2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 23-27, 2022, Kyoto, Japan

Development of a Research Testbed for Cooperative Driving in Mixed Traffic of Human-driven and Autonomous Vehicles

Jiaxing Lu, Ryan Stracener, Weihua Sheng, He Bai, Sanzida Hossain

***Abstract*— This paper presents a cooperative driving testbed based on vehicle-to-vehicle (V2V) communication, which can be used for research in intelligent transportation systems, such as collision avoidance in mixed traffic of both human-driven vehicles and autonomous vehicles. To achieve the goal, an intelligent copilot is developed. The copilot can share the data regarding vehicle status, intention, etc, with other nearby vehicles through V2V communication. Several case studies are conducted to validate the proposed testbed and evaluate the performances of cooperative driving. When dangerous situations occur, the copilot solves the collision avoidance problem using Mixed Integer Programming (MIP), which either provides control commands to the autonomous vehicle, or advises the human driver to take action. Experimental results show that the safety and stability of the involved vehicles have been significantly enhanced. This cooperative driving testbed can be used by researchers to develop and test cooperative driving algorithms before they are deployed in real vehicles.**

***Index Terms*— V2V communication, cooperative driving, driving simulator, ROS**

I. Introduction

The global autonomous car market is expected to grow from 5.6 billion in 2018 to 60 billion in 2030 [1]. While more and more autonomous vehicles are emerging on the roads, many people have doubt about autonomous vehicles' safety and vulnerability. It is expected that a transportation system with mixed traffic of both traditional human-driven vehicles and autonomous vehicles will exist for a long time. Therefore, it is highly desirable to develop cooperative driving approaches that can help improve transportation safety through vehicle to vehicle (V2V) communication.

Although real vehicle-based testing is critical in the development of cooperative driving algorithms, such tests are still costly and dangerous. On the other hand, simulation plays a significant role in accelerating and supplementing the real world testing [2]. According to the annual Autonomous Mileage Report published by the California Department of Motor Vehicles, Waymo has logged 15 billion miles in simulation on vehicle-related experiment and research [2]. Driving simulators have been used for a long-time in vehicle algorithm development and experiment. However, to the best of our knowledge, there is no simulation testbed for cooperative driving based on V2V communication [3]. This paper aims to develop such a cooperative driving testbed using driving simulators. The Carnetsoft simulator [4] is adopted for this purpose which offers various interfaces to support V2V communication and vehicle control. Connected with a Logitech G29 driving kit that includes a steering wheel, a shifter, and pedals, this driving simulator offers realistic driving experience.

In order to facilitate cooperative driving, we propose to develop a copilot, which is attached to the dashboard of a vehicle (human-driven or autonomous), similar to a GPS unit. The copilot has sensors such as a camera, a microphone, an accelerometer, as well as a speaker and a touch screen. The copilot connects with other copilots in nearby vehicles and exchange data with each other. Equipped with machine learning algorithms, the copilots can recognize driver intentions, detect distracted driving behavior (for human-driven vehicles only) and participate in cooperative driving with nearby vehicles.

The major contribution of this paper is three-fold. First, this work develops the first cost-effective experimental testbed that consists of two driving simulators equipped with a copilot and can be used for research in cooperative driving involving multiple vehicles, including both human-driven and autonomous vehicles. Second, this work designs and implements an innovative copilot and its associated software architecture, which facilitates various intelligent applications including cooperative driving. Third, we conduct several case studies to validate and assess the proposed testbed in a small-scale cooperative driving setting.

This paper is organized as follows. The related work is presented in section II. Sections III introduces the overall system architecture of the copilot. Section IV describes the cooperative driving testbed. Section V presents the experimental setup and results for collision avoidance. The final section provides the conclusion and the future work.

II. Related Work

Cooperative driving is a vehicle control method that relies on V2V communication to achieve coordinated vehicle movement so that the driving safety can be improved for all involved vehicles. In recent years, there has been increasing interest in cooperative driving in the intelligent transportation system (ITS) research community. Khayatian *et al.* proposed an advanced and generic version of Responsibility-Sensitive Safety (RSS) rules for Connected Autonomous Vehicles

Jiaxing Lu, Ryan Stracener and Weihua Sheng (corresponding author) are with the School of Electrical and Computer Engineering, Oklahoma State University, Stillwater, OK 74078, USA, email: jiaxing.lu@okstate.edu, ryan.stracener@okstate.edu, weihua.sheng@okstate.edu.

He Bai and Sanzida Hossain are with the School of Mechanical and Aerospace Engineering, Oklahoma State University, Stillwater, OK 74078, USA, email: he.bai@okstate.edu, sanzida.hossain@okstate.edu.

This work is supported by the National Science Foundation (NSF) Grants CISE/IIS 1910933, EHR/DUE 1928711 and CPS 2212582.

2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS) | 978-1-6654-7927-1/22/$31.00 ©2022 IEEE | DOI: 10.1109/IROS47612.2022.9981436

(CAVs) that can be applied to any driving scenario [5]. Wang *et al.* presented a cooperative ramp merging system and demonstrated an online feedforward/feedback longitudinal controller for CAVs, in order to reduce the average travel time, energy consumption, and pollutant emissions [6]. A new cooperative driving strategy for connected and automated vehicles at unsignalized intersections was presented in [7], which is based on tree representation of the solution space by combining Monte Carlo tree search and some heuristic rules to find a globally optimal passing order within a very short time. Xie *et al.* proposed two cooperative driving strategies for connected vehicles that move in heterogeneous traffic of regular vehicles and connected vehicles, in order to stabilize the traffic flow [8]. Liu *et al.* proposed a two-step distributed model predictive control approach for providing collision-free properties in cooperative driving between connected autonomous vehicles [9].

Advanced driver-assistance systems (ADAS) are technologies that assist human-drivers in order to improve driving safety [13]. A lot of research has been conducted to enhance the effectiveness and functionality of ADAS. For example, Cueva *et al.* developed an ADAS based on computer vision to detect drowsiness of a human driver, thereby avoiding potential traffic accidents [10]. Divakarla *et al.* proposed a cognitive ADAS for level 4 autonomous vehicles which achieved 23% energy economy increase compared to human-driven vehicles [11]. Lin *et al.* developed a video-based lane-departure and blind-spot detection algorithm based on machine learning [12]. However, existing ADAS only functions in individual vehicles. There lacks an ADAS that can assist drivers in a cooperative driving setting. There has been growing interest in human-machine interface for vehicle driving. Walch *et al.* proposed a user interface for drivers to switch between driving and monitoring effectively [14]. Zimmermann *et al.* proposed a theory-driven interaction and user interface concept for a cooperative lane-change scenario in a highly automated driving environment [15]. Li *et al.* proposed a novel learning-based human-machine cooperative driving scheme with active collision avoidance capacity using deep reinforcement learning [16]. Improving human-in-the-loop decision making is becoming a meaningful challenge for developing vehicle-related algorithms and controlling systems [17].

Apparently, conducting real world tests involving multiple vehicles is very expensive and risky. Therefore it would be highly desirable to develop realistic simulation testbeds for cooperative driving. However, according to [3], there is no existing cost-effective research testbed that simulates cooperative driving involving multiple vehicles, not to mention in mixed traffic of both human-driven and autonomous vehicles.

III. System Architecture

A. Copilot Hardware Architecture

The design of the copilot is shown in Fig. 1. It has two embedded computers which share the computational tasks. Raspberry Pi 4B is an excellent platform for implementing user interface functions while Jetson Nano is an excellent computation engine for machine learning tasks. In the copilot, Jetson Nano is responsible for driver distraction detection, environmental understanding and road condition analysis, etc. Dividing the computational tasks between these two computers improves the performance and makes it easier to debug, test and maintain the copilot system. Two cameras, one facing the road in front of the vehicle and the other facing the driver, are used to collect visual data of the traffic and the driver, respectively.

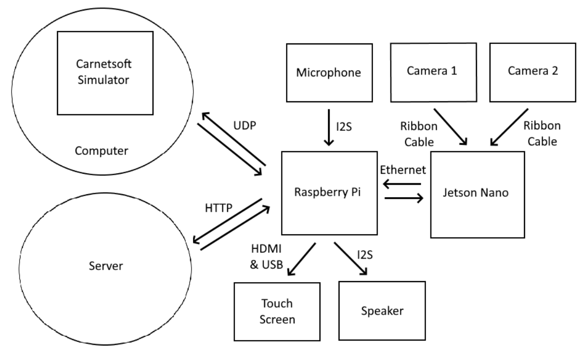

Fig. 1: Copilot hardware architecture.

The Carnetsoft driving simulator provides an open environment for application development such as autonomous driving and human driving. It has 1) a road map database, which consists of pre-designed maps and is also open for adding user-designed maps; 2) a script language, which allows to create various driving scenarios; 3) an interface to other devices, which communicates data through a UDP protocol; 4) user-friendly tools to configure the simulator, through which we can customize light and shadow effects, displays, resolutions, and driving traffic rules. This simulator makes it possible to test vehicle-control algorithms without entailing physical damage to real vehicles or test participants.

B. Software Architecture of Copilot

The software architecture of the copilot is shown in Fig. 2. The software on Jetson Nano has mutliple ROS packages to interface with two cameras and is responsible for publishing vision data to other packages, detecting driver's status, and recognizing and detecting objects in front of the vehicle. The software architecture also includes a computational engine called Gurobi solver [19] which is used to solve the optimization problem of cooperative driving.

The copilot software consists of a backend and a frontend. The backend has several ROS packages for different tasks, including communicating with the driving simulator and other vehicles based on ROS topics, logging vehicle data and copilot data, interacting with machine learning modules and the Gurobi solver, and implementing cooperative driving control.

The frontend is the interface between the user and copilot. The screen has an animated face, which is capable of changing expressions depending on the commands sent from the backend. The purpose of the frontend is to give the

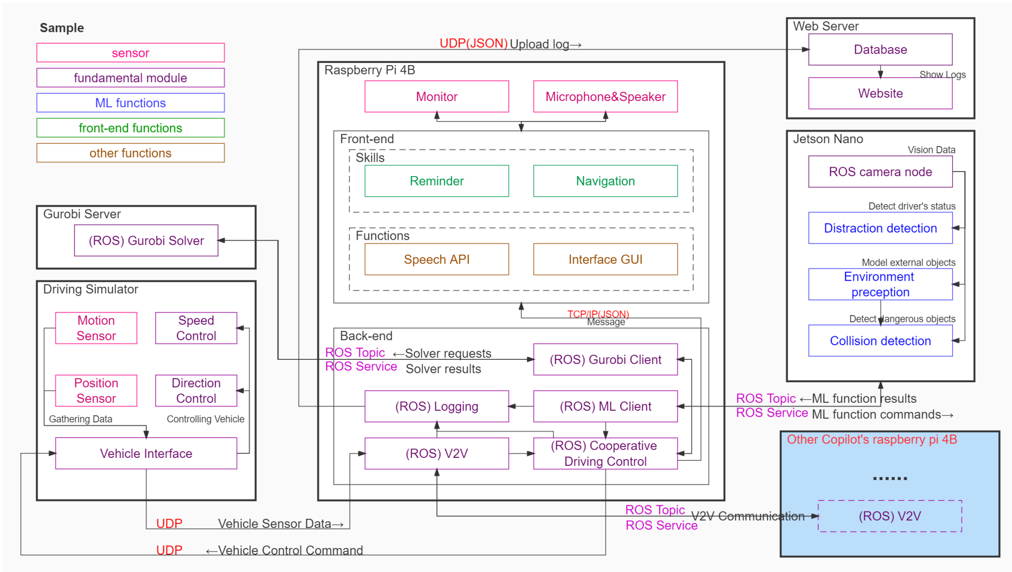

Fig. 2: Copilot software architecture.

Fig. 3: The prototype of the copilot.

user feedback during driving, warn the user of hazardous conditions, and give access to convenient features such as setting reminders. For autonomous vehicles, the copilot only has the backend, serving as a control agent to the rest of the autonomous driving software. Communicating with the backend via a TCP connection, the frontend's functions could easily be migrated to a smartphone or other mobile devices.

IV. Cooperative Driving Testbed

The cooperative driving testbed consists of two human-driven vehicles each assisted by a copilot (human driving assistant) and multiple autonomous vehicles each assisted by a copilot (autonomous driving control agent). The copilot is deployed next to the steering wheel, as shown in Fig. 3. Each human-driven vehicle is controlled by a user through a Logitech G29 driving setup, which includes a steering wheel, floor pedals, and a shifter. Each autonomous vehicle is controlled by a separate thread in the Carnetsoft simulator. The vehicle data is sent by the simulator to their corresponding copilots. On the other hand, vehicle control commands are sent from the copilots to the corresponding vehicles in the simulator that specify the speed and direction of involved vehicles.

A. V2V Communication

There are two UDP connections between the driving simulator and the backend of each copilot, as shown in Fig. 4. The ROS V2V module, a UDP server, periodically receives vehicle data sent by the driving simulator which in turn receives commands sent by the cooperative driving control module in the backend. Each vehicle, whether autonomous or human-driven, has both connections established.

The data sent from the backend to the frontend consists of reminder messages, ego vehicle status and adjacent vehicle status, which are in JSON format. The backend analyzes the vehicle data sent by the driving simulator and formats it as a ROS message to be published, which includes five parts: 1) vehicle ID; 2) speed; 3) vehicle intention; 4) longitudinal position; 5) lateral position.

B. Data Logging on Server

A Flask webserver is used to collect and analyze the data generated by the simulated vehicles. The database is an H2 embedded database which stores session data relevant to the experiment. Once an experiment is concluded, the data is exported as a CSV file to be analyzed by tools such as Microsoft Excel and MATLAB.

V. Case Study

We conducted experiments to demonstrate the effectiveness of the proposed cooperative driving testbed. First, we formulated a cooperative driving problem as a mixed integer optimization problem. Second, we tested the performance of cooperative driving in two lane-merging scenarios: 1) the copilot warns the human driver to avoid collision when an adjacent autonomous vehicle is trying to merge into the human-driven vehicle's lane, and 2) the copilot controls an autonomous vehicle to avoid collision when the human driver

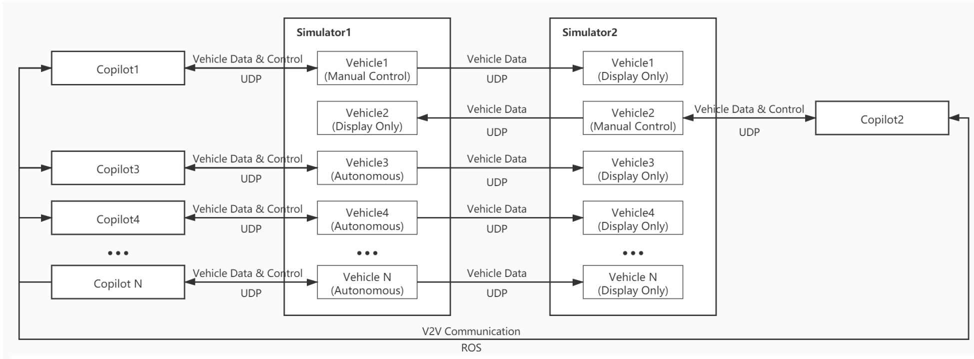

Fig. 4: The architecture of the cooperative driving testbed.

is trying to merge into the autonomous vehicle's lane. In each scenario, experimental results with or without copilot advising were collected. Only one driving simulator was involved in these case studies of cooperative driving, where a human-driven vehicle was the ego vehicle and autonomous vehicles were adjacent vehicles.

A. Cooperative Driving as Mixed Integer Optimization

After connecting with the driving simulator, the backend will publish vehicle data as a ROS message mentioned above to all vehicles once it receives the corresponding vehicle data from the driving simulator. In the meantime, each backend subscribes to data published by other vehicles. Cooperative driving between the two vehicles is modeled as an mixed integer optimization problem, which will be solved on a central server through the Gurobi solver. When the message from another vehicle indicates a merging intention, the copilot will send the parameters to the Gurobi solver, including velocity and position for both vehicles. The solver calculates the control command of vehicle speed to establish the safe distance for the merging. For the copilot on the autonomous vehicle, the command with the updated speed will be sent every 0.8 sec after the merging starts. On the human-driven vehicle the copilot's frontend will use voice to remind the driver to speed up or brake.

We formulate a mixed-integer program (MIP), which is solved using the Gurobi solver to coordinate the behaviors of the vehicles under various operating conditions for coordinated lane merging. The dynamic model of the autonomous vehicle is given by

$$x^r_{k+1} = A_r x^r_k + B_r u^r_k, \tag{1}$$

where k is the time index with 0.8 sec interval, $x^r_k \in R^2$ represents the longitudinal position and velocity of the autonomous vehicle, A_r and B_r are matrices of suitable dimensions, and u^r_k is the autonomous vehicle acceleration.

For the human-driven vehicle, the dynamics of $x^h_k \in \mathbb{R}^2$ is described by

$$x^h_{k+1} = \begin{cases} A_h x^h_k + B_h u^a_k, & \text{follow the advising} \\ A_h x^h_k + B_h u^h_k, & \text{otherwise} \end{cases}, \tag{2}$$

where u^a_k is the suggested acceleration from the copilot, and u^h_k is the acceleration under human control. In the simulations, we use the double integrator dynamics for both (A_r, B_r) and (A_h, B_h).

Let $x_k = [(x^r_k)^T \ (x^h_k)^T]^T$ denote the states of both vehicles, $x^B = \{0, 1\}$ denote the event of the human-driven vehicle following the advising (1) or not (0), and $z^u_k = x^B_k(u^a_k - u^h_k)$. Following the formulation of mixed-logical dynamical systems [18], we obtain the dynamics of x_k as

$$x_{k+1} = \begin{pmatrix} A_r & 0 \\ 0 & A_h \end{pmatrix} x_k + \begin{pmatrix} B_r & 0 & 0 \\ 0 & B_h & B_h \end{pmatrix} \begin{pmatrix} u^r_k \\ u^h_k \\ z^u_k \end{pmatrix}. \tag{3}$$

Let M_u and m_u be the upper and lower limit of the acceleration difference $(u^a_k - u^h_k)$, respectively. Then z^u_k satisfies

$$z^u_k \le M_u x^B_k, z^u_k \ge m_u x^B_k \tag{4}$$

$$z^u_k \le (u^a_k - u^h_k) - m_u(1 - x^B_k) \tag{5}$$

$$z^u_k \ge (u^a_k - u^h_k) - M_u(1 - x^B_k). \tag{6}$$

To avoid the collision during the lane merging, we would like the longitudinal distance between the two vehicles to be greater than a threshold $d_r > 0$ at some time step k, i.e.,

$$\left| x^r_{k,1} - x^h_{k,1} \right| \ge d_r, \tag{7}$$

where $x_{k,1}$ denotes the position state. To make (7) amenable for optimization, we introduce two binary variable $b_{1,k}$ and $b_{2,k}$, where $b_{1,k} = 1$ when $x^r_{k,1} - x^h_{k,1} \ge d_r$ and $b_{2,k} = 1$ when $x^r_{k,1} - x^h_{k,1} \le -d_r$. We then convert (7) to

$$x^r_{k,1} - x^h_{k,1} \le -d_r + \bar{M} b_{1,k} \tag{8}$$

$$x^r_{k,1} - x^h_{k,1} \ge d_r - \bar{M} b_{2,k} \tag{9}$$

where $\bar{M}$ is sufficiently large. When $b_{1,k} = 0$ and $b_{2,k} = 1$, it follows from (8)–(9) that $x^r_{k,1} - x^h_{k,1} \le -d_r$, and $x^r_{k,1} - x^h_{k,1} \ge d_r - \bar{M}$ that holds for a sufficiently large $\bar{M}$. Similarly, when $b_{1,k} = 1$ and $b_{2,k} = 0$, we have $x^r_{k,1} - x^h_{k,1} \ge d_r$ and $x^r_{k,1} - x^h_{k,1} \le -d_r + \bar{M}$. We also note that it is invalid to have $b_{1,k} = b_{2,k} = 0$. We then introduce a constraint

$$b_{1,k} + b_{2,k} \ge 1. \tag{10}$$

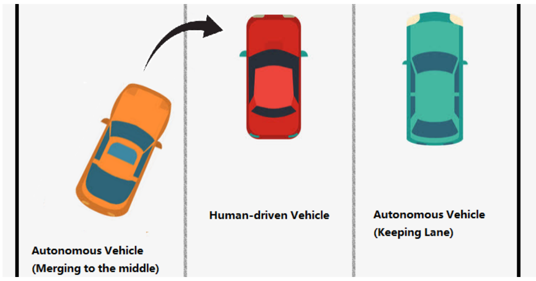

Fig. 5: Experiment Scenario 1.

When $b_{1,k} = b_{2,k} = 1$, whether $|x^r_{k,1} - x^h_{k,1}| \geq d_r$ is undetermined. It then follows that (7) is satisfied at time k if and only if $b_{1,k} + b_{2,k} = 1$.

To reduce the time to reach the condition (7) in the next K time steps, we propose to solve the following MIP at each time step $j = 1, 2, \cdots$.

$$\min_{\substack{u^a_k, u^r_k, z^u_k, b_{1,k}, b_{2,k}, \\ k = j, \cdots, j+K}} \sum_{k=j}^{j+K} (b_{1,k} + b_{2,k}) + w_a (u^a_k)^2 + w_r (u^r_k)^2 \tag{11}$$

subject to $(3), (4) - (6), \ (8) - (10)$,

$$m_u \leq u^a_k \leq M_u, \ m_u \leq u^r_k \leq M_u,$$
$$m \leq x_k \leq M, \ k = j, \cdots, j+K,$$

where w_a and w_r are positive weights. In particular, $m \leq x_k \leq M$ can include any speed and position constraints. At time step j, the Gurobi solver solves (11) to obtain $u^r_k, u^a_k, z^u_k, k = j, \cdots, j+K$. Then u^r_j is implemented for the autonomous vehicle while the copilot of the human vehicle advises u^a_j to the driver. The human input variable u^h_j is predicted based on the current and the previous human-driven vehicle's state.

B. Human Reaction with Copilot Advising

In this experiment, the road has three lanes in one direction. Three vehicles are driving alongside with the same initial speed. As shown in Fig. 5, the rightmost vehicle is an autonomous vehicle driving on the road and keeps its lane the whole time. The vehicle in the middle is the human-driven vehicle (ego vehicle). The leftmost vehicle is in our blind spot and will try to merge into the center lane at a random time after the experiment starts.

Four participants, each with driving experience of over three years and good vision, took a 40-minute test. Each participant completed the test ten times for the given scenario. Five tests were conducted without advising from the copilot, while the other five tests were conducted with the advising from the copilot. For each test, the participant did not have prior knowledge of whether the copilot would remind them. As a control variable, the speed of the autonomous vehicle is set at 20 metre/second. The speed of the human-driven vehicle and the reaction time from when the merging started to when the human driver took action were recorded. The copilot reminds the driver through voice advising, for example, "slow down" and "speed up". Table I shows the comparison of reaction times between with and without the copilot's advising. During the experiment, collision occurred three times when there was no copilot advising. However, for the tests with copilot advising, no collision occurred and the safety distance was overall larger than without copilot advising. An example of the recorded speed history during experiment is shown in Fig. 6. Test example A is the recording of volunteer 2's first test. The driver noticed the autonomous vehicle in blind-spot 2 sec after it started to merge and then braked suddenly, which will be risky if there is a vehicle behind. Test example B is the recording of volunteer 2's last test without copilot advising, where he could notice the merging intention in time and the braking process was more smooth. Test example C shows that, with copilot advising, the driver could notice the merging intention earlier.

TABLE I: Average time of reaction.

Participant	With Copilot	Without Copilot
participant1	$0.88sec$	$1.44sec$
participant2	$0.92sec$	$1.84sec$
participant3	$1.08sec$	$1.56sec$
participant4	$0.96sec$	$2.04sec$

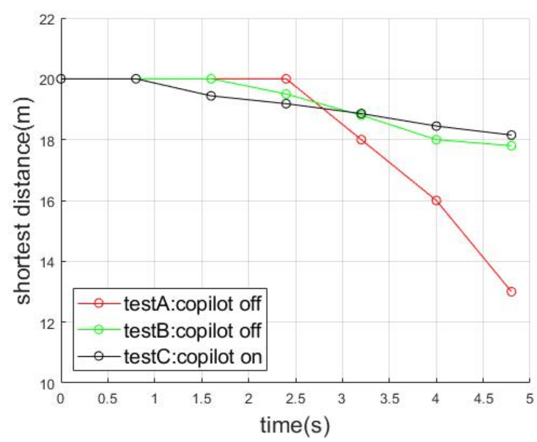

Fig. 6: Examples for recorded speed history of voluteer 2.

From the above results, we can see that with the copilot's advising through cooperative driving, the safety of driving is improved significantly.

C. Copilot's vehicle control to avoid collision

In order to test the copilot's capability in autonomous vehicles, we also conducted an experiment in which a human driver is trying to merge into an autonomous vehicle's lane. The experiment setting is shown in Fig. 7. The human driver tried to merge into the middle lane, where an autonomous vehicle was. For five times of the tests, the copilot of the autonomous vehicle provided the correct control commands to guide the autonomous vehicle to brake and therefore establish the safety distance for the merging. The performance of copilot detecting adjacent vehicle's abnormal activity and controlling its associated vehicle was evaluated. To compare, a test without copilot control was conducted.

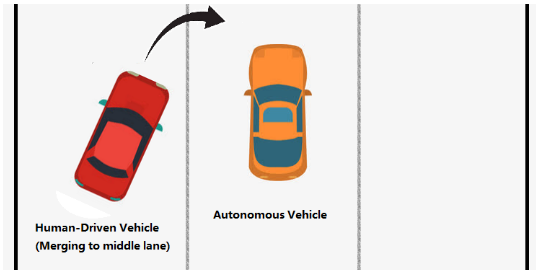

Fig. 7: Experiment Scenario 2.

Fig. 8 shows the curve of the shortest distances between these two involving vehicles. The blue curves are the results of the experiments with copilot controlling vehicles, while the red curves are the results of the experiments without copilot, which stops at the point when the collision happened. From the above results, we can find that with copilot's guidance, an autonomous vehicle can achieve improved safety with nearby vehicles.

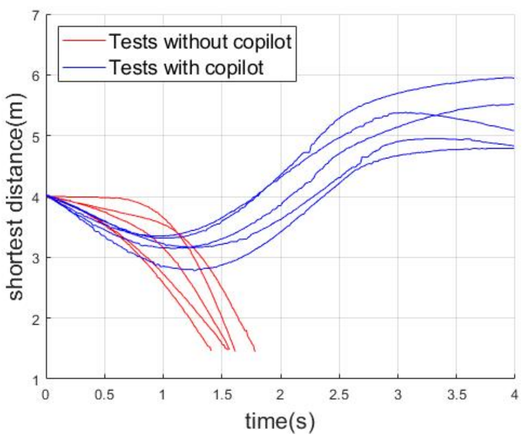

Fig. 8: Autonomous vehicle's shortest distance

VI. Conclusions and Future Work

This paper proposed a cooperative driving testbed based on V2V communication which can be used in ITS research that aims to enhance the safety of mixed traffic consisting of both autonomous and human-driven vehicles. The testbed is based on an intelligent copilot serving as a driving assistant to human drivers and a control agent to autonomous vehicles. Experiment results show that copilot advising helps human drivers to react faster in dangerous situations, while autonomous vehicles effectively avoid collision with other vehicles guided by the copilot. In the future, we will enhance the work in the following aspects. First, we will improve the advising method from audio advising to visual feedback, such as a circular chart on the copilot screen showing the desired speed and the current speed. Second, we will develop more advanced cooperative driving algorithms in more complicated driving scenarios involving more vehicles.

References

[1] Projected size of the global autonomous car market from 2019 to 2023. [Online].Available:https://www.statista.com/statistics/428692/projected-size-of-global-autonomous-vehicle-market-by-vehicle-type/

[2] Off road, but not offline: How simulation helps advance our Waymo Driver. [Online]. Available: http://bit.ly/WaymoBlog

[3] P. Kaur, S. Taghavi, Z. Tian and W. Shi, "A Survey on Simulators for Testing Self-Driving Cars," 2021 Fourth International Conference on Connected and Autonomous Driving (MetroCAD), 2021, pp. 62-70, doi: 10.1109/MetroCAD51599.2021.00018.

[4] Carnetsoft driving simulator for training and research. [Online]. Available: https://cs-driving-simulator.com/

[5] Khayatian, Mohamamd Mehrabian, Mohammadreza Allamsetti, Harshith Wei, Kai Yu, Po Lin, Chung-Wei Shrivastava, Aviral. (2021). Cooperative Driving of Connected Autonomous Vehicles Using Responsibility-Sensitive Safety (RSS) Rules. 10.1145/3450267.3450530.

[6] Wang, Ziran Wu, Guoyuan Boriboonsomsin, Kanok Barth, Matthew Han, Kyungtae Kim, BaekGyu Tiwari, Prashant. (2019). Cooperative Ramp Merging System: Agent-Based Modeling and Simulation Using Game Engine. SAE International Journal of Connected and Automated Vehicles. 2. 1-14. 10.4271/12-02-02-0008.

[7] H. Xu, Y. Zhang, L. Li and W. Li, "Cooperative Driving at Unsignalized Intersections Using Tree Search," in IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 11, pp. 4563-4571, Nov. 2020, doi: 10.1109/TITS.2019.2940641.

[8] Dong-Fan Xie, Yong-Qi Wen, Xiao-Mei Zhao, Xin-Gang Li, Zhengbing He (2020) Cooperative driving strategies of connected vehicles for stabilizing traffic flow, Transportmetrica B: Transport Dynamics, 8:1, 166-181, DOI: 10.1080/21680566.2020.1728590.

[9] P. Liu, A. Kurt and U. Ozguner, "Distributed Model Predictive Control for Cooperative and Flexible Vehicle Platooning," in IEEE Transactions on Control Systems Technology, vol. 27, no. 3, pp. 1115-1128, May 2019, doi: 10.1109/TCST.2018.2808911.

[10] L. D. S. Cueva and J. Cordero, "Advanced Driver Assistance System for the drowsiness detection using facial landmarks," 2020 15th Iberian Conference on Information Systems and Technologies (CISTI), 2020, pp. 1-4, doi: 10.23919/CISTI49556.2020.9140893.

[11] K. P. Divakarla, A. Emadi and S. Razavi, "A Cognitive Advanced Driver Assistance Systems Architecture for Autonomous-Capable Electrified Vehicles," in IEEE Transactions on Transportation Electrification, vol. 5, no. 1, pp. 48-58, March 2019, doi: 10.1109/TTE.2018.2870819.

[12] Y. -S. Lin, W. -C. Chen and T. P. -C. Chen, "Tensor-Centric Processor Architecture for Applications in Advanced Driver Assistance Systems," 2021 International Symposium on VLSI Design, Automation and Test (VLSI-DAT), 2021, pp. 1-3, doi: 10.1109/VLSI-DAT52063.2021.9427310.

[13] C. Wang, Q. Sun, Y. Guo, R. Fu and W. Yuan, "Improving the User Acceptability of Advanced Driver Assistance Systems Based on Different Driving Styles: A Case Study of Lane Change Warning Systems," in IEEE Transactions on Intelligent Transportation Systems, vol. 21, no. 10, pp. 4196-4208, Oct. 2020, doi: 10.1109/TITS.2019.2939188.

[14] Walch, Marcel Sieber, Tobias Hock, Philipp Baumann, Martin Weber, Michael. (2016). Towards Cooperative Driving: Involving the Driver in an Autonomous Vehicle's Decision Making. 10.1145/3003715.3005458.

[15] Zimmermann, Markus Bengler, Klaus. (2013). A multimodal interaction concept for cooperative driving. IEEE Intelligent Vehicles Symposium, Proceedings. 1285-1290. 10.1109/IVS.2013.6629643.

[16] Li, Junxiang Liang, Yao Xu, Xin Cheng, Bang Ren, Junkai. (2020). Deep Reinforcement Learning for Pedestrian Collision Avoidance and Human-Machine Cooperative Driving. Information Sciences. 532. 10.1016/j.ins.2020.03.105.

[17] C. Lam, A. Y. Yang, K. Driggs-Campbell, R. Bajcsy and S. S. Sastry, "Improving human-in-the-loop decision making in multi-mode driver assistance systems using hidden mode stochastic hybrid systems," 2015 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), 2015, pp. 5776-5783, doi: 10.1109/IROS.2015.7354197.

[18] Alberto Bemporad and Manfred Morari, "Control of systems integrating logic, dynamics, and constraints", Automatica, vol. 35, pp. 407–427, 1999.

[19] Gurobi optimization. https://www.gurobi.com. Accessed in Feb. 2022.