Connectivity-Based Cooperative Ramp Merging in Multimodal and Mixed Traffic Environment

September 2022

A Research Report from the Pacific Southwest Region University Transportation Center

Guoyuan Wu, University of California at Riverside

Xuanpeng Zhao, University of California at Riverside

Xishun Liao, University of California at Riverside

Kanok Boriboonsomsin, University of California at Riverside

Pacific Southwest Region UTC
University Transportation Center

UCR CE-CERT

TECHNICAL REPORT DOCUMENTATION PAGE

1. Report No. PSR-21-20	**2. Government Accession No.** N/A	**3. Recipient's Catalog No.** N/A
4. Title and Subtitle Connectivity-Based Cooperative Ramp Merging in Multimodal and Mixed Traffic Environment		**5. Report Date** September 8th, 2022 **6. Performing Organization Code** N/A
7. Author(s) Guoyuan Wu, 0000-0001-6707-6366 Xuanpeng Zhao, 0000-0001-7073-6927 Xishun Liao, 0000-0002-7219-8971 Kanok Boriboonsomsin, 0000-0003-2558-5343		**8. Performing Organization Report No.** TBD
9. Performing Organization Name and Address METRANS Transportation Center University of Southern California University Park Campus, RGL 216 Los Angeles, CA 90089-0626		**10. Work Unit No.** N/A **11. Contract or Grant No.** USDOT Grant 69A3551747109
12. Sponsoring Agency Name and Address U.S. Department of Transportation Office of the Assistant Secretary for Research and Technology 1200 New Jersey Avenue, SE, Washington, DC 20590		**13. Type of Report and Period Covered** Final report (09/16/2021 – 09/15/2022) **14. Sponsoring Agency Code** USDOT OST-R
15. Supplementary Notes N/A		

16. Abstract

Freeway ramp merging involves conflict of vehicle movements that may lead to traffic bottlenecks or accidents. Thanks to advances in connected and automated vehicle (CAV) technology, a number of efficient ramp merging strategies have been developed. However, most of the existing CAV-based ramp merging strategies assume that all the vehicles are CAVs or do not differentiate vehicle type (i.e., passenger cars vs. heavy-duty trucks). In this study, we propose a decentralized cooperative ramp merging application for connected vehicles (both connected trucks and connected cars) in a mixed traffic environment. In addition, we develop a multi-human-in-the-loop (MHuiL) simulation platform that integrates SUMO traffic simulator with two game engine-based driving simulators, allowing us to investigate the interactions between two human drivers under various traffic scenarios. The case study shows that the decentralized cooperative ramp merging application, which provides speed guidance to the connected vehicles involving in ramp merging, helps increase the time headways of the involved vehicles and smooths their speed profiles. With the speed guidance, the median minimum time headway for the yielding car on the mainline increases by 57%. Also, its speed variation decreases by 17% while the speed variation of the merging truck from the on-ramp decreases by 19%. These results demonstrate the potential for the proposed application to improve the safety and efficiency of ramp merging for heavy-duty trucks, which will be particularly useful at on-ramps with relatively short merging lane. The experiments conducted also validate the effectiveness of the developed MHuiL platform for human factor research.

17. Key Words Connected and automated vehicles, cooperative ramp merging, V2X communications, multi-human-in-the-loop (MHuiL)		**18. Distribution Statement** No restrictions.	
19. Security Classif. (of this report) Unclassified	**20. Security Classif. (of this page)** Unclassified	**21. No. of Pages** 39	**22. Price** N/A

Form DOT F 1700.7 (8-72) Reproduction of completed page authorized

Contents

About the Pacific Southwest Region University Transportation Center

The Pacific Southwest Region University Transportation Center (UTC) is the Region 9 University Transportation Center funded under the US Department of Transportation's University Transportation Centers Program. Established in 2016, the Pacific Southwest Region UTC (PSR) is led by the University of Southern California and includes seven partners: Long Beach State University; University of California, Davis; University of California, Irvine; University of California, Los Angeles; University of Hawaii; Northern Arizona University; Pima Community College.

The Pacific Southwest Region UTC conducts an integrated, multidisciplinary program of research, education and technology transfer aimed at *improving the mobility of people and goods throughout the region.* Our program is organized around four themes: 1) technology to address transportation problems and improve mobility; 2) improving mobility for vulnerable populations; 3) Improving resilience and protecting the environment; and 4) managing mobility in high growth areas.

U.S. Department of Transportation (USDOT) Disclaimer

Disclosure

Dr. Guoyuan Wu (PI), Dr. Kanok Boriboonsomsin (Co-PI), Xuanpeng Zhao and Heeson Liao (both are PhD students), conducted this research titled, "Connectivity-Based Cooperative Ramp Merging in Multimodal and Mixed Traffic Environment" at Bourns College of Engineering – Center for Environmental Research & Technology (CE-CERT) at University of California, Riverside. The research took place from 09/16/2021 to 09/15/2022 and was funded by a grant from the U.S. Department of Transportation in the amount of $99,999. The research was conducted as part of the Pacific Southwest Region University Transportation Center research program.

Acknowledgements

N/A.

Abstract

On freeways, ramp merging has been considered as a representative scenario that may cause traffic bottlenecks or hotspots of traffic accidents. Thanks to advances in connected and automated vehicle (CAV) technology, efficient ramp merging strategies have been developed. However, most existing CAV-based ramp merging strategies assume that all the vehicles are CAVs or do not differentiate vehicle type (i.e., passenger cars vs. heavy-duty trucks). In this study, we propose a decentralized cooperative ramp merging application for connected vehicles (both connected trucks and connected cars) in a mixed traffic environment. In addition, we develop a multi-human-in-the-loop (MHuiL) simulation platform, consisting of SUMO and two game engine-based driving simulators, which allows us to investigate interactions between two drivers simultaneously. The case study shows that the application can help increase the vehicle time headways between involved vehicles and smooth their speed profiles. It turns out that with the speed guidance, the median minimum time headway for the involved car increases by 57%, and speed variations of the trucks driving on the mainline decrease by 19%. These experiments also validate the effectiveness of the developed MHuiL platform for human factor research.

Connectivity-Based Cooperative Ramp Merging in Multimodal and Mixed Traffic Environment

Executive Summary

Problem Statement

Ramp merging attracts a significant amount of attention from researchers and traffic operators, due to the potential safety and mobility concerns caused by the chaotic nature (e.g., lane change, speed variations) of traffic. The emergence of connected and automated vehicle (CAV) technology unlocks unprecedented opportunities to address ramp merging problems. However, most of CAV-based cooperative ramp merging algorithms assume the traffic flow is homogeneous, i.e., all the vehicles are light-duty CAVs, although some may differentiate the powertrain types (e.g., gasoline vs. electric). In addition, many studies consider a full penetration rate of CAVs in the testing scenarios and assume the involved vehicles are able to follow the reference speed perfectly. However, all these assumptions are far from being realistic, which reduces the practicality and flexibility of those algorithms.

Proposed Solution

To address these research gaps, the research team has developed a vehicle-to-everything (V2X) based decentralized cooperative ramp merging system application for a multimodal (e.g., cars, trucks) and mixed traffic (including both connected and non-connected vehicles) environment. In addition, the research team has developed a multi-human-in-the-loop (MHuiL) simulation platform that integrates SUMO traffic simulator with two game engine-based driving simulators. This platform enables the research team to collect driving behaviors from two drivers simultaneously, facilitating research on modeling and validation of interactions between drivers in an immersive simulation environment. The research team recruited 8 volunteers (one for truck driving simulator and the other 7 for passenger car driving simulator) to drive the passenger car simulator in the simulation experiments where a research team member drove the truck simulator. The experimental scenarios include both on-ramp and mainline driving where the drivers were either provided or not provided with speed guidance by the developed cooperative ramp merging application.

Findings

The experiments for the case study show that the developed MHuiL simulation platform is a valuable analysis/modeling/simulation tool for driving behavior related research. In addition, the experimental results indicate that the proposed cooperative ramp merging application is effective at improving traffic safety and driving smoothness. Using the application, the median minimum time headway for the yielding passenger car on the mainline increases by 57%. Also, its speed variations decrease by 17% while the speed variation of the merging truck from the on-ramp decreases by 19%. These results demonstrate the potential for the proposed application to improve the safety and efficiency of ramp merging for heavy-duty trucks, which will be particularly useful at on-ramps with relatively short merging lane.

Next Steps

Based on the experience in this project, there are some potential improvements that can be made in future work:

- Sensor noises, such as Global Navigation Satellite System (GNSS) errors, and personalized behavior/preference can be considered in the cooperative ramp merging algorithm design.
- Communication modules via co-simulation with network simulators (e.g., NS-3) can be integrated into the current MHuiL platform to mimic a more realistic wireless communication environment (e.g., with packet loss, communication delay).
- More effective human-machine interfaces or information delivery approaches (e.g., using audio signals) can be explored for improving the effectiveness of this advanced driving assistance system (ADAS).

Introduction

As a representative scenario along freeways, on-ramp merging and its related research attracts a significant amount of attention from researchers and traffic operators due to the potential safety, mobility and environmental concerns caused by the chaotic nature of traffic in the merging area. The intensive speed fluctuations and weaving maneuvers often lead to traffic congestion or shockwaves along the upstream segments on both mainline and on-ramp, leading to increased energy consumption and mobile-sourced pollutant emissions.

Ramp metering is a widely used ramp merging management method, which utilizes traffic signals (usually consisting of two-phase signal timing, i.e., green and red) installed at highway on-ramps to regulate the inflow rate of traffic entering the mainline in response to prevailing mainline traffic conditions [1]. Existing research on ramp metering can be divided into three categories [2], namely rule-based approaches, control-based approaches, and learning-based approaches. However, since ramp metering may introduce stop-and-go maneuvers to the on-ramp vehicles, it may significantly increase their travel time and energy consumption. In addition, merging maneuvers between on-ramp vehicles and mainline vehicles cannot be well coordinated via ramp metering, which may still pose certain safety risks for merging vehicles or cause disturbances for the mainline traffic as well. At some poorly-designed locations, on-ramp vehicles (especially heavy-duty trucks) may experience difficulties accelerating to a desired speed for safe and comfortable merging maneuvers, due to lack of acceleration lane.

Thanks to the advancement of connected and automated vehicle (CAV) technology, a variety of algorithms have been proposed and developed to address ramp merging problems [3, 4]. Most of these algorithms assume the traffic flow is homogeneous, i.e., all the vehicles are light-duty CAVs, although some may differentiate the powertrain types (e.g., gasoline vs. electric) [5]. In addition, many studies apply optimal control of vehicular strings/groups in a centralized manner, where the merging sequence has been well determined and all the involved vehicles are assumed to be able to follow the guidance and planned trajectories exactly. However, all these assumptions are far from being realistic, which reduces the practicality and flexibility of those algorithms.

To address the aforementioned research gaps, the research team herein designs, develops, implements, and evaluates a vehicle-to-everything (V2X) based cooperative ramp merging system for a multimodal (e.g., cars, trucks) and mixed traffic (including both connected and non-connected vehicles) environment, which can improve safety performance and smooth traffic flow at the highway ramp merging area.

Literature Review

In this section, we will review the state-of-the-art studies on both ramp merging algorithms based on vehicle-to-everything (V2X) communications and emerging simulators that enable connected and automated vehicle (CAV) research.

V2X-based ramp merging algorithms

Numerous ramp merging methodologies have been proposed recently. In a fully connected environment, a number of ramp merging strategies have been developed to increase road safety and efficiency by leveraging CAV technology [6]. The cooperative ramp merging was formulated as two optimal trajectory planning problems for a pair of the ramp and mainline vehicles by Zhou et al. [7] without presuming a merging location. Rios-Torres et al. presented an optimization framework and an analytical closed-form solution that allowed online coordination of CAVs at ramp merging zones [8], and further studied the impact of partial penetrations of CAVs on fuel consumption and traffic flow for the ramp merging scenario [9]. Besides optimal control approaches, game theory with full information was adopted by Min et al. [10] to find a Nash equilibrium for the ramp merging case that involved multiple vehicles, utilizing the fully connected environment.

However, most of these studies rely on a strong assumption of a 100% CAV penetration rate. In contrast, especially in mixed traffic with a low penetration rate, CAVs can only acquire limited information from the legacy vehicles within the detection range of CAVs. To coordinate the cooperation in mixed traffic ramp merging, Huang and Sun [11] proposed a dynamic programming-based approach, capturing the cooperative and non-cooperative behaviors. Considering the mixed traffic environment, Liao et al. [12] developed a game theory-based ramp merging strategy for CAVs in the mixed traffic, which was a decentralized agent-based algorithm for each vehicle to choose the optimal merging sequence.

Besides passenger cars, studies on the truck-involved scenario become popular. Truck platooning impacts near the merging area were pointed out by Duret et. al. [13], who assumed that the on-ramp cars should yield to the mainline trucks and proposed a heuristic-based solution to provide a suggestion of merging time and merging location. Moreover, the off-ramp area could be affected by truck platooning as well. Zhao et. al. [14] discussed how the truck platoon affects traffic efficiency in off-ramp regions and devised an approach to calculate the boundary conditions for severe congestion mitigation. To study the ramp merging interaction between trucks and cars, driving patterns of both sides cannot be neglected. Lu et. al. [15] analyzed lane-change conflicts between cars and trucks in the merging section and compared the time-to-collision between different types of lane change conflicts (i.e., car-car, car-truck, etc.). Different from cars, there are many constraints for trucks that need to be considered in the algorithm, e.g., acceleration and safe intervehicle gap. The gap acceptance of truck drivers during lane change was studied by Nobukawa et al. [16], providing a guideline for the parameter selection in this study.

Nevertheless, existing literature on-ramp merging coordination only considers the coordination for passenger car but seldom discusses the algorithm for truck operation. In this study, we aim to develop an algorithm to coordinate both passenger cars and trucks for smoother and safer merging at ramps.

Advanced simulators for CAV research

For evaluating advanced driving assistance systems (ADAS) and cooperative automated driving systems (CADS), many advanced simulators have been developed in the past few decades. They can be categorized into two types: microscopic traffic simulators and vehicle-level simulators. Each type of these simulators has its own advantages and focus arenas. From the traffic aspect, commonly used microscopic simulators include SUMO [17], Aimsun [18], VISSIM [19], and etc. They can generate realistic traffic flows and simulate their behaviors with well-calibrated car-following and lane-changing models. In addition, vehicle-level simulators such as SVL [20], CARLA [21], Gazebo [22], Carsim [23], and PreScan [24] can model realistic vehicle dynamics and complex sensor characteristics including camera, Radar, LiDAR, GNSS, and etc. Among these vehicle-level simulator, game engine-based simulators have more advanced features such as creating photo-realistic virtual environments with varying weather conditions and providing human input interfaces, enabling human-in-the-loop (HuiL) simulations.

However, considering the pros and cons of an individual type of simulators, it is accepted that a single simulator is not enough for modeling and evaluating ADAS or CADS in a realistic testing environment. To leverage the capabilities of multiple simulators, some recent research integrated VISSIM with driving simulators to assess the influence of adverse weather on traffic flow characteristics [25]. Besides, SUMO was coupled with an open-source software framework, called CommonRoad [26] which can provide a benchmark for motion planning of automated vehicles. With such integration, motion planners can be evaluated and compared under realistic traffic scenarios provided by SUMO [27]. Pariota. et al. integrated SUMO and Matlab/Simulink to provide realistic vehicle dynamics, driver behaviors and traffic models [28]. V2XSim is a V2X simulator platform designed for connected vehicles and built with Gazebo simulation engine to model the vehicle mechanics and physics [29]. In another study [30], researchers developed a microscopic driver-centric simulator by integrating Unity and SUMO. Moreover, to provide a comprehensive driving simulator platform, more advanced simulation platforms aim to cover all three main aspects by merging traffic simulators, vehicle (driving) simulators, and network simulators together. Xu et al. proposed a generalized framework called OpenCDA which consists of CARLA and SUMO [31]. However, none of these studies consider more realistic driving behavior via the human-in-the-loop (HuiL) approach.

By taking advantage of 3D vehicle-level simulators, researchers can analyze human behaviors under CAV application environments using the HuiL approach. The HuiL simulator is a prototype platform for quickly exploring novel in-the-loop applications that can enhance the interactions between human beings and the physical world [32]. HuiL is widely used in different research topics highly related to human interaction with control systems. For example, in the work of rollover prevention for sport utility vehicles, the researchers validated the performance of the anti-rollover control via HuiL [33]. They emphasized that the reason for involving HuiL is that the driver's interaction with and perception of the system performance may cause inconsistency problems and many safety systems are designed without such consideration. With HuiL, some researchers proposed a synthetic approach to solving safety-critical interaction problems in the SAE Level 3 automated vehicles which are mostly autonomous and only need limited driver intervention [34].

To understand human behaviors in response to ADAS applications and the according effects to the traffic safety and environmental sustainability, many integrated simulators are not only capable of assessing ADAS effects and efficiency but also taking human factors into account. Hussein et al. proposed a 3D simulator for cooperative ADAS, and AVs called 3DCoAutoSim which is composed of SUMO, ROS, and Unity [35]. With the simulator, driving behavior can be evaluated by different types of vehicles and the collected driving data can be used for calibration. Gao et al. proposed a co-simulation by integrating ROS and Aimsun, which allows a user to drive an ego vehicle in the traffic flow to investigate driving behavior [36]. The integrated traffic-driving-networking simulator (ITDNS) exploited PARAMICS, NS-2, and driving simulator to create a virtual environment, allowing human driver to control a vehicle while communicating with other vehicles and infrastructure [37]. To evaluate the performance of the integrated simulator, the researchers conducted a case study to analyze the benefits of an eco-signal system regarding the human responses. Zhao et al. developed a co-simulation platform incorporating SUMO, Unity, and AWS to collect driving data via HuiL and provide personalized data analysis and data storage [38].

In addition to single driver HuiL, some multi-driver simulation systems have been developed for investigating the dynamic interaction between human driven vehicles and interrelationship between individual driver’s behavior. In [39], researchers devised a multi-driver simulation to evaluate two cooperative ADAS applications and concluded the cooperative ADAS could help drivers to have larger safety distance and less driving anger. To enable the interactive driving behavioral analysis, Xu et al. developed a multi-driver simulator system and validated the performance with merging strategies [40]. Yue et al. designed a merging scenario with three human drivers to investigate the impacts of merging strategies [41]. However, none of them take advantage of microscopic traffic simulator for realistic traffic environment generation or consider potential problems caused by heavy-duty vehicles involved in transportation. In this study, we designed a co-simulation platform integrating SUMO and Unity to investigate ADAS effects on the interactions between a passenger vehicle and a truck both driven by human drivers in a highway-ramp merging scenario.

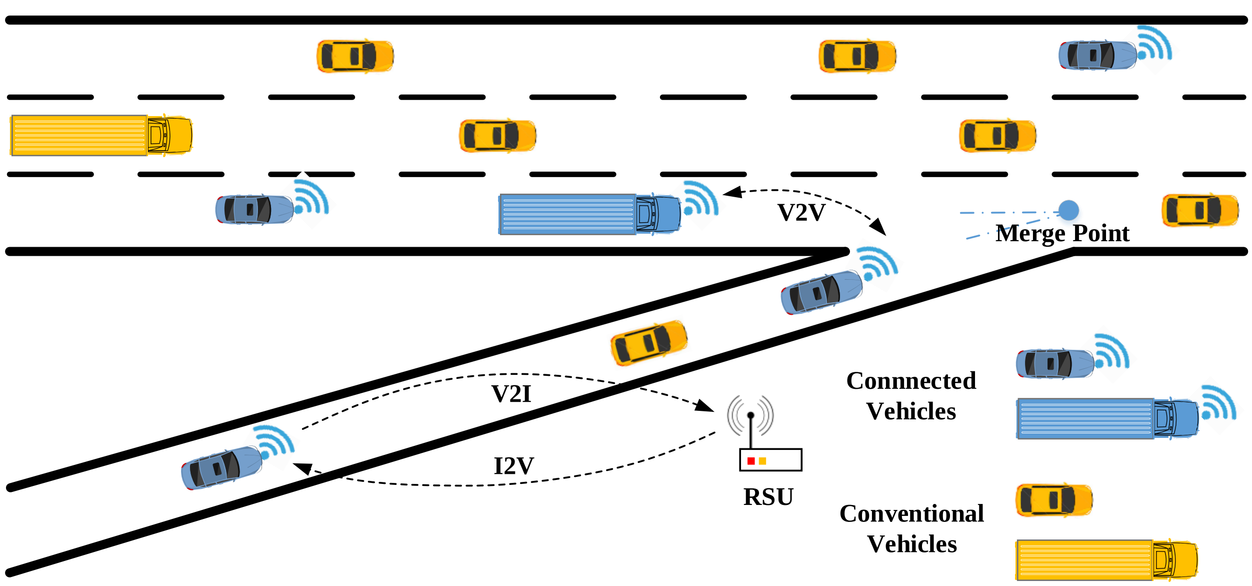
Methodology

In this project, the proposed cooperative ramp merging system leverages the emerging vehicle-to-everything (V2X) technology for connected vehicles (considering both human-driven and automated vehicles). In addition, the driving guidance or control strategies are developed with the consideration of involved vehicles' types (e.g., passenger cars, heavy-duty trucks), their dynamics (e.g., maximum acceleration rate, braking distance), as well as imperfection of human behaviors.

Problem formulation

The proposed system is illustrated in Figure 1. By leveraging the V2X communications, dynamic decision making and motion control is adopted to flexibly and efficiently coordinate the merging sequence, time, and speed in a mixed traffic environment. The research team develops decentralized sequencing and speed guidance algorithms that can take into account heterogeneous characteristics of different vehicles (such as maximum acceleration limits and braking rates). The ramp merging scenario is a representative game problem with multiple players considering competition and/or cooperation of the right-of-the-way (i.e., spatio-temporal occupation) in the merging area. Therefore, the research team formulates the problem as a cooperative or non-cooperative game, depending on the connectivity and/or sensor availability of involved vehicles. The *Game Theory* is applied to dynamically determine roles of the leader and follower(s) or the necessity of lane change(s) for safe and efficient merging. In addition, a decentralized multi-agent system (MAS) approach, such as consensus-based algorithm [42] is developed for provision of driving guidance or vehicle control. Compared to centralized algorithms, the proposed decentralized algorithm is more suitable for multi-modal and mixed traffic scenarios and more adaptive to handle disturbances (such as lane changes). More details of the developed algorithm will be presented in the following section.

Figure 1. Connectivity-based Cooperative Ramp Merging System

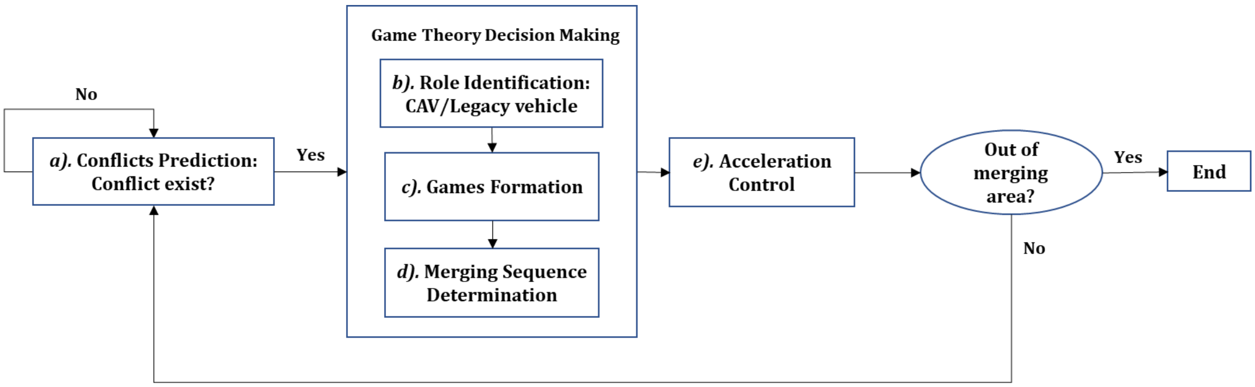


Connectivity-based cooperative ramp merging

Strategy Workflow

Our strategy is designed from a decentralized agent-based model perspective, allowing vehicles to act independently. The strategy workflow is shown in **Figure** , where every vehicle goes through this process at each time step. Five major modules are functioning to support the strategy.

Figure 2. System Workflow of the Mixed Traffic Ramp Merging Strategy for CAVs



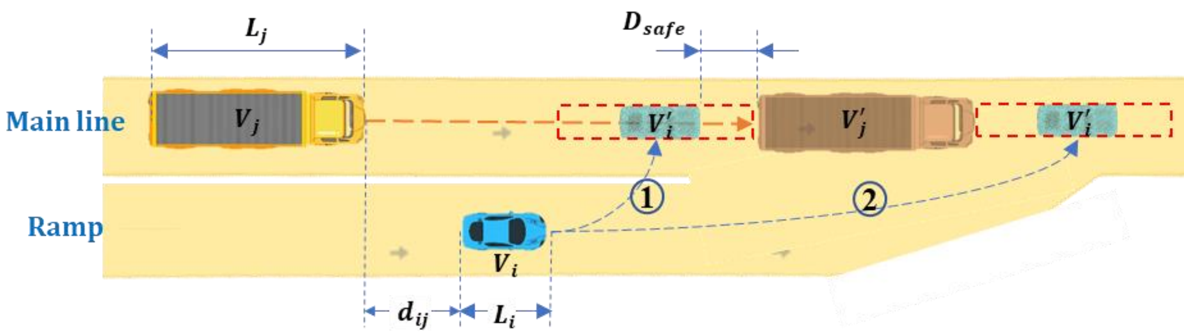
a) *Conflict Prediction Module*: Based on the information from the radar system or other CAVs, this module projects the ego vehicle and its surrounding vehicles into the future to see whether conflicts exist in the next prediction window. As

Figure illustrates, when the projected surrounding vehicle V_j' is out of the safe distance of the projected ego vehicle V_i', there is no conflict; otherwise, this surrounding vehicle will be classified as a potential conflict and added to the conflict list. The analytical form of projection and conflict prediction can be expressed in Equation (1).

$$\begin{cases} No\ conflict, & \begin{array}{l} v_i \times \Delta t + D_{safe} + L_i/2 - d_{ij} \leq v_j \times \Delta t - L_j/2 \\ or \\ v_i \times \Delta t - D_{safe} - L_i/2 - d_{ij} \geq v_j \times \Delta t + L_j/2 \end{array} \\ Potential\ conflict, & else \end{cases} \tag{1}$$

where v_i is the speed of ego vehicle; v_j is the speed of its surrounding vehicle; $d_{ij} = x_i - x_j$ is the current clearance; Δt is the simulation time step length; D_{safe} is the safe clearance, which is a speed-varying term depending on both the minimum static clearance and safe time headway of the ego vehicle.

Figure 3. Conflict Prediction Module



b) *Role Identification Module*: This module is developed to first classify ego vehicle's potential competitors into either CAVs or legacy vehicles, and then classify the competitors into either cars or trucks, which determine the game type in Module d) and the acceleration in Module (e). Basically, the vehicle type can be identified based on the signal association from both radar detection and wireless communication.

c) *Game Formation Module*: The game formulation process is triggered once it receives a conflict notice from *Conflict Prediction Module*, where ego vehicle forms an individual (two-player) game with each of its competitors (i.e., each potentially conflicting vehicle). If the competitor is CAV, the game will be a cooperative one. Otherwise, it is a non-cooperative game. In each game, each player may choose to be a follower or a leader. The corresponding expected acceleration and costs are calculated for each leader-follower combination, and the acceleration rates are computed by Acceleration Control Module to be introduced later. To dynamically adapt to the mixed traffic environment, a game is played in each time step, if a conflict exists. The game starts when the conflict emerges and ends until this conflict is solved.

d) *Merging Sequence Determination Module*: The merging sequence determination module is the last part of the game theory-based algorithm. The purpose of this module is to coordinate the merging sequence dynamically by utilizing results from the game formation module. Each vehicle will obtain its role with respect to its competitor (i.e., leader or follower), as well as an optimal longitudinal acceleration that satisfies the safety constraints, by solving a cost function. If the competitor is a truck, the cost function will be modified, in terms of safety (e.g., gap acceptance), mobility and comfort. If two competitors are both CAVs, they will share respective costs and make a game-wise optimal decision together. Details of the game theory-based algorithm will be introduced in the next subsection.

e) *Acceleration Control Module*: This module is responsible for two main goals. The first one is to ensure ego vehicle can run at the desired longitudinal speed and track the lane. The second one is to perform the lane change maneuver safely, once the lane change condition is satisfied. Since our algorithm focuses on acceleration control, we formulate the vehicle longitudinal motion with a standard vehicle kinematic as equation (2), where the longitudinal acceleration is used as the control input.

$$\begin{bmatrix} s(k+1) \\ v(k+1) \end{bmatrix} = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} s(k) \\ v(k) \end{bmatrix} + \begin{bmatrix} \Delta t^2/2 \\ \Delta t \end{bmatrix} a(k) \tag{2}$$

where s, v and a represent the displacement, speed, and acceleration of the vehicle, respectively.

Once the ego vehicle confirms its target vehicle and the associated states, the consensus control algorithm [42] from our previous research is adopted to compute the acceleration. This allows the ego vehicle i to maintain a desired inter-vehicle gap and the same speed as its target vehicle j:

$$\begin{aligned} a_{ref}(k+1) = -\alpha_{ij}\beta_{ij} \cdot [(s_i(k) - s_j(k) + l_j + v_i(k) \\ \cdot (t_{ij}^g(k) + \tau_{ij}(k))) + \gamma_i \cdot (v_i(k) - v_j(k)) \end{aligned} \tag{3}$$

Where α_{ij} denotes the value of adjacency matrix; β_{ij} and γ_i are control gains; $\tau_{ij}(t)$ denotes the time-varying communication delays between two vehicles; and $t_{ij}^g(t)$ is the time-varying desired time gap between two vehicles.

Game Theory-Based Merging Sequence Determination

During the merging process, complex conflict can be summarized with three types of scenarios, including the interactions between 1) two legacy vehicles, 2) two CAVs, and 3) a CAV and a legacy vehicle. This algorithm will only discuss the CAV(s) or truck-involved conflicts, since the conflicts between two legacy vehicles cannot be coordinated directly by CAVs, and merging without a truck is not in the scope of this project. Hereafter, this study will analyze the merging strategy from the perspective of the ego vehicle (CAV). Assumptions and specifications that are generally common in related literature are made as below:

The proposed algorithm aims to control the longitudinal speed to provide a safe merging space, and low-level control of the steering angle is outside the scope of this study.

All CAVs that are involved in conflicts act cooperatively to achieve an optimal goal.
The communication module and perception system of CAVs in the platform are assumed to be ideal. Therefore, no communication delay and packet loss are considered, and CAVs are capable of acquiring perfect information.

The proposed algorithm guarantees string stability within a pure CAV platoon instead of the whole mixed traffic flow. Since the string stability of mixing two car-following models is a complicated problem, it is still an open topic and out of our research interest.

Game Formulation

When a potential conflict exists in the merging area, at least one of the mainline vehicles and ramp vehicles needs to adjust its speed for a certain merging sequence. For the decision-making purpose, Game Theory is adopted for CAVs to evaluate their situation and then figure out the optimal merging strategy. A two-player non-zero-sum game is used in this study to handle each conflict by providing a merging sequence for each player in the game.

In such a game, ego vehicle is named Player 1 (hereafter "P1"), while its competitor is named Player 2 (hereafter "P2"). Both P1 and P2 can choose either to be a leader or a follower, with the action set given as $A(P1) = \{1\text{: To be the leader}, 2\text{: To be the follower}\}$, and $A(P2) = \{1\text{: To be the leader}, 2\text{: To be the follower}\}$.

The motivations of mainline vehicles and ramp vehicles may be different: mainline vehicles attempt to drive safely without compromising in travel speed, while ramp vehicles have to worry about the remaining distance to the end of merging area. As the remaining distance decreases, the merging intention of ramp vehicles may grow, and this anxiety can be expressed as the risk value in the cost function.

Cost Function

Safety is always the first priority to be considered. For each action of ego vehicle in the game, a corresponding suggested acceleration $\hat{a}$ is calculated by the control algorithm. Therefore, we can predict the time-to-collision (TTC) in next time step of each action. From the perspective of ego vehicle, the predicted TTC for any pair of players can be formulated in Equation (4).

$$\hat{t}_{TTC} = [d_{gap} + \Delta\hat{d}_{gap}]/[v_f + \Delta v_f - (v_p - \Delta v_p)], if\ v_f + \Delta\hat{v}_f > v_p - \Delta\hat{v}_p \quad (4)$$

where v_f, and $\Delta\hat{v}_f$ are the current speed and the predicted speed change of the following vehicle, respectively; v_p and $\Delta\hat{v}_p$ are the current speed and the predicted speed change of the preceding vehicle, respectively; d_{gap} is the current inter-vehicle gap; $\Delta\hat{d}_{gap}$ is the predicted gap change.

In the "two CAVs" scenario, these predicted values in Equation (4) are shared with each player because of the vehicle communication. For the "a CAV and a legacy vehicle" scenario, the predicted values of legacy vehicles can be assumed to be unchanged during a small time interval Δt (0.02s in the simulation), allowing CAV to estimate legacy vehicles' action. For example, in Equation (4), if the following vehicle is a legacy vehicle, the predicted speed change $\Delta\hat{v}_f = 0$.

However, using only TTC is not enough to quantify the safety risk [43]. For example, if the preceding vehicle is faster than the following vehicle, TTC will be negative. Moreover, if the difference between the following vehicle and the preceding vehicle is small, it will generate a huge TTC indicating safety even though the inter-vehicle gap is small. By combining predicted TTC ($\hat{t}_{TTC}$) and predicted time headway of ego vehicle ($\hat{h}_{ev}$). The cost of rear-end collision risk (J^{c}_{risk}) for each action can be evaluated below as in Equation (5):

$$J^{c}_{risk} = \begin{cases} \begin{pmatrix} [1 - tanh(\hat{t}_{TTC}/H_{min})] + \\ [1 - tanh(\hat{h}_{ev}/H_{min})] \end{pmatrix} /2, & \hat{t}_{TTC} \geq 0 \\ [1 - tanh(\hat{h}_{ev}/H_{min})]/2, & \hat{t}_{TTC} < 0 \end{cases} \tag{5}$$

$$\hat{h}_{ev} = [d_{gap} + \Delta\hat{d}_{gap}]/(v_f + \Delta\hat{v}_f) \tag{6}$$

where $\hat{h}_{ev}$ is the predicted time headway of ego vehicle, H_{min} is the minimum safe time headway based on the 3-second rule [44], if ego vehicle is a truck, H_{min} = 4 s, based on the statistic result in [16]. With a larger H_{min}, the safety cost of truck is higher than car.

To consider the merging urgency of a ramp vehicle, the distance to the end of merging area should be added to the risk value of ramp vehicle, as shown in Equation (7). The closer to the end of merging area, the higher cost vehicles should pay. The risk of merging (J^{m}_{risk}) can be formulated as follows:

$$J^{m}_{risk} = [1 - tanh(\hat{h}_{rv}/H_{min})]/2 \tag{7}$$

$$\hat{h}_{rv} = [d_r + \Delta\hat{d}_r]/(v_e + \Delta\hat{v}_e) \tag{8}$$

where $\hat{h}_{rv}$ is the predicted remaining time headway to the end of merging area for ramp vehicle; d_r is the remaining distance of merging area; $\Delta\hat{d}_r$ is the predicted remaining distance; v_e and $\Delta\hat{v}_e$ are the current speed and predicted speed change of ego vehicle, respectively. To summarize, the risk for mainline vehicles and ramp vehicles used in this study can be expressed in Equation (9):

$$J_{risk} = \begin{cases} J^{c}_{risk}, & Mainline\ vehicles \\ (J^{c}_{risk} + J^{m}_{risk})/2, & Ramp\ vehicles \end{cases} \tag{9}$$

As aforementioned, saving travel time may be another target for players in the game. In addition, if we only consider safety in the cost function, players in the game will incline to more conservative behaviors (e.g., encouraged to be followers or to decelerate), resulting in unnecessary congestion along with the upstream. Therefore, adding a mobility term would help CAVs find the balance between safety and speed, and improve the traffic efficiency at the same time. Both mainline and ramp CAVs are encouraged to take actions with minimum speed drop, if the safety performance is not compromised. As shown in Equation (10), the mobility cost function puts more penalties on deceleration maneuvers. The term $tanh(\Delta\hat{v}_e/v_e)$ is more sensitive when the speed of ego vehicle is slow, as this algorithm cares more about mobility for low-speed driving, but safety for high-speed driving.

$$J_{mobility} = 1 - tanh(\Delta\hat{v}_e/v_e) \tag{10}$$

where $\Delta\hat{v}_e$ is the ego vehicle speed difference of either being a follower or a leader in the game, compared to the current speed.

To improve the driving comfort, hard braking and drastic acceleration are penalized with a cost as shown in Equation (11).

$$J_{comfort} = \begin{cases} \hat{a}/acc_{lim}\ , & \hat{a} \geq 0 \\ \hat{a}/dec_{lim}\ , & \hat{a} < 0 \end{cases} \tag{11}$$

where $\hat{a}$ is the acceleration of ego vehicle in next time step, $acc_{lim} > 0$ is the acceleration limit, and $dec_{lim} < 0$ is the deceleration limit. Moreover, the different limits for truck and car are designed based on [45].

To summarize, the overall cost ($\bar{J}$) is:

$$\bar{J} = \alpha_1 J_{risk} + \alpha_2 J_{mobility} + \alpha_3 J_{comfort} \tag{12}$$

where $\alpha_i \geq 0, i = 1,2,3$, is the weight for each term in the cost function, and $\sum_i \alpha_i = 1$. In this study, we choose $\alpha_1 = 0.4$, $\alpha_2 = 0.4$ and $\alpha_3 = 0.2$.

Non-Cooperative Game and Cooperative Game

After estimating the cost of each player's action, the optimal result can be obtained from a decision table, which depends on the game type, either non-cooperative or cooperative game. In this study, a game can be only initiated by an equipped CAV. Once the CAV recognizes a potential conflict, it sends out a cooperation invitation and keeps waiting for a reply. If the CAV receives no response from the other party, a non-cooperative two-player game will be formed. In this type of game, the CAV will adopt a selfish strategy, since it can only rely on the information from the radar system and optimize its own cost. The decision table of the non-cooperative game is shown in **Table 1**.

To avoid collision, ego vehicle will not choose to play the same role with its competitor at the same time. Therefore, the costs for both players being the leaders or followers simultaneously are set to be infinity (or very large values). At each time step, ego vehicle will choose the option with the minimum expected cost, as described in Equation (13).

$$Action = \min_{actions} \{\bar{J}_{lead}, \bar{J}_{follow}\} \tag{13}$$

Table 1. Decision Table for Non-Cooperative Two-Person Game

	Competitor		
	Role	Leader	Follower
Ego vehicle	Leader	∞	$\bar{J}_{lead}$
	Follower	$\bar{J}_{follow}$	∞

The game between two CAVs would be a cooperative one, where players can make decisions together. The decision table of the cooperative game between two CAVs is shown in **Table 2**. Unlike a non-cooperative algorithm which can provide the optimal solution only for ego vehicle regardless of system conditions, a cooperative game can optimize the total cost (based on the information shared via vehicle-to-vehicle communication) for both CAVs.

Table 2. Decision Table for Cooperative Two-Person Game

	Partner		
Ego vehicle	Role	Leader	Follower
	Leader	∞	$\bar{J}^{ego}_{lead} + \bar{J}^{p}_{follow}$
	Follower	$\bar{J}^{ego}_{follow} + \bar{J}^{p}_{lead}$	∞

As described in Equation (14), both CAVs will take the action to achieve the system optimum.

$$Action = \min_{actions} \{\bar{J}^{ego}_{follow} + \bar{J}^{p}_{lead},\ \bar{J}^{ego}_{lead} + \bar{J}^{p}_{follow}\} \quad (14)$$

where $\bar{J}^{ego}_{follow}$ and $\bar{J}^{ego}_{lead}$ are the costs of being a follower or a leader for ego vehicle, respectively; $\bar{J}^{p}_{follow}$ and $\bar{J}^{p}_{lead}$ are the costs of being a follower or a leader for its partner, respectively.

Speed Guidance using Head-Up-Display

Although the proposed algorithm is originally designed for CAVs, it can be generalized for human-driven connected vehicles. We design Advisory Speed Assistance (ASA) system to provide speed suggestions for the human driver, which is displayed using a head-up-display (HUD). With the merging sequence determined and the acceleration calculated by Equation (3), the speed suggestion displayed on the HUD can be computed as

$$v_i(t + \delta t) = v_i(t) + a_{ref}(t + \delta t) \cdot \delta t \quad (15)$$

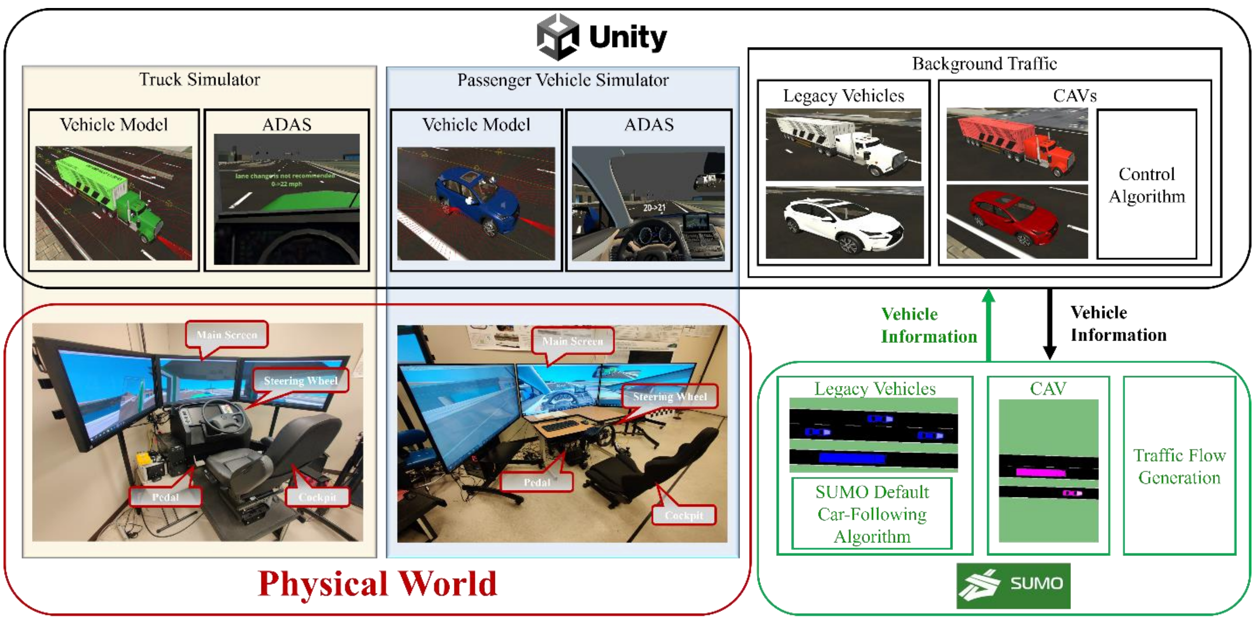
where $v_i(t + \delta t)$ is the advisory speed shown to the driver, and $v_i(t)$ is the current speed of the ego vehicle.

Once the advisory speed $v_i(t + \delta t)$ is computed by Equation (15) and displayed to the driver by the aforementioned HUD design, the driver of the controlled on-ramp vehicle needs to input executions to Unity to track that advisory speed in the longitudinal direction while keeping the vehicle at the center of the current lane.

Development of multi-human-in-the-loop (MHuiL) co-simulation platform

The purpose of building the platform is to create a multi-driver simulation platform to evaluate interaction between a heavy-duty truck and a passenger vehicle under a mixed traffic scenario through MHuiL simulation. Figure shows the overall system architecture of the multi-driver co-simulation platform. From the software aspect, we select 1) Unity which is a game engine suitable for modeling and visualizing the surrounding environment of ego vehicle, and providing high-fidelity (on-board) sensor information; and 2) SUMO which is a microscopic simulator providing realistic traffic flow under various congestion levels and penetration rates of connected and automated vehicles (CAVs). From the hardware aspect, we set up brake, throttle, and steering wheel as the human-machine interface to collect human driving behavior inputs as shown in Figure 4. Overall Architecture of the Multi-driver Co-simulation Platform.

Figure 4. Overall Architecture of the Multi-driver Co-simulation Platform



To integrate SUMO and Unity, we develop a python script called *Edge Gateway* to handle the data exchange and information synchronization. As shown in

Figure , each component can share locally processed data with others via the Edge Gateway. The Edge Gateway uses asynchronous processes to avoid simulator blocking that may introduce Frame Per Second (FPS) drops. There are several User Datagram Protocol (UDP) server and client threads handling the data exchange with queues. With the Edge Gateway, FPS performance is improved to provide a better human driver experience in Unity. In addition, the Edge Gateway also provides the possibility to extend the entire platform for incorporating other simulators or even real-world vehicles.

Figure 5. Components of the Multi-driver Co-simulation Platform

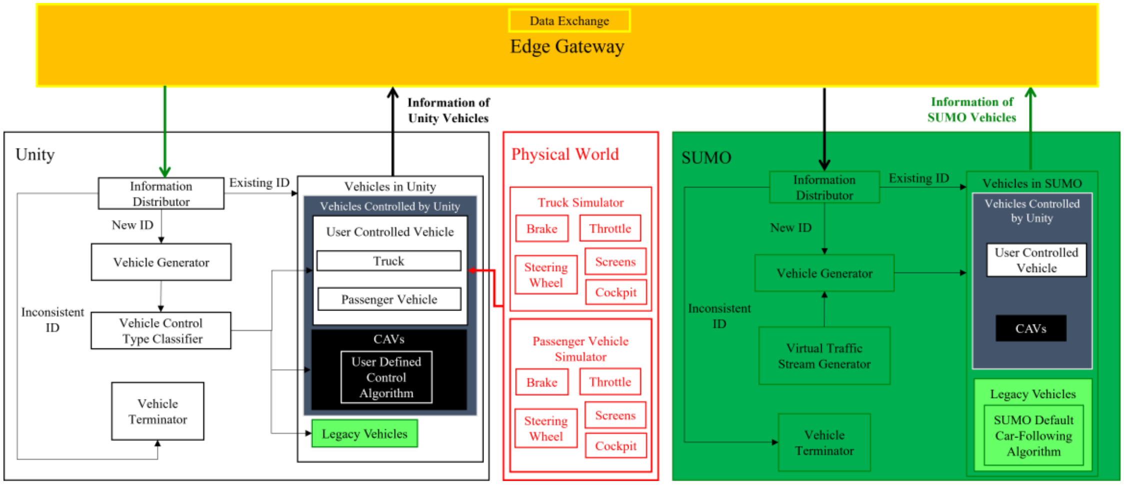
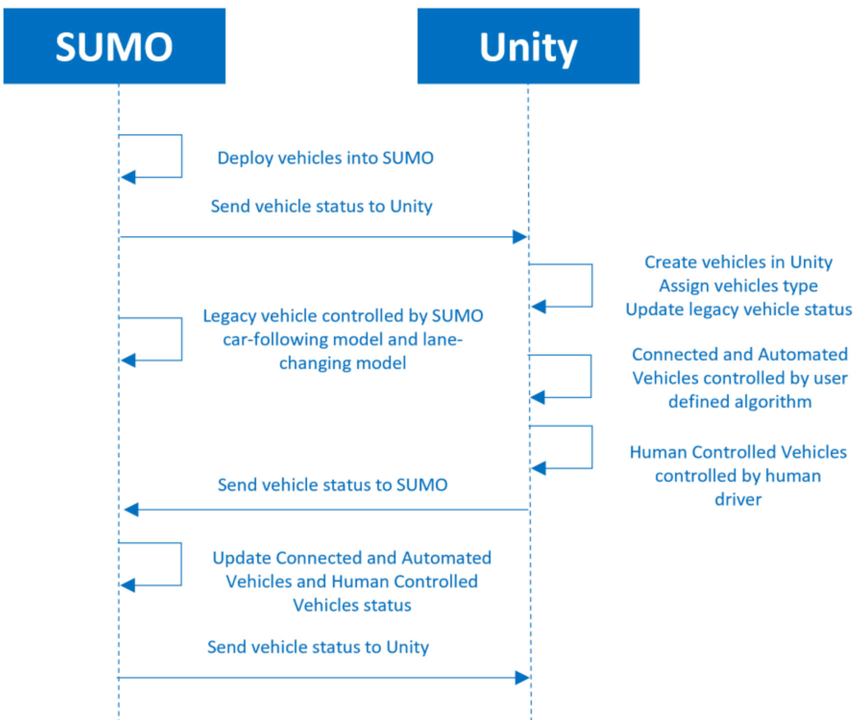


Figure 6. Sequence Diagram for Information Synchronization



Since the connection and information synchronization is critical to the integrated platform, we detail the chronological sequence of each executed component. As shown in Figure , SUMO generates vehicles based on a predefined route file. Keeping the route file unchanged can reproduce the same traffic flow. With this feature, we can easily control the environment variables in different tests. For the first time when vehicles' statuses are shared with Unity, predefined 3D prefabs can be created based on vehicle types selected by the *Vehicle Control Type Classifier*. Although vehicles are initially all governed by SUMO, Unity may take over the control once some (or even all) of them are assigned to be CAVs or human-controlled vehicles (HCVs). After creating vehicles in both simulators and determining the type of each vehicle, legacy vehicles will be controlled based on SUMO models and their information will be shared in real-time to Unity through the Edge Gateway. Then, Unity will update the corresponding vehicles' statuses. Likewise, the Unity controlled vehicles, including HCV and CAVs, will share their states, such as velocities and positions to SUMO, to ensure statuses of the corresponding vehicles on different simulators are synchronized.

Vehicle Modeling

To create a realistic mixed traffic flow for cooperative ADAS designing through the HuiL, we define three general vehicle types participating in the simulation, which are legacy vehicle, CAV, and HCV. Background traffic including CAVs and legacy vehicles, as shown in Figure . At the simulator level, they can also be classified into two types: Unity controlled vehicles and SUMO controlled vehicles. Unity controlled vehicles include CAVs and HCV. In terms of vehicle model type, we define two categories: trucks and passenger vehicles.

Figure 7. Background Traffic Models: (a) CAVs (red), (b) Legacy vehicles (white)

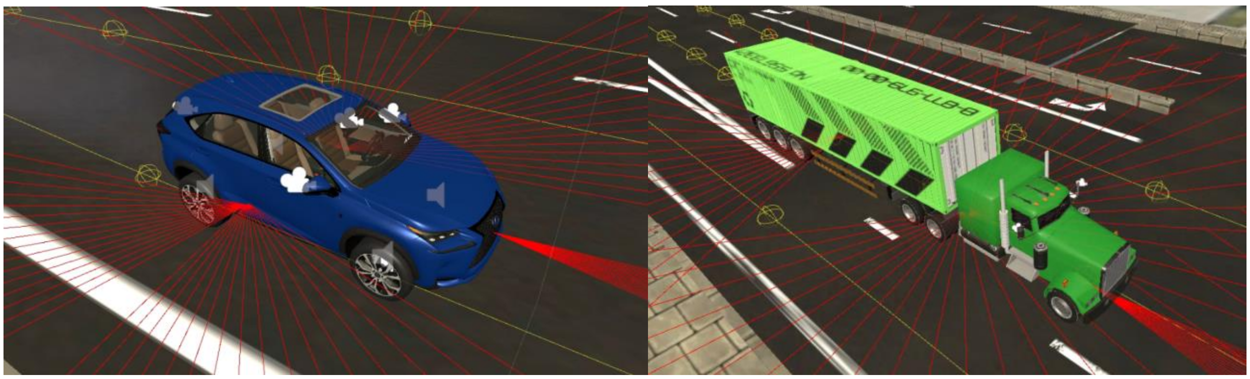
(a)

(b)

Specifically, legacy vehicles are totally controlled by SUMO via car-following and lane-changing models. SUMO can deploy these legacy vehicles based on a predefined route file and remove them when they reach the end of the trip. In terms of CAVs, they are controlled by user-defined algorithms coding in Unity. As aforementioned, we use the game theory-based algorithm for CAVs control in this study. In addition, the on-board sensors (e.g., camera, radar, LiDAR, GPS) equipped on CAVs can provide realistic data formats such as point clouds and images.

As a key player in the HuiL simulation, we create two human control vehicle (HCV) model in Unity which are human-controlled truck (HCT) model and human-controlled passenger vehicle (HCPV) model as shown in Figure .

Figure 8. Human Control Vehicle Models: (a) HCPV (blue), and (b) HCT (green)



(a) (b)

From the exterior perspective, HCPV is a four-wheel conventional vehicle and HCT is a heavy-duty truck composed of a truck head and a trailer which is connected to the head with a hinge. Both HCV models have two side mirrors and one rearview mirror, as shown in Figure , providing an immersive driving experience for CADS evaluation. In addition, on the windshield of HCV models (i.e., HUD), ADAS suggestions are displayed in front of the drivers. From the control perspective, HCPV enables human control by using a script provided by *Vehicle Physics Pro* (VPP) which is an advanced vehicle simulation kit for Unity that supports efficient, realistic, and accurate vehicle simulation. On the other hand, HCT is controlled by a script from "*Universal Vehicle Controller*" which provides a highly customizable vehicle behavior control.

Figure 9. Driver Perspective of HCV Models: (a) HCPV, and (b) HCT.

(a) (b)

Pacific Southwest Region UTC
University Transportation Center

Hardware Setup

As shown in Figure , the multi-driver co-simulation platform supports two human-machine interface setups: HCPV simulator and HCT simulator. For each simulator, it has one cockpit and three screens (including two side mirrors), extending the driver's field of view about surrounding vehicles. Besides, it is necessary to have brake and throttle pedals allowing the longitudinal control and steering wheel allowing the latitudinal control. Specifically, HCPV simulator is composed of Logitech gaming steering wheel and pedal set which is supported by Unity joystick input interface. On the other hand, HCT hardware is connect to a USB4 encoder and outputs can be decoded into voltage values which can be conveniently converted to control signals of HCT.

Simulation Environment Construction

Taking advantage of the game engine, we construct a high-quality simulation environment in Unity including the network, infrastructure, and buildings, based on a real-world map. In addition, to facilitate the lateral and longitudinal control of CAVs, we create a set of waypoints for each lane in the network.

In general, there are two ways to construct the simulation environment in both Unity and SUMO:

- SUMO provides a tool called *NETCONVERT* which can convert the *OpenStreetMap* (OSM) file into a 2D SUMO network file. Once we have the 2D map, we can build the 3D map in Unity accordingly.
- According to the 3D network in Unity, we create the same 2D map in SUMO, ensuring two maps to have the same reference point for position synchronization between two simulators.

Typically, the second approach is more laborious than the first one, however, OSM does not always provide high-quality map compared to the real world, which may affect the fidelity of simulation. In this paper, we choose the second approach as the OSM model of the study area could not well reflect the real-world network geometry and situations. To address this issue, we select a real road network and create the virtual environment from the scratch. The network covers the stretch from the intersection of Chicago Avenue to the intersection of Iowa Avenue along Columbia Avenue in Riverside, California.

Traffic Flow Generation

To generate realistic traffic flows, we apply the *Poisson* distribution. In other words, given a time period T, the probability of the time interval between two departures is larger than T can be calculated as follows:

$$P(T \leq t) = 1 - e^{-\lambda t} \quad (16)$$

where λ represents the departure rate within a period of time. Based on this model, the time interval between departures is exponentially distributed and can be calculated as follows:

$$t = \frac{\log(1-T)}{-\alpha} \quad (17)$$

where α represents the traffic volume in vehicle per hour.

According to the equation, we can assign departure time for each vehicle and save them as a route file containing each individual vehicle's properties, such as vehicle ID, departure time, and predefined route. SUMO can spawn vehicles based on this route file and Unity can create vehicles with the same properties after the first time it receives their information from SUMO. Once vehicles are created in Unity, the platform may determine if each individual vehicle is a CAV, legacy vehicle, or the HCV based on the logic described in the following section.

Vehicle Control Type Classifier

To assign the role for each vehicle, we create a vehicle control type classifier (see

Figure), which determines a vehicle in background traffic to be a CAV or a legacy vehicle based on a random number generator. It uses the vehicle ID as a seed number to generate a random number between 0 and 1. Compare the random number p and a predefined threshold P that represents the penetration rate:

$$Vehicle\ Type = \begin{cases} CAV & ,p<P \\ Legacy\ Vehicle & ,p \geq P \end{cases} \quad (18)$$

Besides this, HCV is selected by a predefined vehicle ID. Once Unity receives the HCV ID from SUMO, the HCV prefab is deployed in Unity with the same position and velocity as in SUMO. Specifically, HCT and HCPV have different vehicle IDs to differentiate the corresponding model spawn in Unity.

Traffic Flow Terminator

The traffic flow terminator mainly fulfills two functions: a) to remove vehicles that already finish their trips; and b) to remove vehicles that are not synchronized in both simulators.

With the traffic flow generation, we can spawn vehicles and have them driven in both SUMO and Unity. After vehicles finish their trips or reach the end of their destinations, SUMO can automatically remove those SUMO controlled vehicles. Similarly, Unity is able to delete those vehicles controlled by Unity after they reach their last waypoints. However, whenever a simulator removes a vehicle in its local side, the other simulator may not remove the same vehicle simultaneously. To resolve the problem, we check the vehicle ID list sent from the other simulator every time step, compare it with the vehicle ID list in the target simulator, and then remove those vehicles only show up in one of the simulators.

MHuiL Co-Simulation Study

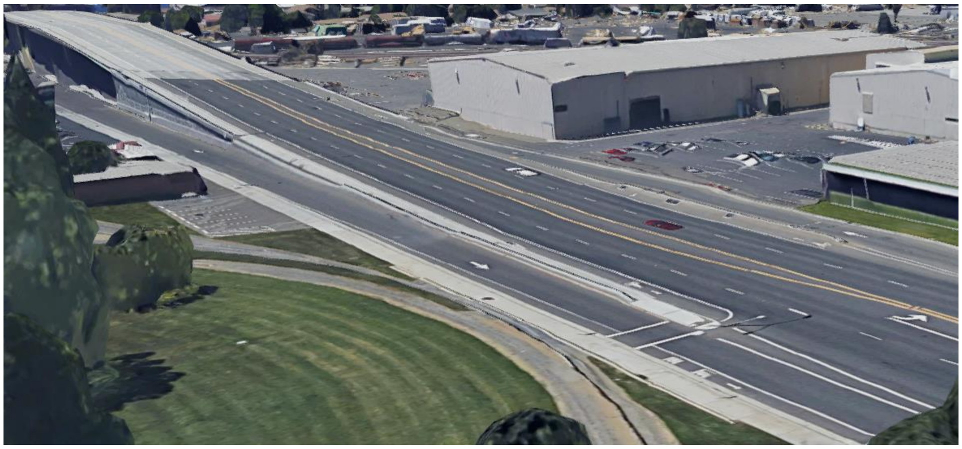
In this project, we aim to study the merging interaction between the truck driver and the car driver. Using the MHuiL platform, we are able to provide immersive driving with mixed traffic environments and replay the merging scenario for a fair comparison, where the driving operations and vehicle states of both truck and car drivers can be captured every time step.

Simulation network environment

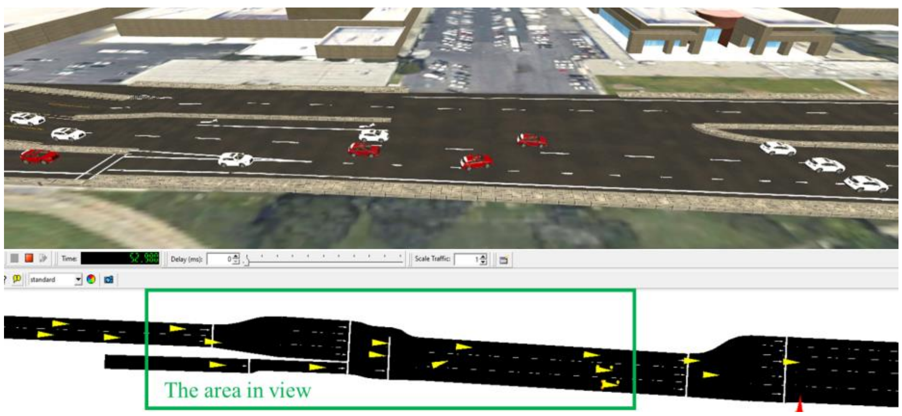
As previously described, a real-world traffic network is coded in the simulation, spanning from Chicago Avenue to Iowa Avenue along Columbia Avenue in Riverside, California. It consists of a single-lane on-ramp and a segment of multi-lane mainline (Google Maps view is shown in **Figure** (a)). The integrated simulation environment is shown in **Figure** (b), where the upper part with terrain details is the Unity environment, and the lower part is the corresponding SUMO network.

Figure 10. The Integrated Simulation based on a Real-world Ramp Merging Area in Riverside, CA: (a) View from Google Maps at the real-world ramp; (b) User interface of the Unity-SUMO

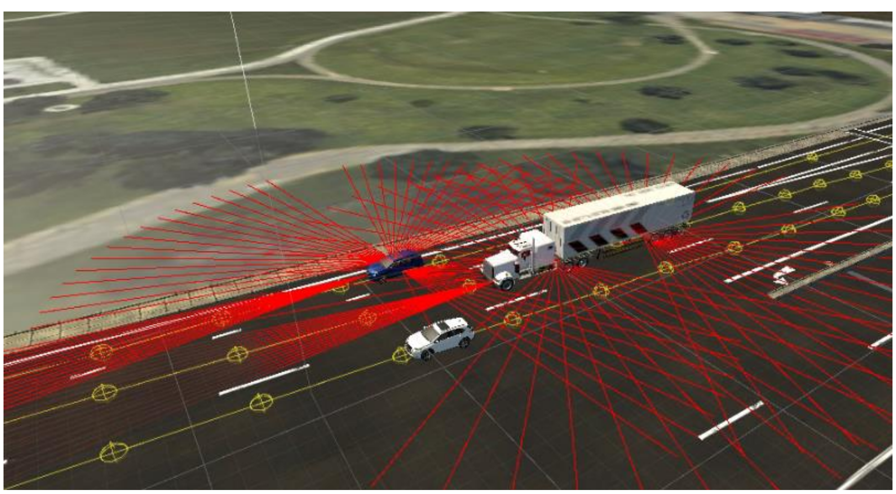
co-simulation platform; (c) Speed guidance enabled car (in blue) and truck with the radar system and a legacy car (in white)



(a)



(b)



(c)

Pacific Southwest Region UTC
University Transportation Center

Table 3. Vehicular Parameters and Simulation Setup

Vehicle type	**Car**	**Truck**
Initial speed (adaptive to traffic)	ramp: 15 m/s; mainline: 20 m/s	
Minimum inter-vehicle gap	2.5 m	5 m
Acceleration range	-5 ~ 3 $\mathrm{m/s^2}$	-4 ~ 1.3 $\mathrm{m/s^2}$
Desired speed (speed limit)	10 m/s	
Desired minimum time headway	1 s	1.5s
Vehicle length	5 m	12 m
Emergency braking	-9 $\mathrm{m/s^2}$	-7 $\mathrm{m/s^2}$
Initial distance to merging point	ramp: 250 m; mainline: 440 m	
Congestion level (v/c ratio)	0.60	
Traffic demand (veh/hr)	2400	
Average duration	Ramp vehicle: 30s; mainline vehicle: 48 s	

As shown in **Figure** (c), there are three types of vehicles in the network, a background car in white, a human-driven car in blue, and a human-driven truck. When the speed guidance system of a truck or car is enabled, the radar system of the vehicle will be activated. The red rays spread from both car and truck indicate the detection range of its onboard radar system, where the long-range radar in the front has a 150m detection range and 18-degree field of view, while short-range radars on the side have a 25m detection range and 70-degree field of view. The characteristics and performance parameters of this onboard radar system are selected based on off-the-shelf radar sensors [46].

To generate a more realistic mixed traffic environment and carry out a fair evaluation, the parameters are carefully selected as shown in **Table 3**. And more details are discussed below.

Study scenarios

We invite 7 volunteers with real-world driving experience to participate in this multi-human-in-the-loop (MHuiL) simulation. To have a fair comparison, we assign the same person to drive the truck simulator for all runs, while the volunteers only drive the car simulator. All volunteers have the chance to drive both the mainline car scenarios and the ramp car scenarios, and for each role, they will experience non-guided and speed-guided cases. For the speed-guided case, the drivers are suggested to try their best to follow the speed guidance during the simulation, so that the ego vehicle can perform the cooperative merging maneuvers more smoothly compared to the scenario when no speed guidance is provided. In the non-guided case, to make sure the truck and car encounter each other for creating merging conflicts, preceding vehicles are set for both drivers, and the car following behavior will take the two drivers to the merging area nearly at a close time.

At the very beginning, to make the user familiar with the driving simulator, each volunteer drives the vehicle on the simulator for two trial runs of non-guided and speed-guided cases, respectively. Note that the volunteer is randomly asked to drive either the ramp vehicle or the mainline vehicle. Additionally, only one volunteer at a time is allowed to enter the room of the simulator. Therefore, the volunteer will not have any prior knowledge regarding the traffic scenario.

In this study, we explore different vehicle interactions of scenario combinations and conditions, considering the vehicle types, road types, and with/without speed guidance. As a result, each volunteer performs eight runs, as shown in Table 4.

Results and analyses

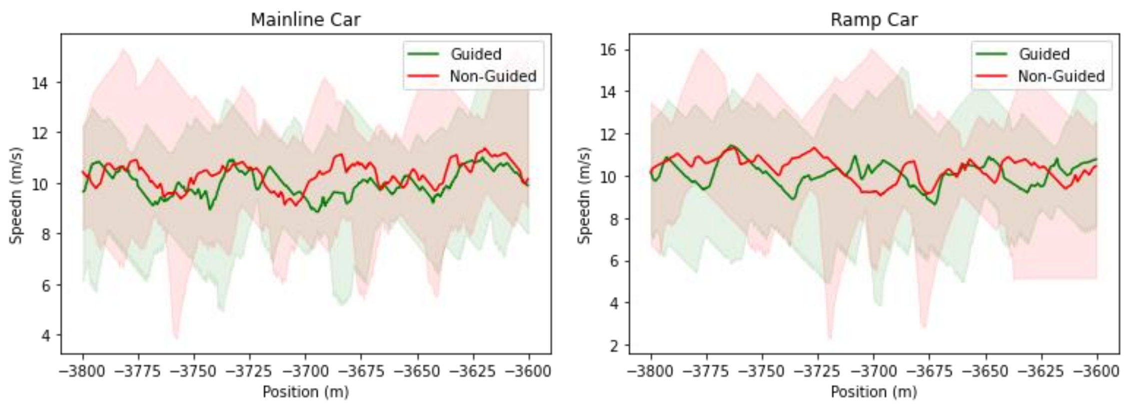
Each volunteer conducts eight simulation trips, so a total of 56 runs are recorded for evaluation. The speed profile of the car driver is visualized using maximum, minimum, and median values in **Figure 2**, where only the segments of the speed profile in the merging zone are shown. The merging zone is defined to start from where the truck and car can first observe each other and end at where the double solid line begins. The peak and valley values (i.e., highest or lowest speed) of guided (in green) cars are smaller than those of non-guided (in red) cars. The main difference is observed in the behavior of the mainline car in **Figure 2**(a). Between -3725 m to -3650m (the midpoint of the merging area), guided mainline cars usually slow down and create gaps in advance for ramp trucks, while non-guided mainline cars usually speed up to surpass ramp trucks.

Table 4. Experiment Scenarios

Scenario	Mainline Vehicle	Guidance	Ramp Vehicle	Guidance
1a	H-Truck*	No	H-Car	No
1b	H-Truck	Yes	H-Car	Yes
1c	H-Truck	No	H-Car	Yes
1d	H-Truck	Yes	H-Car	No
2a	H-Car	No	H-Truck	No
2b	H-Car	Yes	H-Truck	Yes
2c	H-Car	No	H-Truck	Yes
2d	H-Car	Yes	H-Truck	No

* Truck simulator is always driven by the same person, and car simulator is handling by other 7 volunteers in turn.

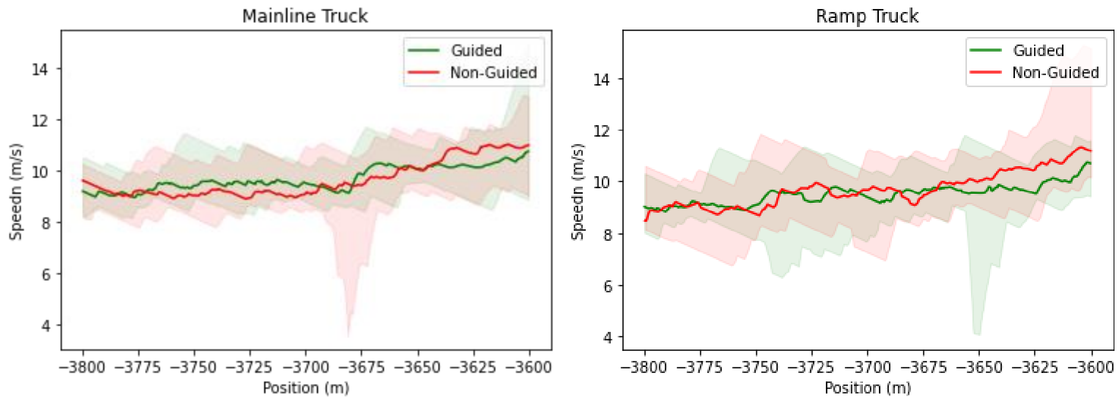
Figure 2. Speed Profiles of Passenger Cars: a) mainline car, b) ramp car



(a) (b)

Figure 3 Speed Profile of Trucks: a) mainline truck, b) ramp truck

Pacific Southwest Region UTC
University Transportation Center



(a) (b)

Compared to the profile of the car, the peak speed values are less observed in the speed profile of trucks, as shown in **Figure 3**. Also, the speed profiles of non-guided and guided trucks are similar, especially for the mainline scenarios. A hypothesis is that the truck driver participates all of the 56 simulations and becomes more familiar with the merging scenario and learns the pattern of the speed guidance over the runs. When the truck runs on the mainline, the driver can easily spot the ramp vehicle due to the road grade. Therefore, it is easier for the truck driver to adjust its speed in advance and the difference of the speed profiles between guided and non-guided scenario is not significant. But if the truck is traveling on the ramp, the driver's field of view gets occluded (again due to the road grade). With the guidance, the driver has to change the speed at the last moment. In this case, the speed guidance would be much more effective.

To quantify the performance of the proposed algorithm, we compare 28 speed guided trips with 28 baseline non-guided trips, regarding merging safety and smoothness. Specifically, the safety is evaluated by the median of minimum time headway to its preceding vehicle, while the smoothness is evaluated by the median of speed standard variance, during the merging process.

As presented in **Table 5**, speed guidance increases the safety of both mainline and ramp car drivers. The largest improvement is observed for mainline car drivers, the median minimum time headway increases by 57%. With the speed guidance, mainline cars generate enough gap for ramp vehicles to prevent dangerous cut-ins. The gap changes for trucks are limited.

Regarding the speed smoothness, both ramp trucks and mainline cars benefit from the speed guidance the most, with an improvement of 19% and 17%, respectively. With the guidance, the ramp truck driver does not need to adjust the speed too often, since the mobility of the car is better than the truck, and the mainline cars make the most adjustment (e.g., creating gaps) during the interaction. Moreover, the algorithm guides the mainline cars to adjust the speed gradually, so the mainline cars do not need to face a difficult situation and perform hard brakes (e.g., for dangerous cut-ins) or drastic acceleration. This will help save energy and enhance driving comfort as well.

Table 5. Algorithm Performance Evaluation

Safety Evaluation - the median of minimum time headway (s)			
	Non-Guided	**Guided**	**Difference**
Mainline Truck	1.11	1.11	0%
Ramp Truck	1.04	1.03	-1%
Mainline Car	0.75	1.76	+57%
Ramp Car	0.73	0.74	+4%
Smoothness Evaluation - the median of speed std. (m/s)			
	Non-Guided	**Guided**	**Difference**
Mainline Truck	1.14	1.19	+4%
Ramp Truck	0.92	0.77	-19%
Mainline Car	1.68	1.43	-17%
Ramp Car	1.68	1.66	-1%

It should be noted that human tracking errors are inevitably introduced into ADAS, since drivers cannot track advisory speed profiles perfectly. As shown in

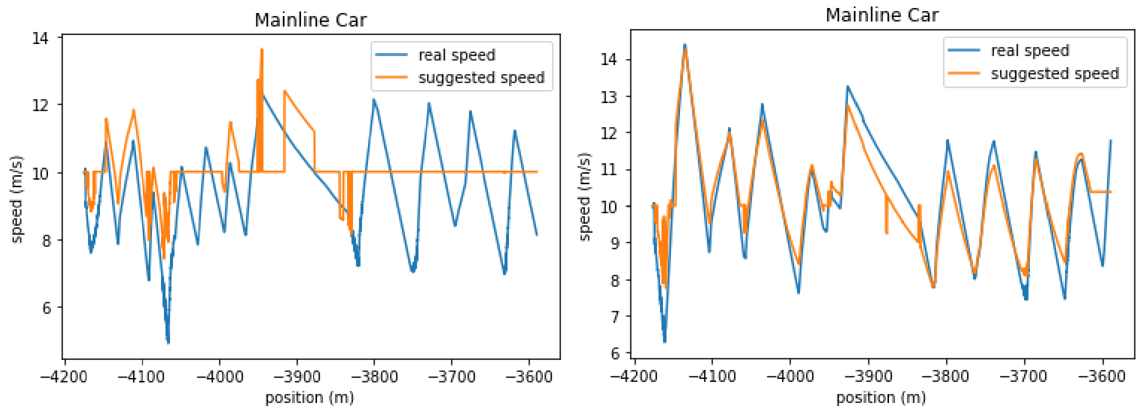
Figure 4, two typical guided mainline trips are selected to conduct an illustrative comparison for bad (in

Figure 4(a)) and good speed tracking (in

Figure 4(b)). The well-planned speed suggestion is ignored by the driver in

Figure 4(a), and the guided trip is similar to a non-guided trip. It is concluded by one of our previous studies that the speed tracking errors may account for as high as 12% degradation in system performance (e.g., energy consumption) of the original ADAS design **Error! Reference source not found.**. Therefore, the design of speed guidance can be improved in many ways, such as the update rate and the interface design (e.g., using a mechanical or digital speedometer).

Figure 4. Speed Suggestion Tracking Performance: (a) speed tracking with large error (b) good speed tracking



(a) (b)

Conclusions and Future Work

In this project, we developed a vehicle-to-everything (V2X) based decentralized cooperative ramp merging system application for a multimodal (e.g., cars, trucks) and mixed traffic (including both connected and non-connected vehicles) environment. In addition, we developed a multi-human-in-the-loop (MHuiL) simulation platform that integrates SUMO traffic simulator with two game engine-based driving simulators. This platform allows us to collect driving behaviors from two drivers (with and without speed guidance) simultaneously, facilitating research on modeling and validation of interactions between drivers in an immersive simulation environment.

Results from the case study (with 7 volunteers who drove the passenger car simulator) have shown that the proposed algorithm is effective for the entire system in terms of safety (e.g., headway) and speed smoothness, for both scenarios including on-ramp and mainline driving where the drivers were either provided or not provided with speed guidance by the developed cooperative ramp merging application. For example, with the speed guidance, the median minimum time headway for the yielding passenger car on the mainline increases by 57%, while the headway change for the involved truck is trivial (0 – 1%). With respect to the speed smoothing effect, both trucks and cars can benefit from the speed guidance the most in the scenario where the truck is traveling on the mainline and the car is driven on the ramp, with the reduction of 19% and 17%, respectively, in the sense of speed variations.

In the future, we will explore more robust cooperative ramp merging algorithms in the presence of sensor noises (such as Global Navigation Satellite System or GNSS errors) and considering personalized driving behavior/preference. In addition, we will further utilize the developed MHuiL simulation platform to retrieve more driving behavior data related to interaction modeling under different scenarios. Besides, we will add a more realistic communication module (via co-simulation with network simulators, e.g., NS-3) to mimic the effects of imperfect wireless communications (e.g., packet loss, communication delays). As for the speed guidance delivery, we will further explore its impacts on the driver's performance (such as tracking errors, distraction) and keep improving the aspect of user friendliness from both parameter selection (e.g., information update rate) and human-machine interface design (e.g., using audio signals or digital speedometer). Regarding the experiment design, more random scenarios will be generated to avoid the test subject getting used to the same scenario and expecting the future situations. This may bias the study results on the effectiveness of ADAS.

References

[1] FHWA. Ramp Metering: A Proven, Cost-Effective Operational Strategy—A Primer. Final Report. FHWA-HOP-14-020, 2014

[2] Zhao, Z. G. Wu, Z. Wang, et al. Optimal Control-Based Eco-Ramp Merging System for Connected and Automated Vehicles. IEEE Intelligent Vehicles Symposium, 2020

[3] Rios-Torres, J. and A. A. Malikopoulos. A Survey on the Coordination of Connected and Automated Vehicles at Intersections and Merging at Highway On-Ramps. IEEE Transactions on Intelligent Transportation Systems, 18(5), 2017, pp. 1066 – 1077

[4] Wang, Z., G. Wu, K. Boriboonsomsin, M. J. Barth, K. Han, B. Kim, and P. Tiwari. Cooperative Ramp Merging System: Agent-Based Modeling and Simulation Using Game Engine. SAE International Journal of Connected and Automated Vehicles, 2(2), 2019, pp. 115 – 128

[5] Wu, G., Z. Zhao, Z. Wang, K. Boriboonsomsin, and M. J. Barth. Development of Eco-Friendly Ramp Control for Connected and Automated Electric Vehicles. Research Report, National Center for Sustainable Transportation, 2020

[6] Wang, Z., Y. Bian, S.E. Shladover, G. Wu, S.E. Li, and M.J. Barth. A Survey on Cooperative Longitudinal Motion Control of Multiple Connected and Automated Vehicles. IEEE Intelligent Transportation Systems Magazine, 12(1), 2020, pp. 4-24.

[7] Zhou, Y., M. E. Cholette, A. Bhaskar and E. Chung. Optimal Vehicle Trajectory Planning with Control Constraints and Recursive Implementation for Automated On-Ramp Merging. IEEE Transactions on Intelligent Transportation Systems, 20(9), 2019, pp. 3409-3420.

[8] Rios-Torres, J. and A. A. Malikopoulos. Automated and cooperative vehicle merging at highway on-ramps. IEEE Transactions on Intelligent Transportation Systems, 18(4), 2016, pp.780-789.

[9] Rios-Torres, J. and A. A. Malikopoulos. Impact of partial penetrations of connected and automated vehicles on fuel consumption and traffic flow. IEEE Transactions on Intelligent Vehicles, 3(4), 2018, pp.453-462.

[10] Min, H., Y. Fang, X. Wu, G. Wu, and X. Zhao. On-ramp merging strategy for connected and automated vehicles based on complete information static game. Journal of Traffic and Transportation Engineering (English Edition), 8(4), 2021, pp.582-595.

[11] Huang, T. and Z. Sun. Cooperative ramp merging for mixed traffic with connected automated vehicles and human-operated vehicles. IFAC-PapersOnLine, 52(24), 2019, pp.76-81.

[12] Liao, X., et al. Game theory-based ramp merging for mixed traffic with unity-sumo co-simulation. IEEE Transactions on Systems, Man, and Cybernetics: Systems, 52(9), 2021, pp. 5746-5757.

[13] Duret, A., M. Wang, and A. Ladino, A. A hierarchical approach for splitting truck platoons near network discontinuities. Transportation Research Part B: Methodological, 2020. Volume:132, pp.285-302.

[14] Zhao, C., L. Li, J. Li, K. Li and Z. Li,. The impact of truck platoons on the traffic dynamics around off-ramp regions. IEEE Access, 9, 2021, pp.57010-57019.

[15] Lu, Y., K. Cheng, Y. Zhang, X. Chen and Y. Zou. Analysis of lane-change conflict between cars and trucks at merging section using UAV video data. arXiv preprint arXiv:2201.07881, 2022.

[16] Nobukawa, S. Bao, D. J. LeBlanc, D. Zhao, H. Peng and C. S. Pan. Gap acceptance during lane changes by large-truck drivers—An image-based analysis. IEEE transactions on intelligent transportation systems, 17(3), 2015, pp.772-781.

[17] Krajzewicz, D., Erdmann, J., Behrisch, M., & Bieker, L. (2012). Recent development and applications of SUMO-Simulation of Urban MObility. International journal on advances in systems and measurements, 5(3&4).

[18] Casas, J., Ferrer, J. L., Garcia, D., Perarnau, J., & Torday, A. (2010). Traffic simulation with aimsun. In Fundamentals of traffic simulation (pp. 173-232). Springer, New York, NY.

[19] Fellendorf, M., & Vortisch, P. (2010). Microscopic traffic flow simulator VISSIM. In Fundamentals of traffic simulation (pp. 63-93). Springer, New York, NY.

[20] Rong, G., Shin, B. H., Tabatabaee, H., Lu, Q., Lemke, S., Možeiko, M., ... & Kim, S. (2020, September). Lgsvl simulator: A high fidelity simulator for autonomous driving. In 2020 IEEE 23rd International conference on intelligent transportation systems (ITSC) (pp. 1-6). IEEE.

[21] Dosovitskiy, A., Ros, G., Codevilla, F., Lopez, A., & Koltun, V. (2017, October). CARLA: An open urban driving simulator. In Conference on robot learning (pp. 1-16). PMLR.

[22] Koenig, N., & Howard, A. (2004, September). Design and use paradigms for gazebo, an open-source multi-robot simulator. In 2004 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)(IEEE Cat. No. 04CH37566) (Vol. 3, pp. 2149-2154). IEEE.

[23] Benekohal, R. F., & Treiterer, J. (1988). CARSIM: Car-following model for simulation of traffic in normal and stop-and-go conditions. Transportation research record, 1194, 99-111.

[24] Tideman, M., & Van Noort, M. (2013, June). A simulation tool suite for developing connected vehicle systems. In 2013 IEEE Intelligent Vehicles Symposium (IV) (pp. 713-718). IEEE.

[25] Chen, X. Zhao, H. Liu, G. Ren, Y. Zhang, and X. Liu, Assessing the Influence of Adverse Weather on Traffic Flow Characteristics Using a Driving Simulator and VISSIM, Jan. 2019.

[26] M. Althoff, M. Koschi and S. Manzinger, CommonRoad: Composable benchmarks for motion planning on roads, 2017 IEEE Intelligent Vehicles Symposium (IV), 2017, pp. 719-726, doi: 10.1109/IVS.2017.7995802.

[27] Klischat, Moritz et al. Coupling SUMO with a Motion Planning Framework for Automated Vehicles. SUMO (2019).

[28] Pariota, L., Coppola, A., Di Costanzo, L., Di Vico, A., Andolfi, A., D'Aniello, C., & Bifulco, G. N. (2020). Integrating tools for an effective testing of connected and automated vehicles technologies. IET Intelligent Transport Systems, 14(9), 1025-1033.

[29] Zhang, E., & Masoud, N. (2020, September). V2xsim: A v2x simulator for connected and automated vehicle environment simulation. In 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC) (pp. 1-6). IEEE.

[30] Biurrun, L. Serrano, and C. O. Monreal, Microscopic Driver-Centric Simulator: Linking Unity3D and SUMO, Advances in Intelligent Systems and Computing, Feb. 2017, doi: 10.1007/978-3-319-56535-4_83.

[31] Xu, R., Guo, Y., Han, X., Xia, X., Xiang, H., & Ma, J. (2021, September). OpenCDA: an open cooperative driving automation framework integrated with co-simulation. In 2021 IEEE International Intelligent Transportation Systems Conference (ITSC) (pp. 1155-1162). IEEE.

[32] G. Schirner, D. Erdogmus, K. Chowdhury, and T. Padir, “The Future of Human-in-the-Loop Cyber-Physical Systems”, Computer, vol. 46, Jan. 2013.

[33] J Bo-Chiuan Chen and Huei Peng, Differential-Braking-Based Rollover Prevention for Sport Utility Vehicles with Human-in-the-loop Evaluations, Vehicle System Dynamics, vol. 36, pp. 359-389, 2001, DOI: 10.1076/vesd.36.4.359.3546.

[34] W. Li, D. Sadigh, S. S. Sastry, and S. A. Seshia. Synthesis for Human-in-the-Loop Control Systems, In: Ábrahám E., Havelund K. (eds) Tools and Algorithms for the Construction and Analysis of Systems. TACAS 2014. Lecture Notes in Computer Science, vol 8413, 2014. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-642-54862-8_40.

[35] Hussein, A., Díaz-Álvarez, A., Armingol, J. M., & Olaverri-Monreal, C. (2018, November). 3DCoAutoSim: Simulator for cooperative ADAS and automated vehicles. In 2018 21st International Conference on Intelligent Transportation Systems (ITSC) (pp. 3014-3019). IEEE.

[36] Gao, L., Bai, W., Leary, R., Varadarajan, K., & Brennan, S. (2021). ROS Integration of External Vehicle Motion Simulations with an AIMSUN Traffic Simulator as a Tool to Assess CAV Impacts on Traffic. IFAC-PapersOnLine, 54(20), 870-875.

[37] Zhao, Y., Wagh, A., Hou, Y., Hulme, K., Qiao, C., & Sadek, A. W. (2016). Integrated traffic-driving-networking simulator for the design of connected vehicle applications: eco-signal case study. Journal of Intelligent Transportation Systems, 20(1), 75-87.

[38] Zhao, X., Liao, X., Wang, Z., Wu, G., Barth, M., Han, K., & Tiwari, P. (2022). Co-simulation platform for modeling and evaluating connected and automated vehicles and human behavior in mixed traffic. SAE Int. J. of CAV, 5(4).

[39] Maag, C., Muhlbacher, D., Mark, C., & Kruger, H. P. (2012). Studying effects of advanced driver assistance systems (ADAS) on individual and group level using multi-driver simulation. IEEE Intelligent Transportation Systems Magazine, 4(3), 45-54.

[40] Xu, J., & Lin, Y. (2014). A Networked Multi-Driver Simulation Platform for Interactive Driving Behavior Study. In The 15th Driving Simulation Conference (Europe) (pp. 4-5).

[41] Yue, L., Abdel-Aty, M., & Wang, Z. (2022). Investigation of Merge Strategies at Ramp Area in Connected Vehicle Environment Based on Multi-Driver Simulator System (No. UCF-2-Y4).

[42] Wang, Z., G. Wu, M. Barth. Developing a Distributed Consensus-Based Cooperative Adaptive Cruise Control (CACC) System for Heterogeneous Vehicles with Predecessor Following Topology. Journal of Advanced Transportation, 2017

[43] Kang, K. and Rakha, H.A. A repeated game freeway lane changing model. Sensors, 20(6), 2020, p.1554.

[44] National Safety Council. Maintaining a Safe Following Distance While Driving. Nsc.org, 2020. Accessed on: Aug. 30, 2021. [Online]. Available: https://www.nsc.org/driveithome/pointers-for-parents/pointers-for-parents-lessons/lesson-48.

[45] Vehicle type parameter defaults - sumo documentation. [Online]. Available: https://sumo.dlr.de/docs/Vehicle_Type_Parameter_Defaults.html. [Accessed: 31-Aug-2022].

[46] de Ponte Müller, F. Survey on ranging sensors and cooperative techniques for relative positioning of vehicles. Sensors, 17(2), 2017, p.271.

Data Management Plan

Products of Research

In this study, the research team collected driving behavior data using the developed multi-human-in-the-loop (MHuiL) simulation platform. Describe the data that were collected and used for the study.

Data Format and Content

The trajectories of each driver and the surrounding vehicles are saved in a txt file using JSON format. Each trajectory of surrounding vehicles consists of position (x,y,z) and speed with a corresponding timestamp, while the trajectory of the human control vehicles consists of position, velocity, suggested speed (if provided), steering angle, the pedal force of braking or acceleration, distance to the preceding vehicle, and the timestamps.

Data Access and Sharing

The data will be available publicly via the UC Riverside instance of Dryad. The hyperlink of dataset: https://datadryad.org/stash/share/qJQBQxqtfpZXIpfvzkmm2Y_BkhSdB66jDgw5tK3s4io. And the unique digital object identifier (DOI): doi:10.6086/D18X0V.

Reuse and Redistribution

The PIs and Regents of the University of California will hold the intellectual property right for the data created. The reuse and redistribution of dataset need appropriate citation of this report. Apart from that, the PIs do not see any other legal requirements that need to be addressed.