2018 21st International Conference on Intelligent Transportation Systems (ITSC)
Maui, Hawaii, USA, November 4-7, 2018

HARDWARE-IN-THE-LOOP TESTING OF CONNECTED AND AUTOMATED VEHICLE APPLICATIONS: A USE CASE FOR COOPERATIVE ADAPTIVE CRUISE CONTROL

Jiaqi Ma, Ph.D.
Assistant Professor
University of Cincinnati
jiaqi.ma@uc.edu

Fang Zhou
Software Engineer
StoneRiver Inc.
fangzhou1227@gmail.com

Zhitong Huang, Ph.D.
Senior Transportation Engineer/Research Analyst
Leidos Inc.
Zhitong.huang@leidos.com

Rachel James
Pathway Student Intern
Federal Highway Administration
Rachel.James@dot.gov

Abstract— **Most existing studies on connected and automated vehicle (CAV) applications apply simulation to evaluate system effectiveness. Model accuracy, limited data for calibration, and simulation assumptions limit the validity of evaluation results. One alternative approach is to use emerging hardware-in-the-loop (HIL) testing methods that allow physical test vehicles to interact with virtual vehicles from traffic simulation models. This provides an evaluation environment that can replicate deployment conditions at early stages of CAV implementation—without incurring excessive costs related to large field tests. In this study, a hardware-in-the-loop (HIL) testing system for CAV applications is proposed. Cooperative adaptive cruise control (CACC) is one of the key CAV applications that has received a lot of attention. This study develops a HIL proof-of-concept testing prototype for CAV applications in a vehicle-to-vehicle (V2V) environment. The contributions of this study include developing a HIL testing architecture for V2V-based CAV applications; implementing the proposed HIL architecture; developing a prototype; and testing CACC as the selected use case to observe HIL system performance. The results of this effort also contribute to a better understanding of CACC string performance.**

Keywords: **cooperative adaptive cruise control, hardware-in-the-loop simulation, vehicle-to-vehicle communication**

1. INTRODUCTION

Connected and automated vehicle (CAV) technologies offer potentially transformative traffic impacts, including significant mobility, safety, and environmental benefits. Numerous modeling studies have been conducted to evaluate these impacts. However, model accuracy and simulation assumptions limit the validity of evaluation results; the lack of field data exacerbates the problem of inaccuracy, leading to improper model calibration. However, large-scale field operational tests are extremely expensive and still cannot support full-scale assessments of the effect that CAVs will have on overall traffic performance. There are also many confounding factors in a real-world field experiment and possible testing scenarios are limited. One approach to overcoming these challenges is to use emerging hardware-in-the-loop (HIL) tools that allow physical test vehicles to interact with virtual vehicles from traffic simulation models; this provides an evaluation environment that can replicate actual conditions at early stages of CAV deployment without incurring excessive costs. Integrating physical CAVs as hardware in a HIL system is a necessary step for CAV application evaluation and achieves many benefits (e.g., greater validity can be achieved by considering the physical vehicle's trajectory in a simulated traffic environment). The possibility of collecting data on CAV applications in various traffic scenarios enables better calibration of modeling tools, which serve as the foundation for large-scale evaluations with corresponding software-in-the-loop (SIL) systems. Using a HIL system in conjunction with physical vehicles also makes it possible to measure performance more accurately; for example, customized fuel meters for fuel consumption evaluation can be used instead of rough estimation models.

Cooperative adaptive cruise control (CACC) is an example of a heavily researched vehicle-to-vehicle application. Previous simulation work has used assumed CACC behavioral models that are not based on any testing data, with only a few exceptions. The previous CACC field testing only demonstrated the feasibility of platooning without deriving insights in CACC string performance based on field data (e.g., [1]). Furthermore, dedicated short range communication (DSRC) may be unreliable during certain conditions, such as during communication congestion or adverse weather. Considering this effect is critical to truly understand a CACC system. A DSRC communication simulation environment can be used to mimic DSRC communication according to different standards protocols (e.g.., SAE J2735, IEEE 1609.x, and IEEE 802.11). This environment is critical for evaluating CAV applications under different external conditions caused by DSRC communication limitations, such as delay, system hack, packet loss, and communication range limit. In addition, there can be a significant delay of radar data and these data are usually quite noisy. This will significantly limit the gap controller of a CACC algorithm and thus impact platoon performance adversely. Hardware-in-the-loop testing platforms provide a suitable environment to test such adverse situations in a safe and controlled manner.

This study develops a HIL proof-of-concept testing prototype for evaluating CAV applications in a vehicle-to-vehicle (V2V) environment. The experiment in this study was piloted at the Federal Highway Administration Turner-Fairbank Highway Research Center (FHWA TFHRC). The system includes a physical CAV, DSRC equipment, a traffic simulator, and a field test track. Ma et al. proposed a similar framework for HIL testing of infrastructure-related applications, such as signalized intersection approach and departure [2]. Note that the goal of the developed HIL prototype is to enable CACC testing with limited resources (i.e., single CAV). The physical vehicle is used in the loop as a piece of hardware and it can fully test the overall performance by taking into account vehicle dynamics, vehicle hardware, and vehicle control software. Additional simulators that replicate the real-world performance of radar sensors and DSRC communications are added to the HIL prototype to improve the validity of the testing. The developed prototype system and software can be easily extended to multi-vehicle testing for other V2V applications, such as cooperative lane change.

The developed HIL system can be applied for the following purposes: (i) testing and verifying CAV applications prior to field experiments; (ii) quantifying potential benefits of CAV applications in different operating environments (e.g., traffic conditions, GPS accuracy); (iii) testing vehicle connectivity and automation functions under different virtually created special conditions (e.g., stop-and-go vehicle movement, emergency braking); and (iv) testing performances of critical hardware and software components of CAV vehicle platforms.

2. Literature Review

Assumptions about traffic controllers in traffic simulation may not be realistic as vendors are not willing to share the exact algorithms used in their products. To overcome this limitation, a physical traffic controller can be integrated directly into the simulation. The most basic HIL architecture in the transportation field is comprised of three components: a microscopic simulation software, physical signal controller, and controller interface device (CID). This platform configuration is widely used by researchers to test different traffic control logic (e.g., [3, 4]). HIL simulation is also suitable for evaluating the performance of newly established signal control logic in a CV environment. Compared to the traditional setup of HIL for signal control testing, two additional physical hardware components are "in the loop" in an environment with V2V and vehicle-to-infrastructure (V2I) communication: on-board units (OBUs) and roadside units (RSUs) [5]. To better understand human driving behavior and its causes and effects on surrounding traffic, driving simulators are used with simulation software (also called human-in-the-loop). [6]. HIL simulation has been widely used to develop and calibrate automotive components. One representative HIL tool for this purpose [7] is a laboratory powertrain research platform, which consists of a physical engine, an engine-loading device (hydrostatic dynamometer), and a virtual powertrain model to represent a vehicle, was connected remotely to a microscopic traffic simulator (VISSIM).

Integrated testing platforms that contain multiple pieces of hardware and software have also attracted researchers' attention. These platforms offer greater convenience by allowing researchers to select the required hardware and software as needed. The integrated traffic-driving-networking simulation (ITDNS) includes three elements: driving simulator, network simulation, and traffic simulation [8]. Texas Transportation Institute (TTI) developed a comprehensive HIL platforms for CV applications. The system consists of four components: traffic simulation, communication simulation, applications testing, and DSRC hardware testing environments [9]. Ma et al. developed a HIL testing system for CAV V2I applications. The involved software and hardware include physical CAVs controlled in real-time, traffic signal controllers, communication devices, and a traffic simulator (VISSIM). Such HIL systems increase validity by considering the physical vehicle's trajectories, which are constrained by real-world factors such as GPS accuracy, communication delay, and vehicle kinematics, in a simulated traffic environment [2].

The previous platforms either focus only on a single piece of hardware or rely on the infrastructure-side equipment or data sources. The performance of some existing HIL systems (e.g., delay in synchronization) may not meet the requirement in a V2V environment especially for safety-critical applications. Therefore, new systems and methods are needed to test CAV V2V applications, which have different requirements, such as low latency in data exchange between the physical and simulated environment, especially when physical vehicles platoon with virtual vehicles. Other differences include requirements for relatively long test tracks and for the HIL system to be installed in the testing vehicles. This study aims to develop such a system framework that can enable the testing of V2V applications with CAVs.

3. Hardware-in-the-Loop Testing System

3.1 System Architecture

One critical goal of this study is to develop a CAV HIL prototype that can be extended to more complex systems when more resources are available. Therefore, using CACC as an example, we propose the development of a fundamental V2V HIL module. This architecture can be used as a starting point in the future when different V2V applications, such as CACC or cooperative lane change, need to be evaluated. These applications are also re-usable when specific evaluation systems are developed. Figure 1 shows the data flow and system configuration for this implementation. Hardware components are in green boxes and software in orange ones. The solid lines represent Ethernet communication between the PC and physical vehicle. The dashed lines are data exchanges between software components. The simulation sends the real-time virtual vehicle status to HIL interfaces, which then convert these information into basic safety messages (BSMs). Both

BSMs of virtual and physical vehicles are sent to a DSRC simulator to consider message latency and packet drops; the DSRC communication simulator can vary in complexity depending on the scope of the research question. The DSRC communication simulator can be a simple package developed by the research team that focuses on consideration of message latency and loss. This approach is sufficient for the sensitivity analysis in this project as the parameters, such as packet drop rate, can be fully controlled. The communication simulator can also be a complex, event-based communication simulator (e.g., ns-3) to simulate the field communication under different environment.

The simulated BSMs are then input into a CACC application. In addition, there is also radar measurement data (i.e., gaps) generated based on the traffic simulation. This enables the physical vehicles to sense the simulated vehicles for the purposes of gap regulation. This simulated radar data will pass through a radar simulator before it is supplied to the CACC application. The radar simulator is responsible for simulating any latency or noise that might be required to mimic real-world radar performance. As with the communication simulator, the radar simulator can be a simple parameter-based system using random delay and noise generators or it can be designed with enough complexity to model the physical process of the radar detection and range estimation.

The CACC application then generates speed commands based on simulated field BSMs and radar information. This application is implemented as a separate software package that was developed by the research team based on the current vehicular platform at the FHWA Saxton Transportation Operations Laboratory (STOL). The existing CACC algorithm and corresponding parameters (e.g., control gains) implemented on the vehicle is based on the previous CACC development effort on the same vehicular platform [1]. Testing and fine-tuning the algorithm is conducted once the HIL system is set up to ensure the algorithm performs well under different scenarios.

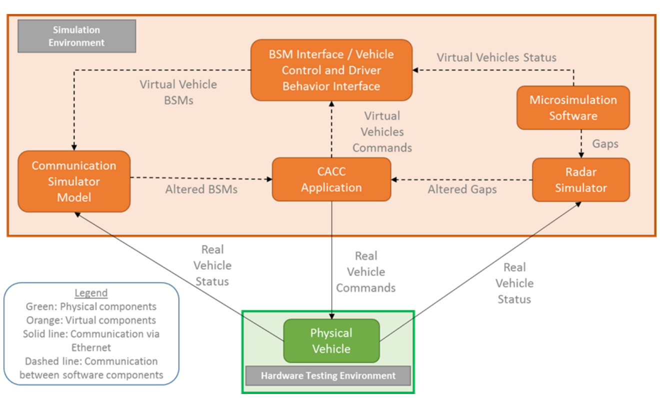


Figure 1. CACC HIL testing system components and data flow

The traffic simulator generates a virtual environment that both virtual and physical vehicles can operate together. The simulated virtual vehicles can follow any pre-specified trajectories or operate under certain car following and lane changing simulation rules. One virtual vehicle in the traffic simulator is used to represent the physical following vehicle in the field. The position and movement of the virtual vehicle in the traffic simulation is synchronized with that of the field vehicle in real time (e.g., based on the real-time data provided by the physical vehicle CAN BUS and GPS).

The basic V2V HIL tool in Figure 1, which is tested in this study, is also applicable to more complex V2V applications with some level of enhancement, such as platooning or lane change. This study is to use this fundamental tool to evaluate multiple vehicle CACC string performance. The purpose of this case study is to illustrate that CACC string performance can be evaluated even with only one CAV available, which is the case in many labs. As shown in Figure 2, one physical vehicle follows a virtual vehicle in the virtual environment, whose trajectory can be pre-specified. The physical vehicle is the follower in the two-vehicle CACC string and its trajectory is generated and recorded in Run 1 of the experiment. In Run 2 of the experiment, the recorded trajectory of the physical vehicle in Run 1 will be input into the virtual environment as a virtual, second vehicle in the CACC string. In the meantime, the physical vehicle is the third vehicle in the CACC string and it operates according to the algorithm based on “real-time” data from both virtual vehicles in the front. With this capability, the system can be used to evaluate the performance of a long CACC string. Note that as long as the CACC control algorithm follows a leader-follower structure, which is the case in most of the CACC applications in the literature, the use case in Figure 2 is applicable. In the field experiment, this study will apply the base HIL system in Figure 1 and use the method in Figure 2 to evaluate CACC.

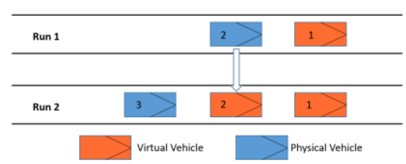


Figure 2. Use case of multi-vehicle CACC evaluation

3.2 Automated Vehicle Platform

Figure 3 demonstrates all required hardware and software components for the HIL testing and the data flow between them. One physical CAV (i.e., Cadillac SRX at the STOL) is used in the HIL testing. The vehicle is equipped with a radar detector, in-vehicle Linux PC, corrected GPS, OBU and MicroAutoBox (MAB). At the center of the vehicle control system is the in-vehicle Linux PC. PinPoint™ transmits real-time high-accuracy GPS data to the in-vehicle computer. The OBU broadcasts BSMs and receives the virtual vehicle’s BSMs; it then transmits this information through the Linux PC. The long-range radar transmits object data to the Linux PC. The MAB receives data from Linux PC, including BSMs from other vehicles and radar data. The MAB control commands are speed recommendations from the control algorithm embedded in Matlab Simulink, which are then injected into the vehicle's controller area network (CAN) BUS. However, not all components of the regular vehicle control system will be used. Only one physical vehicle is used in this HIL testing and the OBU is not necessary since data can be obtained from CAN BUS and GPS directly before being sent to traffic simulation. However, in future research when multiple physical vehicles need to be used, the synchronization may require over the air BSMs by using OBUs. A laptop running the virtual simulation environment is placed in the physical vehicle and connected to the in-vehicle PC directly for data exchange; the computer could have been placed in the roadside or cloud, but would be subject to some communication delay issue. A DSRC communication simulator is developed and run in real time on the PC to alter DSRC messages, including simulation of message latency and packet drops. Additionally, gap information from the traffic simulation can be extracted and sent to the in-vehicle computer for processing before being sent to the vehicle lower-level controller through CAN BUS. This processing requires a radar simulator that also runs on the PC and adds latency and random noise to the range data. Note that the CACC algorithm to be used in this study, similar to the one implemented in the current CARMA platform, only uses radar range data and does not use the range rate as the input.

3.3 Parameter-Based Communication and Radar Simulator

In order to simulate the communication and radar performance in real time, efficient parameter-based random simulators are developed and adopted. For the communication and radar systems, four parameters are identified because of their potential influence on CACC performance. These four parameters are radar delay, radar noise, DSRC latency and DSRC packet loss. For a pre-specified data internal (currently set as 0.1s for both CACC simulation and real-world vehicle implementation), the radar and communication delay are directly imposed on the time when the generated messages can be received by the targeted receiver. For example, if the radar delay is set as 0.1s, this means all the vehicles can only receive an expected message 0.1s after the scheduled times. For other random parameters, Monte Carlo simulation runs are completed to obtain the exact noise that will be added to the true radar range data and select which packets are lost. These effects are also added to the CACC simulator and combined with delay effects.

Note that although in the real world, the physical process of the radar and communication component is much more complex, we believe that the simple parameter-based simulation is beneficial for the purpose of the testing in this study. The goal of the testing is to understand the CACC vehicle and string behavior under various conditions. With the physical vehicle in the loop, we are able to observe real vehicle responses. Then, the parameter-based simulator allows the possibility of conduct controlled experiment by varying communication and radar parameters deterministically, and we can, therefore, observe vehicle and string behavior under these controlled conditions. This helps to gain an in-depth understanding of the extent of the impact of each parameter on the CACC performance.

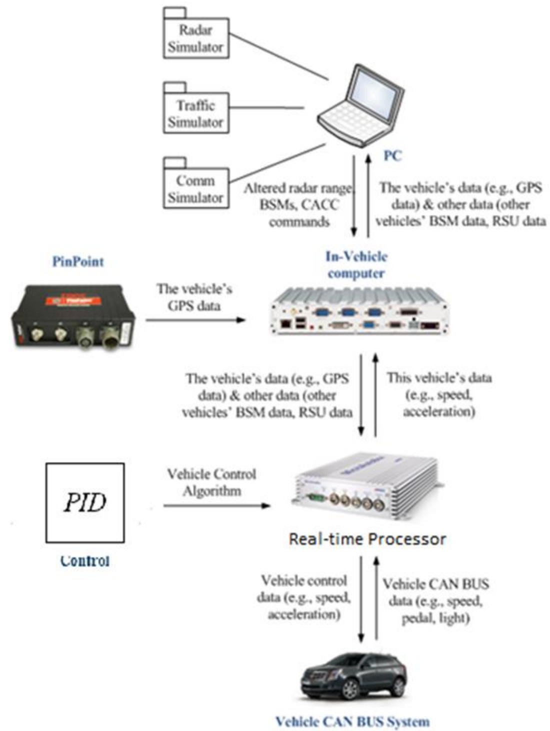


Figure 3. Data flow of the vehicle control systems.

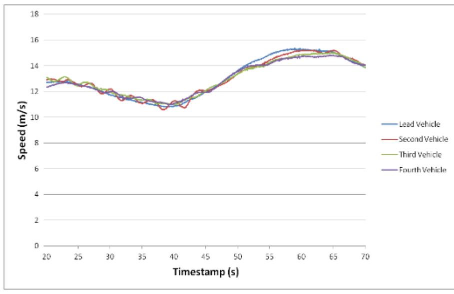
5. EXPERIMENTAL DESIGN, TESTING AND RESULTS ANALYSIS

5.1 Experimental design

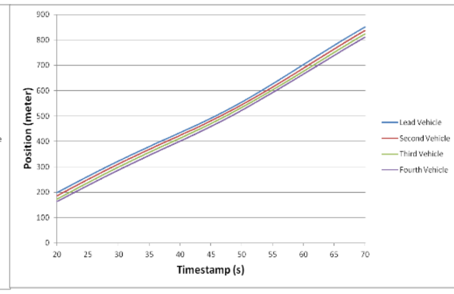
The first goal of this CACC HIL testing is to ensure that the physical test vehicle and virtual vehicles generated by simulation are synchronized and data exchanges between various system components are functional. For each run of the testing, the test platoon consists of two vehicles: a virtual lead vehicle followed by a physical test vehicle. The set speed of the platoon is dependent on the roadway condition of the available test track. Initial testing is conducted at the FHWA TFHRC testbed, where the travel speed is around 30 mph. The DSRC communication is subject to latency and packet losses; it is critical to account for these adverse conditions through the use of DSRC simulation. In addition, the testing will evaluate the effects of different radar conditions on the performance of the CACC system. Since the radar measurements in a traditional CACC system are replaced by simulated measurements to virtual vehicles, these radar conditions will also need to be simulated. Gap measurements from the traffic simulation can be delayed by a variable amount of time to simulate radar latency. Random noise of radar data can also be added to the simulated measurements to simulate radar inaccuracy.

0.05; DSRC Latency = 100 ms and DSRC Packet Loss = 10%.

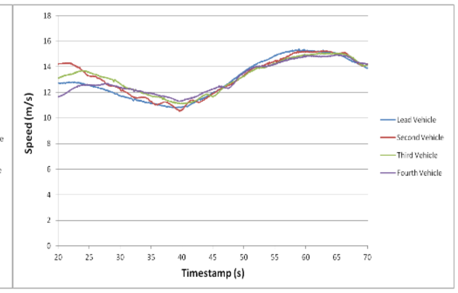
5.2 Hardware-in-the-loop testing results



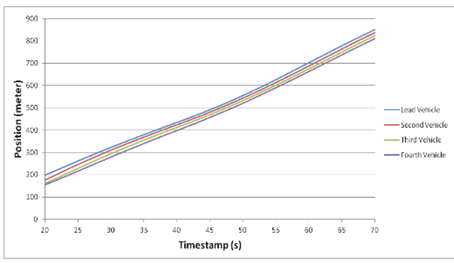
4(a) Type IA Speed profile



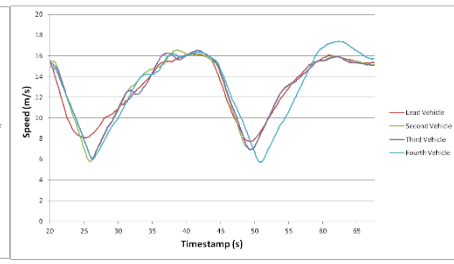
4(b) Type IA Speed profile



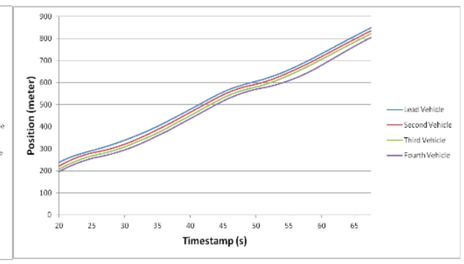
4(c) Type IB Speed profile



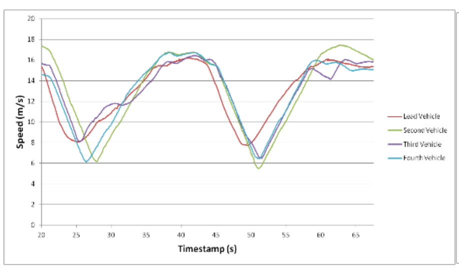
4(d) Type IB Speed profile



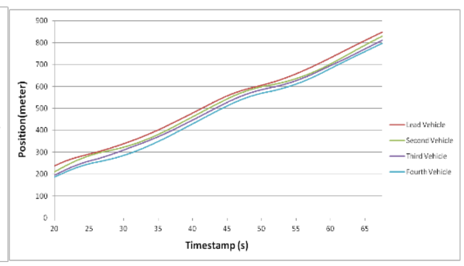
4(e) Type IIA Speed profile



4(f) Type IIA Speed profile



4(g) Type IIB Speed profile



5(h) Type IIB Speed profile

Figure 4. Testing results with different leader profiles and communication performance)

The experiment concerns CACC performance under two general settings. One is the speed profile of CACC string leader: smooth (Type I) and aggressive (Type II). The other is the performance of communication and radar data: perfect (Type A) and compromised (Type B). Therefore, four scenarios are tested by combining the two settings: Type IA, Type IB, Type IIA and Type IIB. The speed profile of CACC string leader is generated by manual driving the test vehicle on the test track. The Type I smooth trajectory is slow acceleration and deceleration between 20 and 30 mph. The Type II aggressive profile involves multiple sharp brakes from 30 mph to 10 mph and quick recovery to 30 mph, replicating stop-and-go traffic behavior. The Type A perfect communication/radar indicates there is no noise and no delay added to the data input to the CACC algorithm. The Type B compromised communication/radar uses the following parameters: radar delay = 100 ms; radar noise =

Figure 4 shows the testing results, speed and position profiles for all four types of scenarios. Figure 4(a)–(b) shows the results of Type IA scenario with the smooth leader profile and perfect communication/radar performance. Figure 4 (a) shows that all four vehicles maintain similar speed during the test, indicating that the string is relatively stable thanks to the perfect communication/radar. It also shows that the speed oscillation decreases as vehicle position in the string increases. For example, the speed profile smoothness of the fourth vehicle is significantly better than that of the second vehicle. Figure 4(b) also confirms the stability of the CACC string with homogeneous gaps maintained.

Figure 4(c)–(d) shows the results of Type IB scenario with the smooth leader profile and compromised communication/radar performance. Figure 4(c) shows that none of the vehicles maintains similar speeds, although the following vehicles' desired speeds are set to the leader's speed, indicating that the string is not stable due to the compromised communication/radar. However, it shows that the speed oscillation decreases as vehicle position in the string increases, similar to Figure 4(a). This implies the effectiveness of the CACC algorithm in absorbing certain disturbance and oscillation of speeds. Figure 4(d) also confirms the stability problem of the CACC string because the gaps between vehicles constantly vary.

Figure 4**Error! Reference source not found.**(e)–(f) shows the results of Type IIA scenario with the aggressive leader profile and perfect communication/radar. Figure 4(e) shows that all four vehicles maintain similar speeds, though not as good as Figure 4(a), indicating that the string is relatively stable and the aggressive leader profile cause only small issues of stability. Figure 4(f) also confirms the stability problem of the CACC string because the gaps between vehicles constantly vary.

Figure 4(g)-(h) shows the results of Type IIB scenario with the aggressive leader profile and compromised communication/ radar performance. Figure 4**Error! Reference source not found.**(g) shows that none of the following vehicles maintains similar speeds, although the following vehicles' desired speeds are set to the leader's speed, indicating that the string is not stable due to the compromised communication/ radar. However, it shows that the speed oscillation decreases as vehicle position in the string increases, similar to Figure 4(a). This implies the effectiveness of the CACC algorithm in absorbing certain disturbance and oscillation of speeds, even with aggressive leader profiles. Figure 4(h) also confirms the stability problem of the CACC string because the gaps between vehicles constantly vary throughout the experiment. At some locations, the inter-vehicle gaps become very small, and though no crash occurs, and it poses a concern for safety.

As shown in Figure 4, different scenarios generate significantly different CACC string performance. The data collected from a physical vehicle are more credible when trying to understanding CACC string performance. The proposed HIL tool provides a cost-effective approach for testing CACC applications. Along with our previous work in V2I CAC testing HIL system, which proved effective through experimentation, we believe that the HIL tools are a valuable tool for CAV application evaluation.

6. Conclusions and Future Research

This study develops a HIL proof-of-concept testing prototype for CAV applications in a V2V environment. The contributions of this study include: (i) developing a HIL testing architecture for V2V-based CAV applications; (ii) implementing the proposed HIL architecture and developing a prototype; (iii) and testing CACC as the selected use case to observe system performance. The results also contribute to an understanding of CACC vehicle and string performance. The system includes a physical CAV, DSRC equipment (i.e., OBUs), a traffic simulator, and a field test track. The physical vehicle is used in the loop as a piece of hardware and it can fully test the overall performance by taking into account vehicle dynamics, vehicle hardware, and vehicle control software. Additional simulators that replicate the real-world performance of radar sensors and DSRC communications are added to the HIL prototype to improve the validity of the testing.

Future research can take different directions. First, more testing will be conducted by varying different parameters. Second, human factors should be considered to better understand human aspects of CAV applications, such as gap acceptance and decisions to engage automated technologies. Third, there should be the integration of V2V and V2I [2] CAV HIL tool under the same framework and testing complex CAV applications in the same testbed.

Acknowledgment

This study is funded by US Department of Transportation (Project Number: DTFH61-12-D-00030). The work presented in this paper remains the sole responsibility of the authors.

References

[1] V. Milanés and S.E. Shladover, "Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data," *Transportation Research Part C: Emerging Technologies*, 48, pp. 285-300, 2014.

[2] J. Ma, F. Zhou, Z. Huang, C. Melson, and R. James, "Hardware-In-The-Loop Testing Of Connected and Automated Vehicle Applications: A Use Case for Signalized Intersection Approach and Departure," Transportation Research Record: Journal of Transportation Research Board, 2018.

[3] D. Bullock, J.M. Morales, and B. Sanderson, "Impact of Signal Preemption on the Operation of the Virginia Route 7 Corridor," Proceedings of the 1999 ITS America Conference, 1999.

[4] J. So, J. Lee, and B. Park, "Evaluation of Manual Traffic-Signal Control under Oversaturated Conditions Using Hardware-in-the-Loop Simulation," *Journal of Transportation Engineering*, vol. 139, no. 11, Nov. 1, 2013, DOI: 10.1061/(ASCE)TE.1943-5436.0000597.

[5] J. Ding, Q. He, K.L. Head, F. Saleem, and W. Wu, "Development and Testing of Priority Control System in Connected Vehicle Environment," 92th Annual Meeting of the Transportation Research Board, Washington, DC. 2013.

[6] M. Klüver, C. Herrigel, C. Heinrich, H-P. Schöner, and H. Hecht, "The Behavioral Validity of Dual-task Driving Performance in Fixed and Moving Base Driving Simulators," *Transportation Research Part F: Psychology and Behaviour*, 2016.

[7] M. A. M. Zulkefli, P. Mukherjee, Z. Sun, J. Zheng, H. X. Liu, and P. Huang, "Hardware-in-the-loop testbed for evaluating connected vehicle applications," 95th Annual Meeting of the Transportation Research Board, Washington D.C., 2016.

[8] Y. Hou, Y. Zhao, K.F. Hulme, S. Huang, Y. Yang, A.W. Sadek, C. Qiao, "An Integrated Traffic-driving Simulation Framework: Design, Implementation, and Validation," *Transportation Research Part C* 45, pp. 138–153, 2014.

[9] The Texas A&M Transportation Institute, Siemens, and Battelle, "New Approaches for Testing Connected Highway and Vehicle Systems Design Document (Draft)," Publication FHWA BAA, no. DTFH61-13-R-00011. Federal Highway Administration, U.S. Department of Transportation, 2014.