CV was driven by the participants with cooperative driving instruction provided to them. Both SUMO and CARLA collect simulation data and generate output evaluation files that were used for the result analysis.

For each SUMO simulation experiment, the scenario was simulated in the calibrated model. A total duration of 3600 seconds was set for a single simulation, and it was simulated 5 times with different random seeds. Evaluation files including vehicle trip information, aggregated travel time for roads, and surrogate safety measures were generated.

To conduct the driving simulation experiment for task 1, we recruited 12 drivers. IRB approval has been obtained. The drivers were divided into 4 groups with 3 participants evenly distributed to each group. The simulation was done group by group. That is, 3 drivers in the group drove in the CARLA simulator simultaneously by connecting all 3

drivers to the simulation server. The drivers were asked to follow pre-set paths so that they can drive around on the map and pass the intersection multiple times to produce sufficient samples. The paths of a single loop designed for each driver are demonstrated in Table 4.2. Also, the type of vehicle interaction of CV is shown in the table. The naming rule of paths is clarified in the microscopic traffic simulation sub-section in section 3.1.1, and also the paths are visualized in Figure 3.4.

Table 4.2 Path arrangement

Path index	Driver 1	Driver 2	Driver 3	Expected Interaction of CV
1	r32	r52	r14	crossing, merging
2	r16	r14	r32	following
3	r54	r32	r16	no interaction
4	r36	r16	r54	merging
5	r52	r54	r36	crossing, following
6	r14	r36	r52	crossing

As shown in Table 4.2, a single loop contains 6 different paths, which covers all the possible origin-destination combinations for a 3-way intersection. The drivers may encounter different types of interactions with another driver, including crossing, merging and vehicle following. For a single experiment, the 3 drivers in the group will complete 3 loops of paths. Hence, for each experiment layer, there are 4 groups with 3 drivers who will complete 3 loops of driving that contain 6 paths. Theoretically, 4*3*3*6=216 vehicle passing intersection events will be collected for each layer. All three layers with driving simulation (layers 2,3,4) have the same abovementioned experiment design, but are different in the traffic mixture state (HDV, CV, CAV shown in Table 4.1).

4.2 Task 2

To test the effects of different in-vehicle guidance interface, another set of driving simulator experiments are designed. The experiments share the same simulation

environment as in task 1 with identical driver and vehicle model settings in Sumo and Carla, but are different in experiment design.

The experiment is a within-subjects experiment. Each scenario has three types of interface design; each participant driver experienced all designs in all scenarios in a randomized order. The advantage of a within-subjects experiment is that it controls extraneous participant variables and makes it easier to detect the relationships between the independent and dependent variables. There are two layers of experiments in total: mixed traffic environment and CV-only environment, which corresponds to the second and third layer in Table 4.1.

In the mixed traffic layer, when the CV enters the CDS control zone, a HDV will be spawned simultaneously at the other two approaches of the intersection, as shown in Figure 4.1. The HDVs are spawned with the same initial speed as the CV's speed, and the spawn point is located at the CDS control boundary (150 meters from the intersection stop line) with a random integer ranging from [-8,8] meters. Hence, the distance to the intersection is in the range of [142, 158] meters when the HDVs are spawned. With the same initial speed as the CV, the HDVs are expected to arrive at the intersection around the same time as the CV if the CV maintains its speed. However, the CDS will broadcast the suggested speed to the CV drivers and guide them to avoid conflict with HDV. The effects of the different speed guidance interfaces will be explored in the experiments.

There are 20 participants recruited in total for this experiment, among which 7 are females and 13 are males with an average age of 31.1 years old and average driving experience of 9.2 years. Each participant held a valid driving license. Upon arriving at the driving simulator lab, each participant completed both a consent form and a demographic survey. Afterwards, they were instructed about the possible speed guidance interface designs they might meet during the experiment. Before the formal

experiment, each participant was given a practice drive to get used to the simulator and the different interfaces. In the formal experiment, each participant completed the task of driving with 3 types of CDS speed guidance, and within each guidance type, the participants drove through the intersection 12 times.

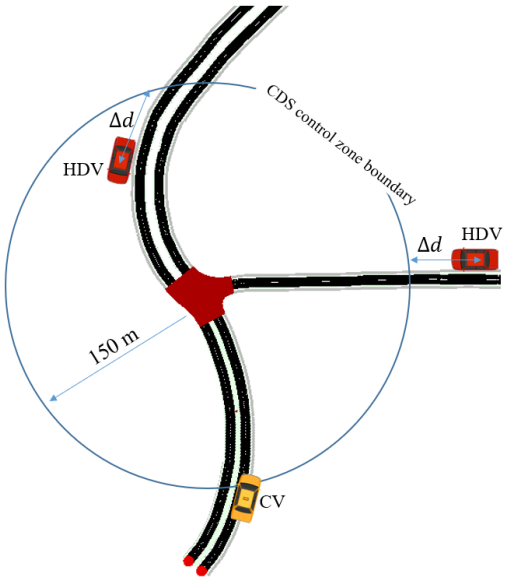


Figure 4.1 Mixed traffic environment

4.3 Task 3

To train and test the MARL algorithm for off-ramp cooperative driving, a simulation environment based on Sumo is developed. An off-ramp road network is built in Sumo as shown in Figure 4.2. The total length of the road network is 1 km, where the diverging point of the off-ramp is at the center of the road network, and the downstream and upstream are both 500 m. The main line has 2 lanes and the off-ramp has a single lane. A vehicle group of 6 vehicles is simulated, among which 2 are diverging vehicles (the first 2 vehicles on the inner lane) and the rest are through vehicles, as shown in Figure 4.2. The spawn points of the vehicles are shown in Figure 4.3. Spawn point 1 and 4 are at the start of the road network, spawn point 2 and 5 are 20 m away in downstream, and

spawn points 3 and 6 are 40 m away from spawn point 1 and 4. Diverging vehicles (red vehicles) are spawned at spawned point 2 and 3, and the rest (white vehicles for HDVs) are through vehicles. When the simulation is initialized, 6 vehicles are spawned simultaneously at the position of Δt m to the spawn point where Δt is a random integer ranging from -5 to 5. Each vehicle is given a desired speed following a normal distribution that N~(24, 2.4) in m/s. The vehicle following behavior is controlled by the ACC model embedded in Sumo, and lane change maneuver is executed using the lane change command in Sumo through TraCI portal.

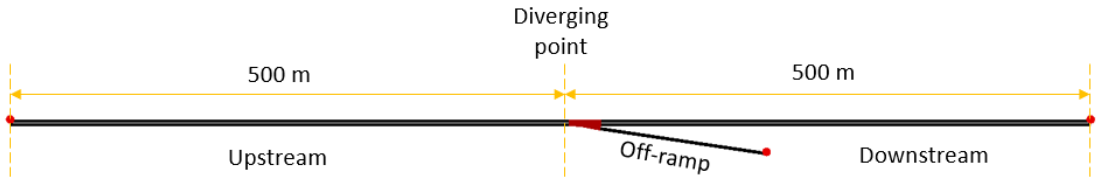


Figure 4.2 Off-ramp simulation model in Sumo

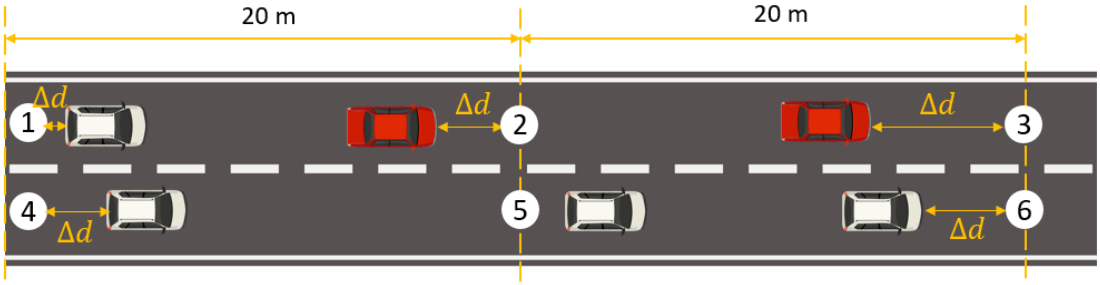


Figure 4.3 Vehicle spawn point

The model is trained for 10,000 episodes, with a discount factor γ=0.99 and learning rate η=e^{-4}. The coefficient w_1,w_2, and w_3 for the reward function are all set to 1. In each episode, the simulation ends when all the vehicles have passed the off-ramp (distance to ramp d_i<0, $i \in V$) or a collision happens. Only the vehicle in the control zone (0<d_i<500) will be coordinated by the algorithm, vehicles passed or diverged the off-ramp maintains

their driving pattern. The simulation is performed in a Windows 10 Desktop with a 2080 super graphic card and 64G RAM.

5 Experiment Results and Discussion

5.1 Task 1

Five layers of traffic simulation and driving simulator experiments are conducted and the vehicle data and driving behavior data are collected. In order to understand the effectiveness of CDS in terms of traffic throughput efficiency and safety, multiple measurements are used including travel time, average speed, throttle usage, and surrogate safety measures. The results are demonstrated in Figures 5.1 to Figure 5.7, and Tables 5.1 to 5.3. The measure “travel time” is the time vehicle spends from entering the control zone to entering the intersection. The interval for average speed, throttle usage, and jerk calculation (deviation of acceleration) is also from the boundary of the control zone to the entering point of intersection. It has to be mentioned, in Table 5.1 and Table 5.2, the travel time for 100% HDV and 100% CAV layer refer to the travel time of HDV and CAV, respectively, while in other layers it refers to the travel time of CV.

The market penetration rate (MPR) of CV is expected to affect the performance of CDS, thus for a mixture traffic scenario, the CV MPR is divided into 4 intervals [0-25%], [25-50%], [50-75%], and [75-100%] to explore its influence on traffic efficiency and driving comfort. The calculation of CV MPR is the number of CV divided by the number of all the vehicles inside the control zone. In addition, in a CV-HDV mixture environment, CV may encounter congestion when approaching the intersection. Congestion, in this research, refers to two situations: (1) HDV is stopping at the stop sign when CV approaches the intersection in a different entrance; (2) slow-driving HDV is leading the CV at the same lane, and the CV is not allowed to pass the leading HDV inside the control zone. According

to observation, these two situations affect the drivers' perception and judgment and slow down the CV. Hence, travel time and average speed in congested and non-congested scenarios are calculated and presented for HDV-CV mixture layer.

Table 5.1 Travel time summary (vehicle type specified)

Layer	**CV MPR %**	**Mean (s)**	**Min (s)**	**Max (s)**	**Std**	**count**	***t* value**
100 % HDV (baseline)		41.4	16	64	6.14	606	
CV-HDV (CV MPR %)	0-25	35.3	10.3	72.6	21.82	20	3.75 (vs 100 % HDV)
	25-50	30.3	8.5	67.5	16.94	100	1.14 (vs 0-25 % MPR)
	50-75	20.9	7.5	29.2	12.49	52	3.53 (vs 25-50 % MPR)
	75-100	13.5	10.4	18.6	2.71	14	2.19 (vs 50-75 % MPR)
100 % CV		12.1	7.3	23.4	2.70	179	1.86 (vs 75-100 % MPR)
CV-CAV (CV MPR %)	0-25	15.3	10.3	21.3	3.00	12	
	25-50	16.9	7.3	28.5	5.52	70	0.97 (vs 0-25 % MPR)
	50-75	14.6	8.6	27.0	3.84	83	3.02 (vs 20-50 % MPR)
	75-100	14.1	8.0	20.1	3.41	35	0.67 (vs 50-75 % MPR)
100 % CAV		10.9	9.6	12.8	0.59	609	10.2 (vs 100% CV)

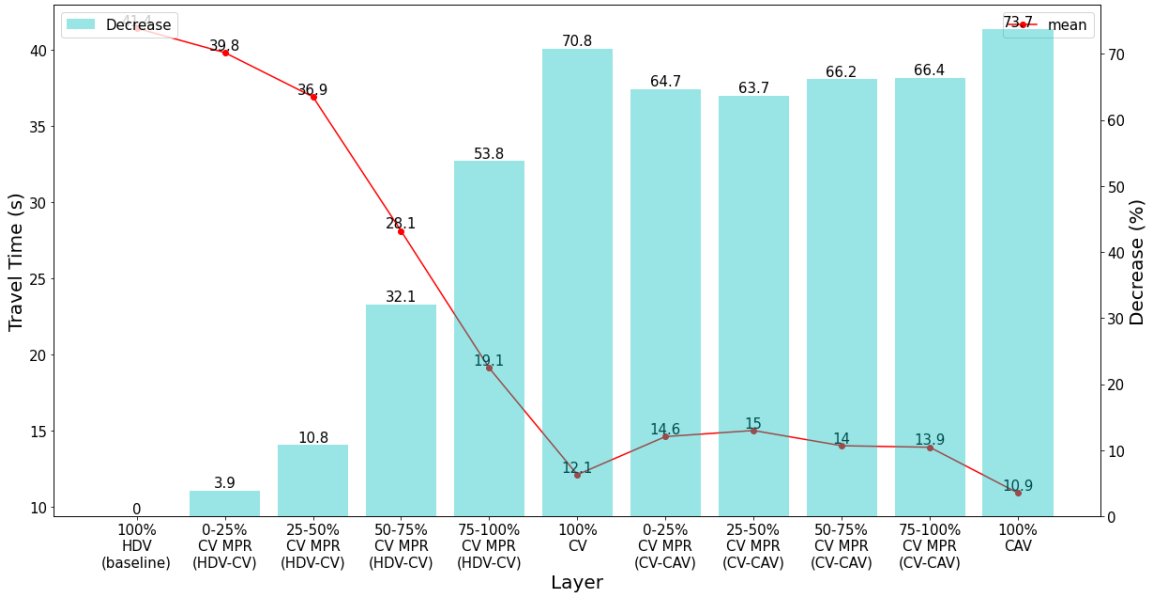


Figure 5.1 Travel time (average travel time for mixed traffic)

Table 5.2 Average speed summary

Layer	CV MPR %	Mean (kph)	Min (kph)	Max (kph)	Std	count
100 % HDV (baseline)		13.1	8.4	33.8	2.62	606
CV-HDV (CV MPR %)	0-25	15.3	7.4	56.4	9.84	20
	25-50	17.8	8.0	63.5	8.52	100
	50-75	25.8	9.1	72	6.84	52
	75-100	40	29	51.9	6.41	14
100 % CV		44.6	23	73.9	4.00	179
CV-CAV (CV MPR %)	0-25	35.3	25.3	52.4	4.24	12
	25-50	32	18.9	73.9	5.69	70
	50-75	36.9	20	62.8	4.30	83
	75-100	38.3	26.8	77.1	4.68	35
100 % CAV		49.5	42.1	56.5	1.23	609

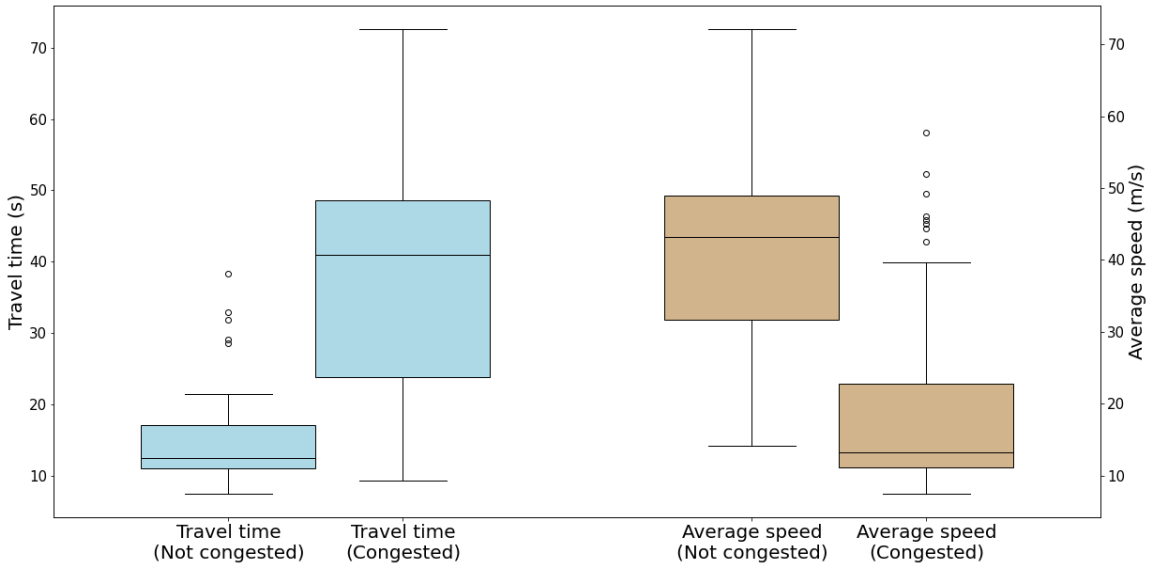


Figure 5.2 Travel time and average speed in congested/uncongested state

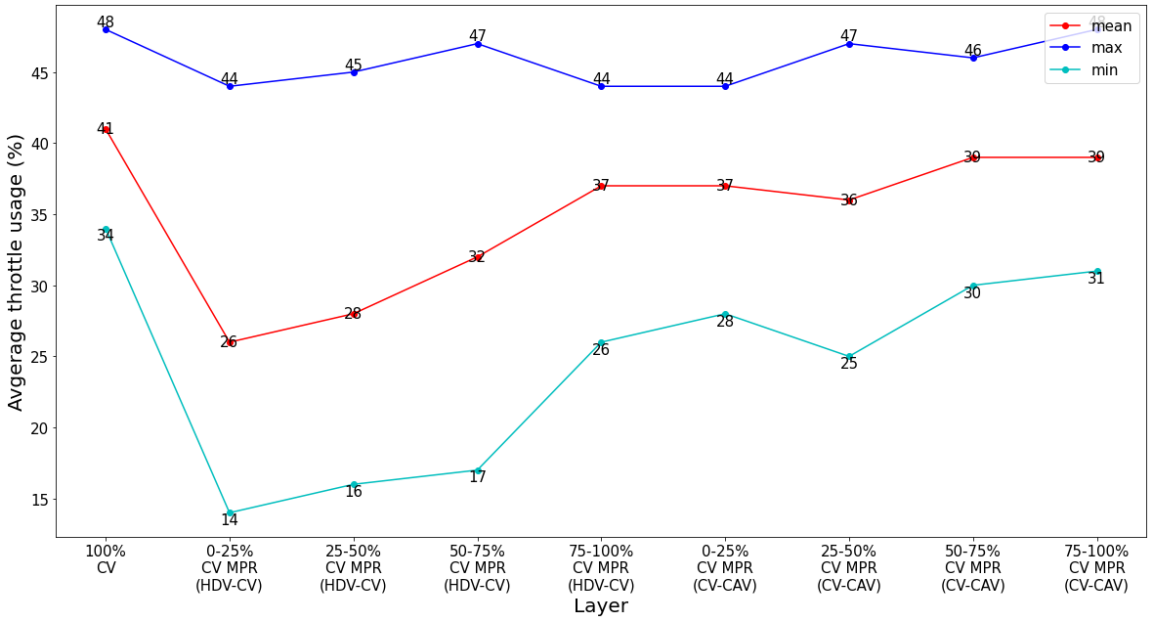


Figure 5.3 Average throttle usage

5.1.1 Efficiency

The travel time is the direct indicator of traffic efficiency. As Table 5.1 demonstrates, compared with the base layer (100% HDV), the travel times of the layers applied CDS have all decreased to a certain extent. For CV-HDV mixture layer, the travel time saving

of CV increases as the MPR increases, and the increases from lower MPR interval to higher MPR interval are all significant ($p<0.05$) except for MPR from [0-25%] to [25-50%]. This shows that MPR of CV has a great positive impact on the traffic efficiency in a CV-HDV mixed traffic. This is because in a relatively low CV MPR driving environment, CV are more likely to be held by the HDV, either being blocked by the HDV in front or need to yield to HDV at the intersection, which is the state of the previously mentioned term "congestion". As Figure. 5.2 shows, when CV that runs into congestion averagely spends 192% more travel time (13.6 to 39.8), which validates the abovementioned finding. Once the CV MPR increase to 75-100%, the travel has seen a dramatic decrease to 13.5 s, which is about the same level as the CV-CAV mixture scenario. Also, the travel time of 100% CV layer (layer 3) is smaller than layer 2 and layer 4. We presume the reason for it is there are only 3 CVs that are driven by the participants in the experiment, and unlike layer 2 and layer 4, they are not affected by the surrounding traffic (HDV or CAV), and they may be prone to drive faster and gain a smaller travel time. As expected, the efficiency of the 100% CAV scenario is the best, for CAVs' average travel time (10.9) is significantly smaller than all other layers (t=10.2, d.f.>1000, vs 100% CV).

Figure. 5.1 shows the reduction in travel time in terms of traffic flow perspective after apply CDS. In alignment with previous findings, in a low CV MPR rate when mixed with HDV, the efficiency improvement is negligible, for only 3.9% of travel time saving is gained. However, when the CV MPR is sufficient or in a CV-CAV mixed traffic, 60-70% of saving in travel time can be acquired. Furthermore, in a fully CAV scenario, up to 73.7% travel time saving is expected, which are on the same level of previously developed CDS in terms of efficiency improvement [31].

Similar to travel time, the average speed in the control zone also indicates traffic efficiency performance. Although the calibrated value for HDV speed is around 36 kph (10 m/s), the average speed of HDV in the control zone is only 13.1 kph (Table 6.2). This is

because HDV have to slow down and stop at the stop sign, and it brought down the average speed dramatically. It should be noticed that even if the average speed is only 15.3 kph in 0-25% CV MPR scenario, the maximum average speed still reaches 56.4 kph, which is much higher than the average value (the outliers in Figure. 5.2). This could be explained by that the CV did not encounter any traffic or the CDS still finds an optimal pass schedule for the CV in a congested situation, and thus the CV can drive at a high speed as the CDS instructed.

In addition, to understand the speed distribution inside the control zones, we divide the control zone into inner zone and outer zone, as shown in Figure. 5.4. Inner zone is the 50 m area around the intersection, while the outer zone is the remaining part of the control zone. The speed at inner zone for 0-25%, 25-50%, 50-75%, and 75-100% CV MPR are 8.2 m/s, 13.6 m/s, 16.7 m/s, and 29.9 m/s with standard deviation of 24.8, 21.3, 18.2, and 15.3 respectively. This shows that CV speed at the inner zone is very low and is distributed arbitrarily. On the other hand, the average speed at outer zone for the 4 MPR are 28.3 m/s, 30.0 m/s, 37.9 m/s, 46.4 m/s, with standard deviation of 4.2, 5.1, 4.8, 5.0, respectively. It indicates that CV drivers are driving much faster in the outer zone than in the inner zone. Also, their driving in the outer zone is much steadier and thus the speed is easier to estimate.

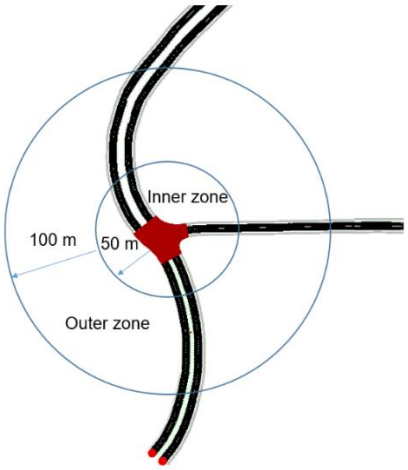


Figure 5.4 Control zone divide

The average throttle usage reflects how much throttle the driver is using under the CDS. It can be inferred from Figure. 5.2 that with a higher CV MPR, the drivers are prone to use more throttle, and therefore their speed should be higher. The results of throttle usage could be used in emission estimation in future research.

To better understand the contributing factors for CDS performance on efficiency, a linear model is estimated for travel time and average speed considering 5 factors: CV MPR (range from 0-1), congestion (1 if yes), go straight at intersection (1 if yes), right turn at intersection (1 if yes), and left turn at intersection (1 if yes). The model estimation results are shown in Table 5.3.

Table 5.3 Model for travel time

	Travel time		**Average speed**	
Variable	Coefficient	p value	Coefficient	p value
Intercept	27.17	<0.001	30.75	<0.001
MPR	-22.77	<0.001	15.79	<0.001
Congestion	21.56	<0.001	-21.19	<0.001
Right turn	-1.64	0.544	3.71	0.110
Left turn	-1.33	0.570	0.31	0.891
Go straight	-3.25	0.177	4.25	0.101

From the modeling results, it can be inferred that CV MPR and congestion have a significant impact on travel time and average speed for both *p* values are smaller than 0.001. Also, going straight at the intersection has a positive effect on time saving and speed increase compared to turning at intersection, which align with common sense.

5.1.2 Driving Comfort

The standard deviation (std) of throttle reflects driving comfort and is also a good indicator of how easy it is to follow the CDS. Smaller std means the drivers are steadier on the throttle and can follow the CDS better. Unsurprisingly, for CV-HDV layer, lower CV MPR has larger throttle std as Figure. 5.5 shows. This means drivers are busy adjusting their throttle usage, and it shows the CDS instruction is not easy to follow. In this case, the driving comfort deteriorated. As MPR increases, throttle std decreases as expected, but still larger 100% CV layer.

In CV-CAV layer, the throttle std is larger than 100% CV, which is not expected. Because CAV will adjust their speed in time to let CV drive more comfortably, and thus CV does not have to change the throttle frequently. However, the opposite trend is gained. The possible reason for this might be: when the driver is driving at the CDS suggested speed, no driving instruction is displayed; thus the driver relaxed and did not hold the throttle steadily.

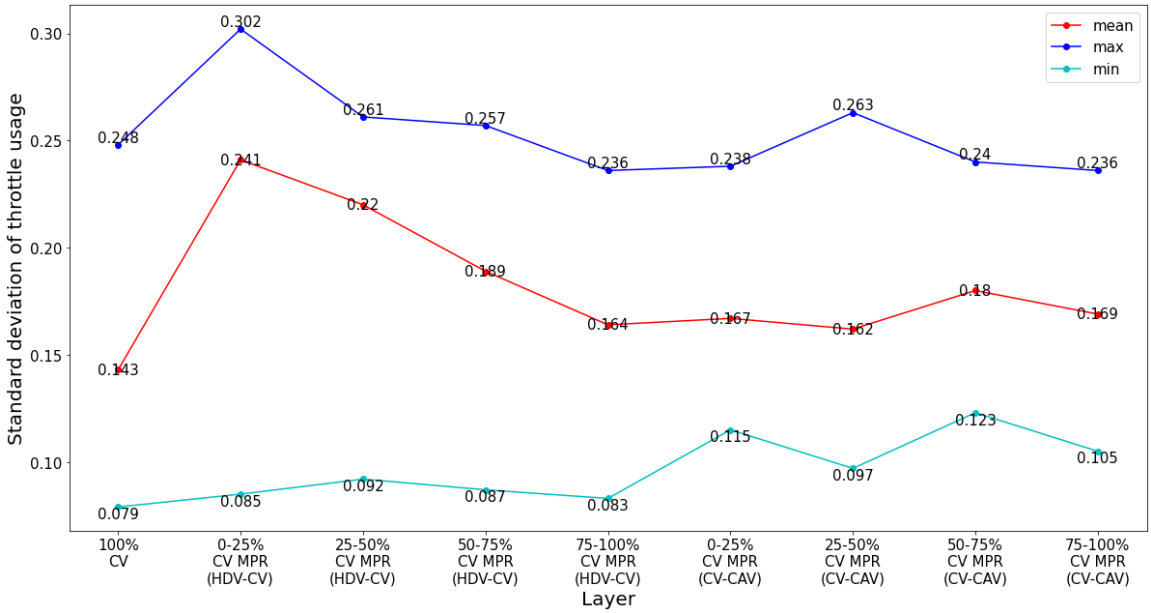


Figure 5.5 Standard deviation of throttle usage

To study driving comfort, jerk, which is the third derivative of vehicle of position with respect to time, has been widely used. Driving style with excessive jerk easily makes the passengers feel discomfort. For a highly comfortable ride, the maximum allowable jerk experienced by the passengers is suggested as 18 m/s^{-3} [32]. Figure. 5.6 shows the maximum longitudinal jerk of the vehicles inside the control zone of different layers. From the results, we could infer that the drivers are driving a bit aggressively, as the average longitudinal jerk for all scenes have exceeded 40 m/s^{-3}, and therefore the comfort of the passengers deteriorated. This might relate to the way that the driving instruction is provided to the drivers, and it may need to optimize the instruction message and GUI displayed for drivers. Nevertheless, for most of the layers, there are trips with small jerks, as the minimum values are smaller than 20 m/s^{-3}, and in these cases, the CDS is working perfectly and the drivers are closely following the instructions. It could be also noticed that the difference of driving comfort between 100% CV MPR and a low CV MPR (CV-HDV mixture) is significant (t=9.82, d.f.=189), which again shows that in a high CV MPR environment the CDS performs better in terms of driving comfort.

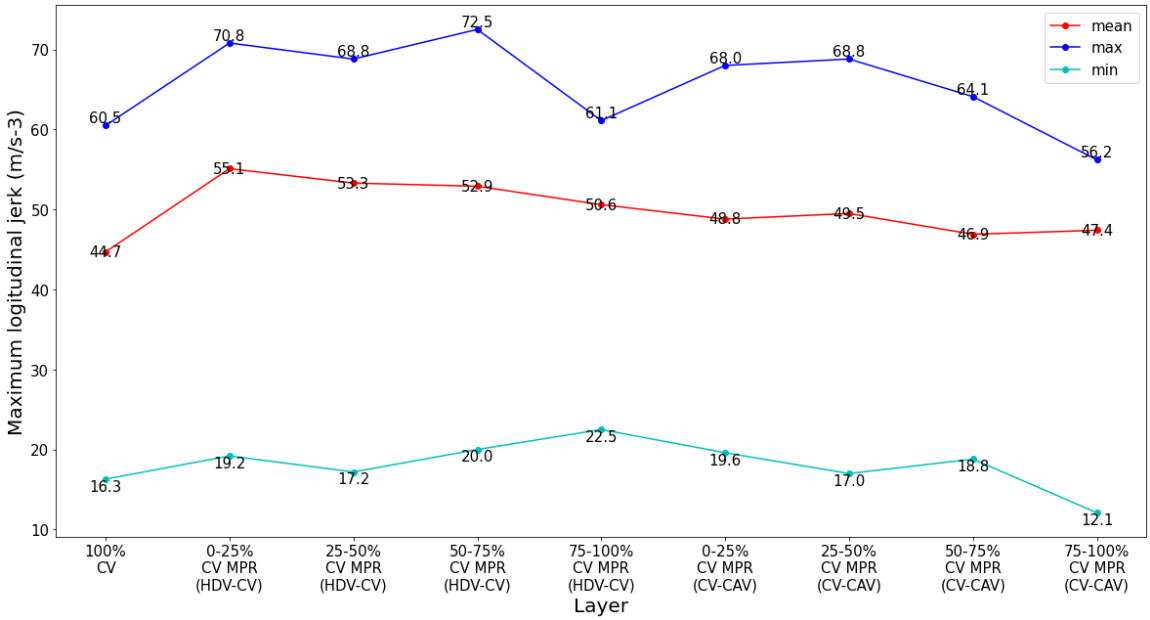


Figure 5.6 Maximum longitudinal jerk

5.1.3 Traffic Safety

Different from most previous research of intersection cooperative driving, we not only focused on efficiency but considered safety. The results of 3 surrogate safety measures (TTC, PET, DRAC) and 4 conflict types (rear-end, crossing, merging, and crash) are shown in Figure. 5.7. Time-to-collision (TTC) and Post-encroachment-time (PET) are surrogate safety measures widely used for safety analysis. DRAC, the deceleration rate to avoid a crash, represents the count of hard brakes. The threshold of TTC and PET is set to 1.5 seconds, which is the most adopted threshold. TTC or PET smaller than the threshold is regarded as a conflict. Also, DRAC larger than 3 m/s^2 is regarded as a conflict. The conflict types are automatically generated and recorded by SUMO.

First, the percentage of conflict occurrence dropped after applying the CDS, which indicates the CDS has safety benefits for CV and CAV. The conflicts of CV-CAV layer were reduced dramatically compared with the CV-HDV layer and baseline for all types of measurement. This shows that the implementation of CAV technology can substantially improve traffic safety. Second, the most frequently occurred conflict is rear-end conflict.

This might due to the huge speed difference between CV and HDV near the intersection which leads to a small TTC. To address this issue, slow down messages should be given to the drivers when approaching the intersection if congestion is encountered. Conflict type of crossing is also observed in the experiments, but not by a large percentage, which means the CDS can avoid most of the crossing conflicts. Also, increase the safe time gap could reduce the potential crossing conflicts. The conflict type of merging conflict is rare in the experiment, and it proves that CDS is able to manage vehicles' arrival time in a merging scene.

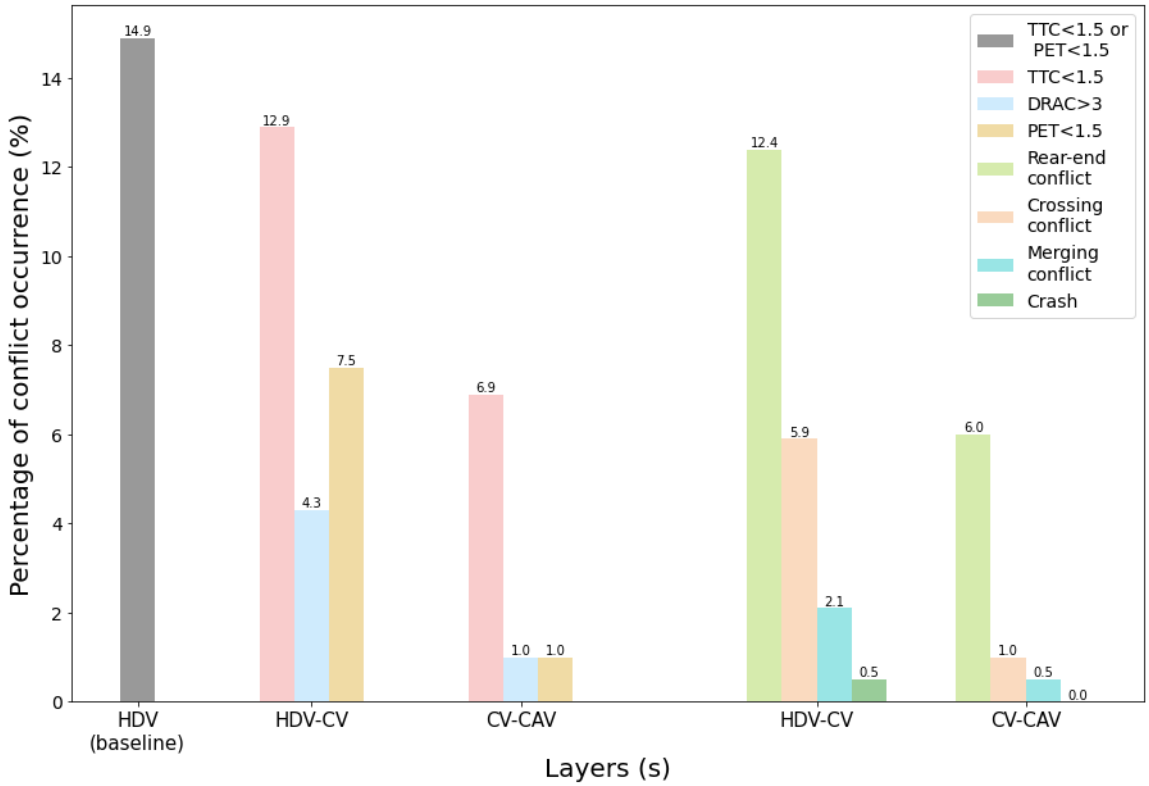


Figure 5.7 Surrogate safety measure (SSM) summary

5.2 Task 2

The objective of task 2 is to design HMI for cooperative driving at non-signalized intersection, and evaluate HMI designs through driving simulator experiments. Multiple vehicle dynamic, driving comfort, and goodness-of-following-guidance indicators are

adopted to measure the performance of different HMIs in various CDS activation conditions.

To define different CDS activation conditions, the following approaches are used. First, based on the driving environment with or without HDV, the traffic condition is divided into two types: the “mixed traffic” and the “CV environment”. Then, the value of Δt (the time difference between optimal and estimated time arrives at the CDS control line) when a CV enters the CDS control zone, denotes as Δt_{ini}, is used to reflect the initial CDS activation state. A Δt_{ini} with greater absolute value indicates a bigger time difference between estimated arrival time and CDS optimal arrival time, which requires the CV driver to change its speed more in order to avoid conflicts. Because big Δt_{ini} needs much speed adaptation and put more driving load on the CV drivers, it is defined as the “hard mode”, and condition with a small Δt_{ini} is defined as “easy mode”. In this research, a threshold of Δt_{ini} of 1 second is adopted to separate two conditions: CDS with Δt_{ini}<1 s is the “easy mode” and CDS with Δt_{ini}>1 s is the “hard mode”. Furthermore, the CDS control zone is divided into two zones based on the distance to the intersection: the “effective zone” and the “steady zone”. The “effective zone” is the area of (100,150] meters from the intersection stop line, where the vehicles entering the CDS control zone and complete the major task of speed changing; the “steady zone” is the area of (30,100] meters from the intersection stop line, where the vehicles’ main task is to drive steadily following the CDS guidance after speed adaptation in the “effective zone”. The data is grouped by the different conditions and the descriptive statistics of vehicles dynamic, driving comfort, and CDS effectiveness measurements are presented below.

5.2.1 Goodness-of-guidance-following

The variable Δt represents the time difference between the estimated and optimal arrival time to the intersection control line for a CV. The average absolute Δt (denote as average $|\Delta t|$) for a segment indicates the overall approximation of a CV's actual speed profile to the optimal speed calculated by the CDS. A smaller average $|\Delta t|$ value suggests the CV driver is following the speed guidance more closely in the segment, which corresponds to better goodness-of-guidance-following. The boxplot of average $|\Delta t|$ at the "effective zone" and the "steady zone" under "easy mode" and "hard mode" is shown in Figure 5.8.

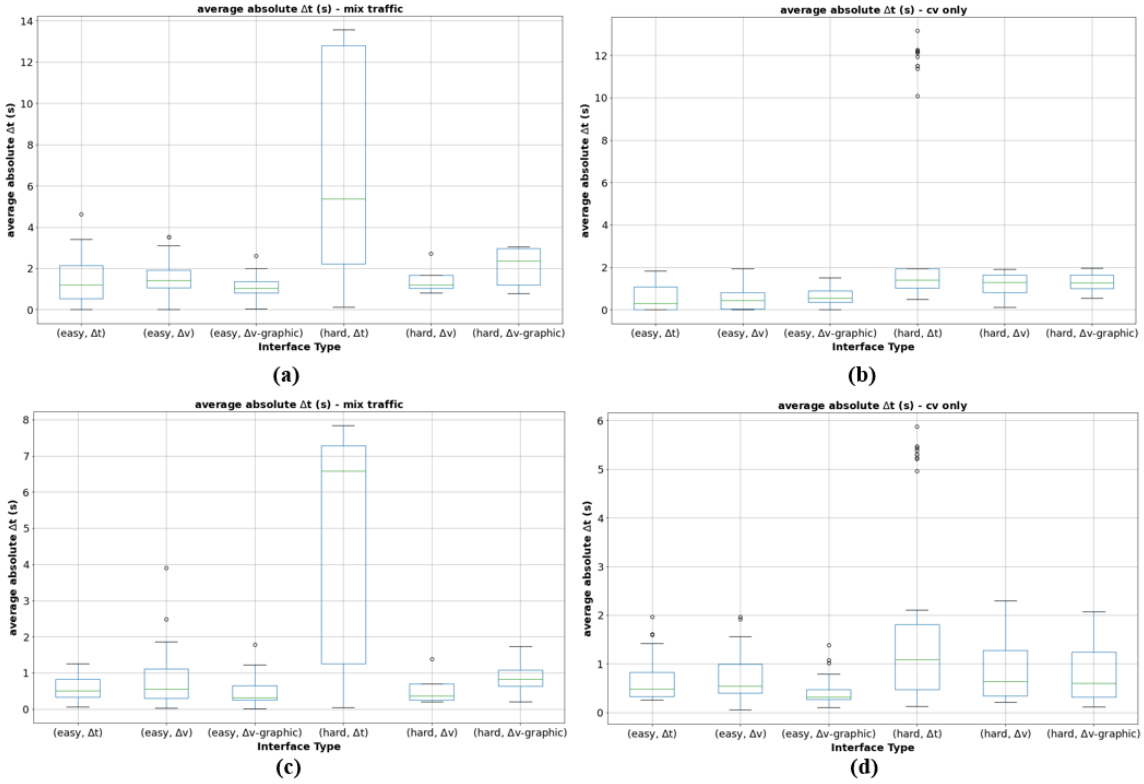


Figure 5.8 Average $|\Delta t|$ under different conditions

Notes: (a) average $|\Delta t|$ at the "effective zone" in mixed traffic; (b) average $|\Delta t|$ at the "effective zone" in CV environment; (c) average $|\Delta t|$ at the "steady zone" in mixed traffic; (d) average $|\Delta t|$ at the "steady zone" in CV environment.

In the “effective zone”, the main task for a CV is to change its speed to approximate the suggested optimal speed, while in the “steady zone” it is important to maintain a steady drive following the speed guidance. Hence, the average $|\Delta t|$ in the “effective zone” (mean=1.58) is significantly greater than average $|\Delta t|$ in the “steady zone” (mean=0.91) as expected ($p<0.001$). The traffic condition also affects the driver’s performance, as a significant difference ($p<0.001$) is observed between “easy mode” (mean=0.90) and “hard mode” (mean=1.92). Furthermore, the drivers are more comfortable following the guidance in a pure CV environment compared to mix traffic, for the mean value of the average $|\Delta t|$ for the two conditions are 1.07 and 1.55 respectively ($p<0.001$).

In the mixed traffic condition, the three in-vehicle HMIs showed different characteristics in terms of CV drivers’ algorithm following behaviors. For “easy mode”, the CV drivers have less pressure to make speed changes, and their primary task is to react to minor changes in optimal speed and try to approximate it. At the “effective zone”, drivers are usually given a relatively larger speed change target, and significant differences of average $|\Delta t|$ are observed for the three HMIs ($F=3.43$, $p=0.035$). The Δv-graphic interface performed best with a mean value of average $|\Delta t|$ of 1.09, then Δt interface with mean value of 1.37 and Δv interface is worst with a mean value of 1.52. The results of average $|\Delta t|$ show similar trend for the “steady zone”, where significant differences between HMIs are captured ($F=5.58$, $p=0.004$). The mean value of the average $|\Delta t|$ for Δt interface, Δv interface, and Δv-graphic are 0.56,0.82,0.46, respectively. The results show that the Δv-graphic interface can perfectly deliver any minor changes in the optimal speed to the CV drivers due to its ability to convert float values into graphic style with great precision. The Δv interface is not good at displaying these small variations and thus causes the drivers to mis operate. For “hard mode”,

significant differences are observed in the average $|\Delta t|$ values between the three interfaces for both the “effective zone” (F=4.03, p=0.034) and the “steady zone” (F=5.85, p=0.014). For both zones, the Δt interface performed worst with a mean value of average $|\Delta t|$ of 7.18 and 4.81 respectively. The Δv outperformed Δv-graphic interface unexpectedly, the potential reason for this is in “hard mode” vehicles are required to accelerate or decelerate much, and the Δv provides more direct instruction of “speed up” or “slow down” that similar to a binary command to the drives, and it might stimulate the drivers to change speed instantly.

In the CV environment where no HDV presents, the CDS does not need to cooperate with the unpredictable behavior of HDV and thus provides steadier speed guidance. At the “effective zone”, the mean values of average $|\Delta t|$ for Δt interface, Δv interface, and Δv-graphic interface are 0.54, 0.55, and 0.62 respectively for the “easy mode”. The differences are not significant (F=0.246, p=0.78), indicating similar performances of the three different HMIs under the specific CDS activation condition. For “hard mode”, the three HMIs performed significantly differently (F=9.53, p<0.0001). The Δt interface is the worst interface, as it does not only have the highest average $|\Delta t|$ value=3.63, but also produces multiple undesired cooperative driving events with average $|\Delta t|$>10. However, there is no significant difference between the Δv interface and the Δv-graphic interface under “hard mode” in a CV environment. At the “steady zone”, a significant performance difference has been observed for both “easy mode” (F=5.13, p=0.007) and “hard mode” (F=8.19, p=0.004). As expected, the Δt is the interface with the highest average $|\Delta t|$ value, which shows the CV drivers are struggling to keep the desired speed suggested by CDS in the “steady zone”. Furthermore, the Δv interface and the Δv-graphic interface show similar performance under “hard mode” (p=0.52), while the Δv-graphic interface is significantly better than Δv interface under “easy mode” (p=0.002).

In general, the Δt interface performed significantly worse than the other two HMIs in terms of goodness-of-guidance-following in most of the CDS activation conditions. The Δv interface and the Δv-graphic interface have their own strength and weakness in different conditions. For cooperative driving that requires much speed adaptation, the Δv interface is better because it presents the drivers with more direct driving command "speed up" or "slow down", which is beneficial for the drivers to make instant operation. However, when the CDS is working in a relatively steady condition that provides drivers with a minor speed change request, the Δv-graphic guidance interface is more suitable, as it converts variations in the optimal speed into graphic information that allow the drivers to capture any small changes more easily.

To further understand the factors that influence CV drivers' performance on following speed guidance and the effects of different HMIs regardless of these factors, the repeated measurement Analysis of Covariance (ANCOVA) model is used, which has the form shown below:

$$y_{ij} = \mu + \tau_i + B(x_{ij} - \bar{x}) + \varepsilon_{ij} \quad (5.1)$$

where y_{ij} is the jth observation under ith categorical group, μ is the grand mean, x_{ij} is the jth observation of the covariance of ith group, $\bar{x}$ is the global mean for the covariance, B is the coefficient, τ_i is the effect of ith categorical variable, and ε_{ij} is the unobserved error term for the jth observation in the ith group. Three variables that affect the average $|\Delta t|$ are included in the model: approaching throttle usage (the throttle usage when entering the CDS control zone), the number of vehicles (number of vehicles in the CDS control zone), and Δt_{ini}. The variables that do not satisfy the model assumptions are excluded. The results of the ANCOVA model for different traffic condition is presented in Table 5.4.

Significant differences are observed between the performances of three HMIs whether in mixed traffic, CV environment or the overall results. The value of Δt_{ini}, which represents the time gap between the idea arrival time and the estimated arrival time for a CV when entering the CDS control zone, significantly impacts the average $|\Delta t|$ in each traffic condition and CDS zones. An increase in Δt_{ini} raise the difficulties for vehicle cooperation, and thus results in greater average $|\Delta t|$. The throttle usage also affects the CDS performance. If the CV drivers are using more throttle when entering the CDS control, the acceleration effect will be less with the same unit of throttle increase, which slows down the approximation to the optimal speed and increases average $|\Delta t|$. In mixed traffic, the number of vehicles in the CDS control is another factor that influences the performance. With more vehicles, especially more HDVs in the control zone, it is harder to find a sufficient gap, which results in greater Δt_{ini} and average $|\Delta t|$. Considering these variables have significant impact on the measurement average $|\Delta t|$, they are set to the covariates in the model. Hence, the performance between different HMIs can be identified while controlling the effects of the covariates. The post-hoc pairwise comparison of average $|\Delta t|$ between three guidance interfaces is shown in Figure 5.9. The Δt interface is significantly worse compared to the other two interfaces in the entire CDS control zone, the effective zone, and the steady zone. The difference between Δv interface and Δv-graphic interface is significant in the CV environment, while in the mixed traffic it is not significant. Nevertheless, in the mix traffic condition, Δv-graphic interface has lower average $|\Delta t|$, value in both “effective zone” and “steady zone” compared to Δv interface (mean=1.16 vs 1.51; mean=0.63 vs 0.72, respectively). In the CV environment, the Δv-graphic interface significantly outperformed Δv interface with 0.65 vs 0.80 in the mean value of average $|\Delta t|$. Furthermore, the random term of participants’ id is significant in the model results. It implies the heterogeneity of CV

drivers in the adaption to different speed guidance interfaces. Thus, in order to enhance the cooperative driving performance under the guidance, sufficient pre-train is recommended.

Table 5.4 ANCOVA model statistics for different traffic condition

		Mixed traffic		**CV environment**		**overall**	
		F-value	p-value	F-value	p-value	F-value	p-value
CDS control zone	**Variables**						
	Guidance interface	28.347	<0.001	70.220	<0.001	60.783	<0.001
	Approaching throttle usage	33.715	<0.001	56.030	<0.001		
	Δt_{ini}	249.143	<0.001	792.641	<0.001	681.980	<0.001
	Number of vehicles	0.528	<0.001				
	Random term						
	Participants ID	39.298	0.002	137.40	<0.001	97.810	<0.001
Effective zone	**Variables**						
	Guidance interface	44.653	<0.001	111.22	<0.001	102.400	<0.001
	Approaching throttle usage	21.721	<0.001	29.54	<0.001		
	Δt_{ini}	397.185	<0.001	1288.910	<0.001	1183.100	<0.001
	Number of vehicles	1.772	0.185				
	Random term						
	Participants ID	89.097	<0.001	718.800	<0.001	439.430	<0.001
Steady zone	**Variables**						
	Guidance interface	5.939	0.003	19.340	<0.001	19.39	<0.001
	Approaching throttle usage	2.180	0.142	0.941	0.333		
	Δt_{ini}	102.308	<0.001	455.863	<0.001	457.95	<0.001
	Number of vehicles	8.271	0.004				
	Random term						
	Participants ID	13.887	0.015				

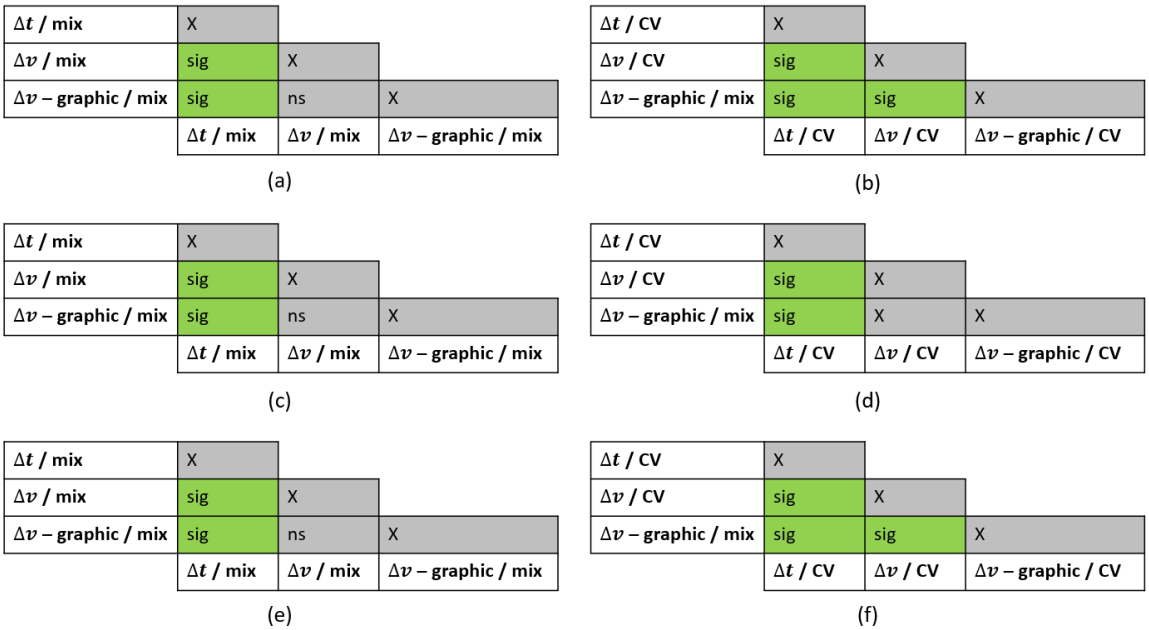


Figure 5.9 Post-hoc pairwise comparison of average $|\Delta t|$ between guidance interfaces

Note: (a) mixed traffic, entire CDS control zone; (b) CV environment, entire CDS control zone; (c) mixed traffic, effective zone; (d) CV environment, effective zone; (e) mixed traffic steady zone; (f) CV environment, steady zone.

5.2.2 Driving Comfort

In the CDS control zone, the CV drivers are receiving real-time speed guidance and adapting their speed constantly. The speed guidance could be unstable because any mis operation of the drivers or the expected behaviors of the HDV in mixed traffic could bring turbulence to the calculation of optimal speed. When the CV drivers are trying to approximate the suggested speed by using throttle or brake constantly, it might make the passenger feal discomfort. The jerk, which is the derivative of acceleration or deceleration, has been widely adopted for driving comfort-related research. The max jerk, min jerk, and average jerk during acceleration and deceleration period in mixed

traffic and CV environment are plotted in Figure 5.10 and Figure 5.11, respectively. The acceleration period is the time period that CDS speed is greater than the current speed, and the drivers accelerate to approximate the optimal speed; while the deceleration period is when the optimal speed is smaller than the optimal speed.

In the mix traffic scenario, the presence of HDV brought turbulence to the CDS when updating the real-time optimal speed for CV. Therefore, the drivers may accelerate or decelerate frequently to adapt to the optimal speed. For “easy mode”, no significant difference was observed for the jerks during acceleration or deceleration period between the three speed guidance interfaces. For “hard mode”, the Δv-graphic caused more changes in the drivers’ acceleration and deceleration behavior, where the maximum jerk (mean=152.49 m/s^3), minimum jerk (mean=-185.55 m/s^3), and average during acceleration period (mean=7.85 m/s^3) are greater in absolute values compared with jerks of the rest two interfaces. However, only the differences of the average jerk during acceleration period between the interfaces are significant (p=0.031).

The level of driving comfort in a CV environment is better compared with mixed traffic in terms of average jerks. During the acceleration period, the average jerk in mixed traffic is 7.62 m/s^3 while in the CV environment the average jerk is only 5.71 m/s^3 (p=0.047). In the CV environment, the Δv-graphic interface is the most comfortable interface in terms of average jerk during acceleration for “hard mode”, while for “easy mode” the Δt interface is best. However, there is also no significant difference between the results for the max jerk, min jerk, and average jerk. Hence, it can be concluded that the design of the proposed HMIs have no significant effect on driving comfort.

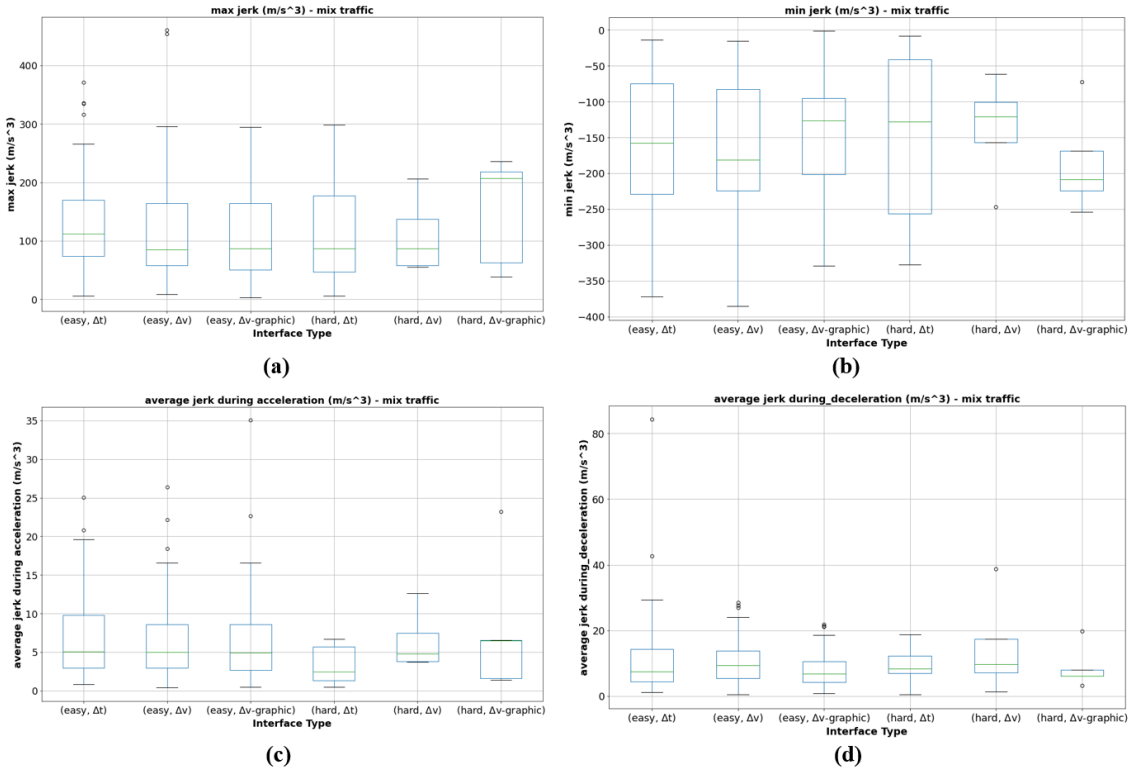


Figure 5.10 Driving comfort in the CDS control zone in mixed traffic

Notes: (a) max jerk; (b) min jerk; (c) average jerk during acceleration; (d) average jerk during deceleration.

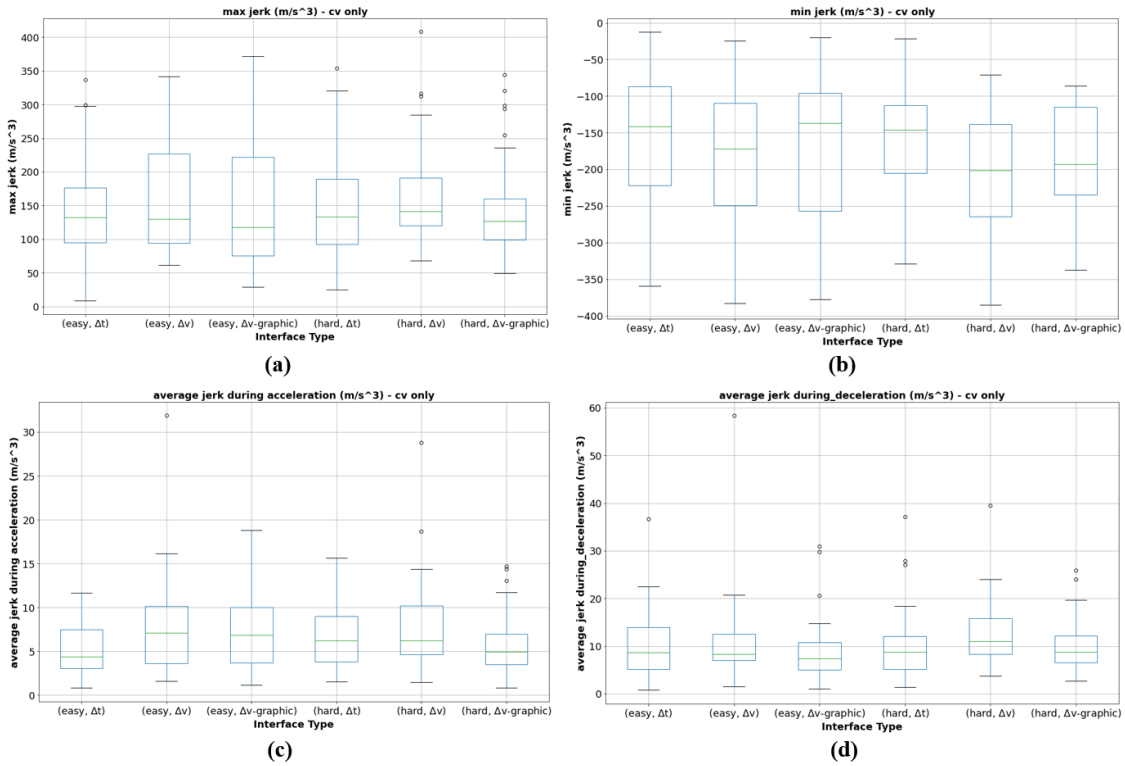


Figure 5.11 Driving comfort in the CDS control zone in CV environment

Notes: (a) max jerk; (b) min jerk; (c) average jerk during acceleration; (d) average jerk during deceleration.

5.3 Task 3

The MADQN model was trained for 10,000 episodes on the Sumo simulator, and the rewards versus episodes curve was plotted in Figure 5.12. For comparison, the baseline model that adopts Sumo default driving behavior model and parameters was also tested and with the same simulation settings as it used to train the MADQN model. It was observed that the proposed MADQN model outperformed the baseline model by a huge margin in terms of rewards. During the training period, the rewards curve starts to raise approximately at 1000 episodes, and it was at 2500 episodes that the rewards curve presents less fluctuation. After 8000 episodes, the rewards curve stopped to increase and ends at the value between 20-30. One reason for causing the fluctuation at the end

of the training periods was the random effects in the initial simulation state, as the vehicles were spawned at random positions in the designated intervals with desired speed distribution of N~(1,0.1). For the baseline model, the reason for obtaining large negative values was the diverging vehicle fails to find a sufficient gap for diverging and reduce speed greatly to wait for the right lane traffic pass through, and thus was rewarded large negative values over time.

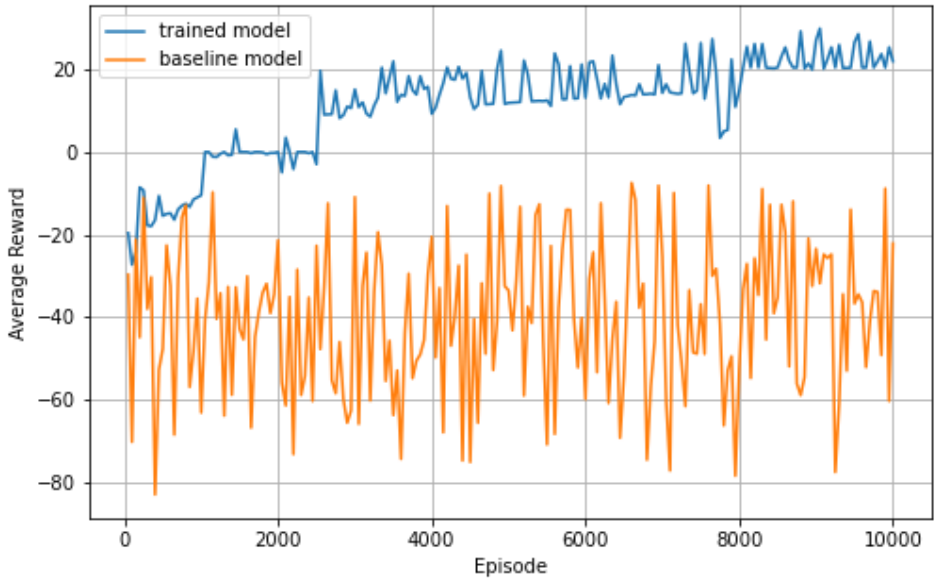


Figure 5.12 Episodes versus rewards

Furthermore, the trained model is applied to the decision making in the diverging process for additional 30 times of simulation. The performance is evaluated through different indicators including average travel time, successful diverge rate, and maximum deceleration rate, as shown in Table 5.5. The travel time is the time duration from the simulation starts to the time point when all six vehicles have passed the diverging point. The average travel time for the trained model is 21.71 seconds while for the baseline model the value is 26.52 seconds ($p<0.001$). It implies the trained decision-making model accelerates the diverging process by cooperating the vehicles to smoothly create a lane change gap rather reduce speed to form a gap at the end of the diverging area.

The average successful diverge rate is 0.92, which indicates 92% of the diverging vehicles successfully exited the freeway. The reaming 8% of the diverging vehicles failed to diverge mainly due to the initial state was designed with high difficulty for lane changing, and therefore under certain conditions the vehicles did not find a sufficient gap for diverging. Compared to the baseline model, the trained model generates a much smoother speed profile by causing less harsh breaking behaviors, as the average maximum deceleration during the diverging period is smaller than the baseline (1.47 vs 3.65). Hence, it can be concluded that the trained model improves the efficiency and reduces harsh deceleration compared to the baseline model while ensuring a decent successful diverging rate.

Table 5.5 Summary of off-ramp diverging simulation results

Model type	**Average Travel time (seconds)**	**Average successful diverge rate**	**Average Maximum deceleration (m/s^2)**
Baseline model	26.52	1	3.65
Trained model	21.71	0.92	1.47

6 Conclusions

The research focused on investigating the effects of cooperative driving at multiple driving scenarios. The study locations in this research include the non-signalized intersection and the freeway off-ramp. There are three tasks in the study: (1) developing the cooperative driving strategy (CDS) at non-signalized intersections for CAVs and CVs considering the mixed traffic; (2) designing the in-vehicle HMI for cooperative driving at non-signalized intersection; (3) proposing multi-agent reinforcement learning-based decision-making models for cooperative diverging at off-ramp. To test the effects of the proposed algorithms and HMIs, UCF SST developed multi-driver-in-the-loop co-simulation platform to conduct simulation experiments.

For task 1, the experiment results showed that the CDS reduced up to 53.8%, 66.4%, and 73.7% of travel time in CV-HDV, CV-CAV, and CAV environments, respectively. Driving speed and average throttle usage also increased significantly after applying CDS, which means drivers were driving faster and efficiency was enhanced. It is also found out that lower CV MPR and the complicated traffic state at intersection have significant negative impact on traffic efficiency. Furthermore, the conflicts number decreased with CDS embedded, especially for CAV. It shows that the proposed CDS is able to avoid potential conflicts and thus enhance safety.

For task 2, the three different HMIs were test in various traffic conditions through driving simulator experiments. The HMI that displays the time difference between optimal and estimated arrival time (Δt interface), is the worst interface, as the results suggest it produced the largest driving error. The other two HMIs, the Δv interface (displays the speed difference between the optimal speed and current speed) and Δv-graphic interface (converts speed difference into graphic display) performed better. In general, the Δv-graphic interface is better than the Δv interface, since its value of average driving error is significantly smaller than the value of Δv in the CV environment. In a mixed traffic environment, when the traffic condition is sophisticated and requires more speed changes, the Δv interface is better, while the Δv-graphic interface is more suitable for cooperative driving with a steadier suggested speed. In summary, although the Δv interface can effectively deliver most of the guidance information to the CV drivers, the Δv-graphic interface works better since it can also capture and display small variations in speed guidance to the CV drivers.

For task 3, a MADQN model was trained based on an off-ramp diverging scenario in the Sumo simulator. The model was trained 10,000 episodes and the reward curve showed that the MADQN model significantly outperformed the baseline model. On the other hand, the trained model accelerated the diverging the process by avoiding

mandatory lane changes at the end of the diverging areas and it also enhanced safety as harsh breaks at reduced while ensuring 92% of successful lane change rate in complicated diverging conditions.

References

[1] N. Highway Traffic Safety Administration and U. Department of Transportation, “TRAFFIC SAFETY FACTS Crash • Stats Critical Reasons for Crashes Investigated in the National Motor Vehicle Crash Causation Survey,” 2015.

[2] U.S. General Services AdministrationOffice of Motor Vehicle Managemen, “Crashes Are No Accident.” Accessed: Aug. 25, 2022. [Online]. Available: https://drivethru.gsa.gov/DRIVERSAFETY/DistractedDrivingPosterA.pdf

[3] S. Tsugawa, “Inter-vehicle communications and their applications to intelligent vehicles: an overview,” pp. 564–569, 2003, doi: 10.1109/ivs.2002.1188011.

[4] L. Li and F. Y. Wang, *Advanced motion control and sensing for intelligent vehicles*. 2007. doi: 10.1007/978-0-387-44409-3.

[5] L. Li, D. Wen, and D. Yao, “A Survey of Traffic Control With Vehicular Communications,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 15, no. 1, pp. 425–432, 2014, doi: 10.1109/JPROC.2011.2139850.

[6] R. Johri, J. Rao, H. Yu, and H. Zhang, “A Multi-Scale Spatiotemporal Perspective of Connected and Automated Vehicles: Applications and Wireless Networking,” *IEEE Intelligent Transportation Systems Magazine*, vol. 8, no. 2, pp. 65–73, 2016, doi: 10.1109/MITS.2016.2523719.

[7] A. R. Bill, “Modeling Reservation-based Autonomous Intersection Control in VISSIM,” 2013.

[8] G. A. Ubiergo and W. L. Jin, “Mobility and environment improvement of signalized networks through Vehicle-to-Infrastructure (V2I) communications,” *Transp Res Part C Emerg Technol*, vol. 68, pp. 70–82, 2016, doi: 10.1016/j.trc.2016.03.010.

[9] Z. Li, L. Elefteriadou, and S. Ranka, “Signal control optimization for automated vehicles at isolated signalized intersections,” *Transp Res Part C Emerg Technol*, vol. 49, pp. 1–18, 2014, doi: 10.1016/j.trc.2014.10.001.

[10] P. Dai, K. Liu, Q. Zhuge, E. H. M. Sha, V. C. S. Lee, and S. H. Son, “Quality-of-Experience-Oriented Autonomous Intersection Control in Vehicular Networks,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 7, pp. 1956–1967, 2016, doi: 10.1109/TITS.2016.2514271.

[11] X. F. Xie, S. F. Smith, L. Lu, and G. J. Barlow, “Schedule-driven intersection control,” *Transp Res Part C Emerg Technol*, vol. 24, pp. 168–189, 2012, doi: 10.1016/j.trc.2012.03.004.

[12] Q. He, K. L. Head, and J. Ding, “PAMSCOD: Platoon-based arterial multi-modal signal control with online data,” *Transp Res Part C Emerg Technol*, vol. 20, no. 1, pp. 164–184, 2012, doi: 10.1016/j.trc.2011.05.007.

[13] A. Gaur and P. Mirchandani, “Method for real-time recognition of vehicle platoons,” *Transp Res Rec*, no. 1748, pp. 8–17, 2001, doi: 10.3141/1748-02.

[14] R. Tachet *et al.*, “Revisiting street intersections using slot-based systems,” *PLoS One*, vol. 11, no. 3, pp. 1–9, 2016, doi: 10.1371/journal.pone.0149607.

[15] M. W. Levin, S. D. Boyles, and R. Patel, “Paradoxes of reservation-based intersection controls in traffic networks,” *Transp Res Part A Policy Pract*, vol. 90, pp. 14–25, 2016, doi: 10.1016/j.tra.2016.05.013.

[16] Y. J. Zhang, A. A. Malikopoulos, and C. G. Cassandras, “Optimal control and coordination of connected and automated vehicles at urban traffic intersections BT - 2016 American Control Conference, ACC 2016, July 6, 2016 - July 8, 2016,” *2016 American Control Conference (ACC)*, vol. 2016-July, pp. 6227–6232, 2016, [Online]. Available: http://dx.doi.org/10.1109/ACC.2016.7526648

[17] J. Wu, F. Perronnet, and A. Abbas-Turki, “Cooperative vehicle-actuator system: A sequencebased framework of cooperative intersections management,” *IET Intelligent Transport Systems*, vol. 8, no. 4, pp. 352–360, 2014, doi: 10.1049/iet-its.2013.0093.

[18] Y. Meng, L. Li, F. Y. Wang, K. Li, and Z. Li, “Analysis of Cooperative Driving Strategies for Nonsignalized Intersections,” *IEEE Trans Veh Technol*, vol. 67, no. 4, pp. 2900–2911, 2018, doi: 10.1109/TVT.2017.2780269.

[19] Z. Yang, H. Huang, G. Wang, X. Pei, and D. ya Yao, “Cooperative driving model for non-signalized intersections with cooperative games,” *J Cent South Univ*, vol. 25, no. 9, pp. 2164–2181, 2018, doi: 10.1007/s11771-018-3905-6.

[20] J. Lee and B. Park, “Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 13, no. 1, pp. 81–90, 2012, doi: 10.1109/TITS.2011.2178836.

[21] G. R. de Campos, P. Falcone, and J. Sjoberg, “Autonomous cooperative driving: A velocity-based negotiation approach for intersection crossing,” *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, no. March 2016, pp. 1456–1461, 2013, doi: 10.1109/ITSC.2013.6728435.

[22] P. Lin, J. Liu, P. J. Jin, and B. Ran, “Autonomous vehicle-intersection coordination method in a connected vehicle environment,” *IEEE Intelligent Transportation Systems Magazine*, vol. 9, no. 4, pp. 37–47, 2017, doi: 10.1109/MITS.2017.2743167.

[23] C. Dong, H. Wang, Y. Li, X. Shi, D. Ni, and W. Wang, “Application of machine learning algorithms in lane-changing model for intelligent vehicles exiting to off-ramp,” *Transportmetrica A: Transport Science*, vol. 17, no. 1, pp. 124–150, 2021, doi: 10.1080/23249935.2020.1746861.

[24] B. Wang, W. Li, H. Wen, and X. Hu, “Modeling impacts of driving automation system on mixed traffic flow at off-ramp freeway facilities,” *Physica A: Statistical Mechanics and its Applications*, vol. 573, p. 125852, 2021, doi: 10.1016/j.physa.2021.125852.

[25] Y. Zheng, B. Ran, X. Qu, J. Zhang, and Y. Lin, “Cooperative Lane Changing Strategies to Improve Traffic Operation and Safety Nearby Freeway Off-Ramps in a Connected and Automated Vehicles Environment,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 11, pp. 4605–4614, Nov. 2020, doi: 10.1109/TITS.2019.2942050.

[26] D. Chen *et al.*, “Deep Multi-agent Reinforcement Learning for Highway On-Ramp Merging in Mixed Traffic,” May 2021, [Online]. Available: http://arxiv.org/abs/2105.05701

[27] Z. el abidine Kherroubi, S. Aknine, and R. Bacha, “Novel Decision-Making Strategy for Connected and Autonomous Vehicles in Highway On-Ramp Merging,” *IEEE Transactions on Intelligent Transportation Systems*, 2021, doi: 10.1109/TITS.2021.3114983.

[28] C. Yu *et al.*, “Distributed Multiagent Coordinated Learning for Autonomous Driving in Highways Based on Dynamic Coordination Graphs,” *IEEE Transactions on Intelligent Transportation Systems*, vol. 21, no. 2, pp. 735–748, 2020, doi: 10.1109/TITS.2019.2893683.

[29] S. Chen, J. Dong, P. Ha, Y. Li, and S. Labi, “Graph neural network and reinforcement learning for multi-agent cooperative control of connected autonomous vehicles,” *Computer-Aided Civil and Infrastructure Engineering*, vol. 36, no. 7, pp. 838–857, Jul. 2021, doi: 10.1111/mice.12702.

[30] M. Tan, “Multi-Agent Reinforcement Learning: Independent vs. Cooperative Agents.”

[31] K. Lin, R. Zhao, Z. Xu, and J. Zhou, “Efficient large-scale fleet management via multi-agent deep reinforcement learning,” in *Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, Jul. 2018, pp. 1774–1783. doi: 10.1145/3219819.3219993.

[32] J. K. Terry, N. Grammel, S. Son, and B. Black, “Parameter Sharing For Heterogeneous Agents in Multi-Agent Reinforcement Learning,” May 2020, [Online]. Available: http://arxiv.org/abs/2005.13625

[33] A. H. Jamson, D. L. Hibberd, and N. Merat, “Interface design considerations for an in-vehicle eco-driving assistance system,” *Transp Res Part C Emerg Technol*, vol. 58, no. PD, pp. 642–656, Sep. 2015, doi: 10.1016/j.trc.2014.12.008.

[34] A. Masola, C. Gabbi, A. Castellano, N. Capodieci, and P. Burgio, “Graphic Interfaces in ADAS: From requirements to implementation,” in *ACM International Conference Proceeding Series*, Sep. 2020, pp. 193–198. doi: 10.1145/3411170.3411259.

[35] S. Azzi, G. Reymond, F. Mérienne, and A. Kemeny, “Eco-driving performance assessment with in-car visual and haptic feedback assistance,” *J Comput Inf Sci Eng*, vol. 11, no. 4, 2011, doi: 10.1115/1.3622753.

[36] C. Maag, A. K. Kraft, A. Neukum, and M. Baumann, “Supporting cooperative driving behaviour by technology – HMI solution, acceptance by drivers and effects on workload and driving behaviour,” *Transp Res Part F Traffic Psychol Behav*, vol. 84, pp. 139–154, Jan. 2022, doi: 10.1016/j.trf.2021.11.017.

[37] A. K. Kraft, C. Maag, and M. Baumann, “Comparing dynamic and static illustration of an HMI for cooperative driving,” *Accid Anal Prev*, vol. 144, Sep. 2020, doi: 10.1016/j.aap.2020.105682.

[38] Y. Ali, A. Sharma, M. M. Haque, Z. Zheng, and M. Saifuzzaman, “The impact of the connected environment on driving behavior and safety: A driving simulator study,” *Accident Analysis and Prevention*, vol. 144. 2020. doi: 10.1016/j.aap.2020.105643.

[39] M. R. Islam, “Safety and efficiency benefits of traffic signal countdown timers: A driving simulator study,” p. 159, 2014.

[40] C. J. G. van Driel, M. Hoedemaeker, and B. van Arem, “Impacts of a Congestion Assistant on driving behaviour and acceptance using a driving simulator,” *Transp Res Part F Traffic Psychol Behav*, vol. 10, no. 2, pp. 139–152, 2007, doi: 10.1016/j.trf.2006.08.003.

[41] H. Bellem, M. Klüver, M. Schrauf, H. P. Schöner, H. Hecht, and J. F. Krems, “Can We Study Autonomous Driving Comfort in Moving-Base Driving Simulators? A Validation Study,” *Hum Factors*, vol. 59, no. 3, pp. 442–456, 2017, doi: 10.1177/0018720816682647.

[42] A. K. Nandi, D. Chakraborty, and W. Vaz, “Design of a comfortable optimal driving strategy for electric vehicles using multi-objective optimization,” *J Power Sources*, vol. 283, no. June, pp. 1–18, 2015, doi: 10.1016/j.jpowsour.2015.02.109.

[43] A. Mashko *et al.*, “Virtual traffic signs-assessment of an alternative ADAS user interface with use of driving simulator.” [Online]. Available: https://www.researchgate.net/publication/306230912

[44] Y. Wu, M. Abdel-Aty, J. Park, and J. Zhu, “Effects of crash warning systems on rear-end crash avoidance behavior under fog conditions,” *Transp Res Part C Emerg Technol*, vol. 95, pp. 481–492, Oct. 2018, doi: 10.1016/j.trc.2018.08.001.

[45] Z. Wang *et al.*, “A Digital Twin Paradigm: Vehicle-to-Cloud Based Advanced Driver Assistance Systems,” in *IEEE Vehicular Technology Conference*, May 2020, vol. 2020-May. doi: 10.1109/VTC2020-Spring48590.2020.9128938.

[46] K. Gajananan *et al.*, “A cooperative ITS study on green light optimisation using an integrated traffic, driving, and communication simulator,” *Australasian Transport Research Forum, ATRF 2013 - Proceedings*, 2013.

[47] M. Heesen, M. Baumann, J. Kelsch, D. Nause, and M. Friedrich, “Investigation of Cooperative Driving Behaviour during Lane Change in a Multi-Driver Simulation Environment,” *Human Factors: a view from an integrative perspective.*

Proceedings HFES Europe Chapter Conference Toulouse, no. figure 1, 2012, [Online]. Available: http://hfes-europe.org

[48] L. Yue, M. Abdel-Aty, and Z. Wang, “Effects of connected and autonomous vehicle merging behavior on mainline human-driven vehicle,” *Journal of Intelligent and Connected Vehicles*, vol. 5, no. 1, pp. 36–45, Feb. 2022, doi: 10.1108/jicv-08-2021-0013.

[49] J. B. Kenney, “Dedicated short-range communications (DSRC) standards in the United States,” *Proceedings of the IEEE*, vol. 99, no. 7, pp. 1162–1182, 2011, doi: 10.1109/JPROC.2011.2132790.