Cooperative Ramp Merging Design and Field Implementation: A Digital Twin Approach Based on Vehicle-to-Cloud Communication

Xishun Liao, *Student Member, IEEE*, Ziran Wang, *Member, IEEE*, Xuanpeng Zhao, Kyungtae Han, *Senior Member, IEEE*, Prashant Tiwari, Matthew J. Barth, *Fellow, IEEE*, and Guoyuan Wu, *Senior Member, IEEE*

***Abstract*— Ramp merging is considered as one of the most difficult driving scenarios due to the chaotic nature in both longitudinal and lateral driver behaviors (namely lack of effective coordination) in the merging area. In this study, we have designed a cooperative ramp merging system for connected vehicles, allowing merging vehicles to cooperate with others prior to arriving at the merging zone. Different from most of the existing studies that utilize dedicated short-range communication, we adopt a Digital Twin approach based on vehicle-to-cloud communication. On-board devices upload the data to the cloud server through the 4G/LTE cellular network. The server creates Digital Twins of vehicles and drivers whose parameters are synchronized in real time with their counterparts in the physical world, processes the data with the proposed models in the digital world, and sends advisory information back to the vehicles and drivers in the physical world. A real-world field implementation has been conducted in Riverside, California, with three passenger vehicles. The results show the proposed system addresses the issues of safety and environmental sustainability with an acceptable communication delay, compared to the baseline scenario where no advisory information is provided during the merging process.**

***Index Terms*— Cooperative merging, vehicle-to-cloud communication, digital twin, field implementation.**

I. Introduction and Background

A. Motivation

RAPID growth of motor vehicles in transportation systems brings economic prosperity to our society, however it also introduces numerous issues worldwide in terms of safety, mobility, and environmental sustainability. According to the report published by U.S. Department of Transportation's National Highway Traffic Safety Administration, approximately 37,000 people perished in motor vehicle traffic crashes in 2018 in the U.S. [1]. It is estimated by INRIX that an average of 97 hours is spent by each driver in the U.S. due to traffic congestion [2]. Traffic congestion also caused 44.3 billion liters of fuel to be wasted worldwide in 2015, according to U.S. Department of Energy [3].

The emergence of intelligent transportation systems technology brings about solutions to the aforementioned issues. Specifically, connected vehicle technology takes advantage of vehicle-to-everything (V2X) communication, allowing vehicles to communicate with other road participants and hence conduct more efficient maneuvers. V2X-enabled connected vehicles are equipped with Dedicated Short-Range Communications (DSRC) [4] and/or a cellular network, allowing for communicating with each other. Representative V2X scenarios include vehicle-to-vehicle (V2V) communication, vehicle-to-infrastructure (V2I) communication, and vehicle-to-cloud (V2C) communication. The communication module of a connected vehicle can provide additional information that cannot be readily detected by perception sensors (if any) and can generally provide information more quickly than through sensor detection and processing [5].

Vehicles merging near the ramp area has been a major concern that generates numerous potential conflicts. The difficulty arises for drivers of ramp vehicles along the on-ramp, where drivers have to discern to accelerate/decelerate to enter the mainline safely without clear line of sight regarding the mainline traffic. Meanwhile, drivers of mainline vehicles may have to modify their vehicle speeds to permit the entrance of ramp vehicles, thus affecting upstream traffic flows and consuming excessive fuel. To address these issues, the cooperative merging of vehicles at ramp has been studied and applied by various researchers around the globe, where connected vehicles take advantage of V2X communication to communicate with vehicles coming from the other lane directly or through road side infrastructure, and hence conduct cooperative merging maneuvers in a safe and smoothed manner [6]–[8].

B. Literature Review

Numerous cooperative ramp merging methodologies have been proposed recently, and their impacts were proved through numerical or microscopic traffic simulations. The concept of utilizing virtual vehicles in the highway on-ramps cooperative

Manuscript received March 2, 2020; revised October 8, 2020 and November 27, 2020; accepted December 10, 2020. Date of publication July 30, 2021; date of current version May 3, 2022. This work was supported by the Toyota Motor North America, InfoTech Labs under the Digital Twin Project. The Associate Editor for this article was A. Hegyi. *(Corresponding author: Xishun Liao.)*

Xishun Liao, Xuanpeng Zhao, and Matthew J. Barth are with the Department of Electrical and Computer Engineering, University of California at Riverside, Riverside, CA 92507 USA (e-mail: xliao016@ucr.edu).

Ziran Wang, Kyungtae Han, and Prashant Tiwari are with InfoTech Labs, Toyota Motor North America, Mountain View, CA 94043 USA.

Guoyuan Wu is with the Department of CE-CERT, University of California at Riverside, Riverside, CA 92507 USA.

Digital Object Identifier 10.1109/TITS.2020.3045123

merging case was originated from Uno *et al.* [9] and got adopted by some consecutive studies [10]–[12]. The proposed approach maps virtual copies of the real vehicles (both with the same longitudinal speed and position) onto the other merging lane before the actual merging happens, so connected vehicles can adjust their formation in advance and avoid last-minute speed changes. Other than the virtual vehicle concept, Dao *et al.* proposed a distributed control protocol to assign vehicles into strings in the merging scenario [13]. Rios-Torres *et al.* presented an optimization framework and an analytical closed-form solution that allowed online coordination of connected and automated vehicles (CAVs) at ramp merging zones [14], and further studied the impact of partial penetrations of CAVs on fuel consumption and traffic flow for the ramp merging scenario [15]. A rule-based cooperative merging strategy was proposed by Ding *et al.* very recently, and the simulation results showed their strategy outperformed other rule-based strategies and optimization-based strategies [16]. Besides numerical simulation and/or microscopic traffic simulation adopted by the aforementioned studies, game engine simulation was also used to evaluate the cooperative ramp merging system, where human-in-the-loop simulation can be conducted and compared with the proposed methodology as a baseline [17].

Although many studies proved the effectiveness of their proposed cooperative ramp merging methodologies with simulation, far fewer studies conducted field experimentation. Milanes *et al.* proposed a fuzzy logic-based controller for the CAV's longitudinal motion control, and their experimental implementation emulated a congested traffic situation by allowing a Citroen to use Adaptive Cruise Control to follow another one at a low speed [18]. The Federal Highway Administration conducted a field implementation of cooperative lane change maneuvers with two CAVs and a manually driven vehicle [19]. A simplified PID controller was developed to coordinate the longitudinal speeds and positions of different vehicles, and the evaluation results at the Federal Law Enforcement Training Center in Cheltenham in Maryland showed its effectiveness in cut-in, front-join and back-join scenarios, respectively. Researchers from the University of Minnesota [20], East Tennessee State University [21], [22], as well as Technical University of Madrid [23] also conducted similar field implementations on highway on-ramp merging scenarios using V2X communication.

C. Contribution of the Paper

Although the aforementioned studies conducted preliminary field implementation about respective cooperative merging systems, they all used DSRC-based V2X communication. The information of other merging vehicles was received by the DSRC On-Board Unit (OBU) on the ego vehicle, and the related computation of the actuator input or advisory speed was conducted by the on-board computer. In this study, we adopt 4G/LTE-based vehicle-to-cloud communication to design our cooperative ramp merging system. Compared with DSRC method, the V2C method improves the scalability of vehicle communication, provides a larger data storage which enables data driven human behavior modeling, and break the constraint of computational power.

A field implementation is conducted by three passenger vehicles which are all equipped with the "Digital Twin" system (a system that is highly portable, does not require vehicle on-board computation and requires no level of automation). All computations are conducted by the Digital Twin model on the cloud server and sent to different vehicles through 4G/LTE cellular communication. The advisory speed calculated by the cloud server is shown on a Human-Machine Interface (HMI), so the driver can simply follow that advisory speed and conduct the cooperative merging maneuvers in a safe and smoothed manner.

Compared to all the existing studies about cooperative merging of connected vehicles, this study has the following major contributions:

- As a realization of the "Digital Twin" concept, a flexible cloud-based CAV system framework has been developed and demonstrated.
- To the best of our knowledge, this is the one of the first research studies to implement cooperative ramp merging in the field using 4G/LTE communication.
- The cellular-based V2C communication delays are measured and analyzed in the real-world environment.

D. Organization of the Paper

The remainder of this article is organized as follows: Section II introduces the problem statement of this study. Section III explains the detailed design of the proposed V2C based cooperative ramp merging system. Section IV introduces the field implementation of the proposed system using three passenger vehicles, and its results are illustrated and analyzed in Section V. Finally, the article is concluded with some future directions in Section VI.

II. Problem Statement

The objective of this study is to design a cooperative ramp merging system, so drivers can drive those vehicles with the help of this system to conduct safe and smooth merging maneuvers with other merging vehicles cooperatively. The proposed system can be simply illustrated as Fig. 1, where all relevant merging vehicles are assumed to have access to the cloud server through V2C communication. The cloud server processes the data received from various vehicles in real time, and also sends advisory information back to the relevant vehicles with certain updating frequency.

As can be seen from the figure, once the merging sequence of various merging vehicles is decided by the proposed model, the merging maneuver between any two vehicles is simplified into a car-following problem, where the virtual vehicle concept is adopted from our previous study [12]. Instead of following the physical preceding vehicle, the ego vehicle follows the virtual copy of that preceding vehicle, where the virtual one has exactly the same longitudinal speed and position as the physical one (except they are on two different merging lanes). Although the control topology (i.e., the following topology) is directly from one vehicle to another, the actual communication

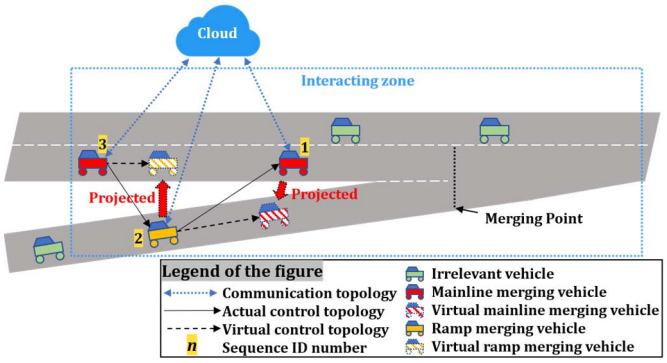


Fig. 1. Cloud-based cooperative merging scenario at on-ramp.

is conducted through the cloud server, thus making it a vehicle-cloud-vehicle communication topology.

Note that to the current stage of this cooperative ramp merging system, some reasonable specifications and assumptions are made as follows to expedite the implementation:

- The proposed system, including its motion planning and motion control models, only considers the longitudinal vehicle dynamics. Lateral planning and control of the vehicle, such as lane change, are out of the scope of the current study.
- The whole length along the ramp only has one lane, and only vehicles on the rightmost lane of the mainline traffic are considered in this system.
- All relevant merging vehicles in this system are enabled with V2C communication, so they can exchange information with the cloud server in real time.

Above specifications and assumptions are also considered as the target items to be finished at the next stage of this study, where they are also discussed in the future work in Section VI.

III. Methodology

The general architecture of the proposed cooperative ramp merging system is designed as Fig. 2, where the key modules on the cloud server are introduced in this section.

A. Map Matching

The main functions of the map matching module are position synchronization and geo-fencing. A pre-built map of the test field is available on the cloud server, with information such as the road type, road length, road ID, waypoints, direction, road speed limit, merging zone, and interacting zone. For position synchronization, vehicles' coordinates (i.e., longitude, latitude, and altitude) received from the GNSS will be uploaded to the cloud at each time step and be matched to the pre-built map by the proposed map matching algorithm (Algorithm 1). Based on the position given by the map matching algorithm, a geo-fencing function is defined to check the positions and conditions of the vehicles at each time step, so associated actions can be conducted accordingly.

B. Motion Planning: Merging Sequence Determination

Shown as Fig. 4, there are different ramp merging scenarios, including 1) merging into a vehicle string (of two or more vehicles), 2) merging after a vehicle or string, 3) merging in front of a vehicle or string, and 4) merging to an empty mainline. In Fig. 4(a), merging to a busy mainline and cutting into a string is the most challenging case for human drivers among these scenarios. Therefore, the following analysis will be based on this representative case, with two mainline vehicles and one ramp vehicle. Note that the overall system designed in this study can handle all these four cases.

Algorithm 1 Map Matching Algorithm

Input: Coordinates of nodes (A_i) and associated links (l_i) along the directed path or network, subject vehicle's position (P), shown as Fig. III-B.
Output: Map-matched position (P') of subject vehicle.

According to the subject vehicle's position (P), determine the set of nodes, $\mathbf{N} \triangleq \{A_i\}$ and associated set of links, $\mathbf{L} \triangleq \{l_i\}$, based on the spatial proximity;
for each node $A_i \in \mathbf{N}$
  calculate the length between P and A_i;
  calculate the angles $\angle PA_iA_{i-1}$ and $\angle PA_iA_{i+1}$;
  if $(\angle PA_iA_{i+1} < 90° \& \angle PA_{i+1}A_i < 90°) \,|\angle$
    $PA_iA_{i+1} = 90°$ put l_i into the candidate set $\mathbf{\Theta}$;
  end if
end for
for each link $l_j \in \mathbf{\Theta}$
  calculate the distance between P and l_j;
  determine the projection of P on link l_j, i.e., P'_j
end for
return $P^* \triangleq \text{argmin}_j \left\{ \bar{PP'_j} \right\}$;

Motion planning module is designed to cooperatively plan the motions of three merging vehicles, with two vehicles on mainline and one on ramp. Considering the succinctness of description, with regard to the initial state, we name the first vehicle on mainline as "MV1", the second vehicle on mainline as "MV2", and the vehicle on ramp as "RV", hereafter.

Algorithm 2 is designed to conduct the merging behavior of the three vehicles. There are two stages in the motion planning module: cruise stage and cooperative merging stage. During the cruise stage, MV1 and MV2 keep a relative constant speed and inter-vehicle gap (as a vehicle string) before entering the ramp merging interacting zone. At the same time, an actuation signal will be sent to RV, a countdown for driver will be shown on the HMI, and the advisory speed will appear after the countdown.

In the cooperative merging stage, *Algorithm 3* is defined to check whether RV satisfies the basic requirement, which is designed to constrain the advisory speed of RV to be feasible and safe. If RV follows the advisory speed without large difference, the sequence of the string will be reorganized by switching the leader of MV2 vehicle to RV. Once MV2 is hooked up with RV, the proposed feedforward lookup table will be called up to select a proper control gain for the consensus algorithm, which will be introduced in the III-C

Algorithm 3, as a parallel thread running in background, is designed to judge whether to deactivate the HMI considering RV may fail to track the advisory speed. Considering the speed tracking error, incompliance by human drivers or even

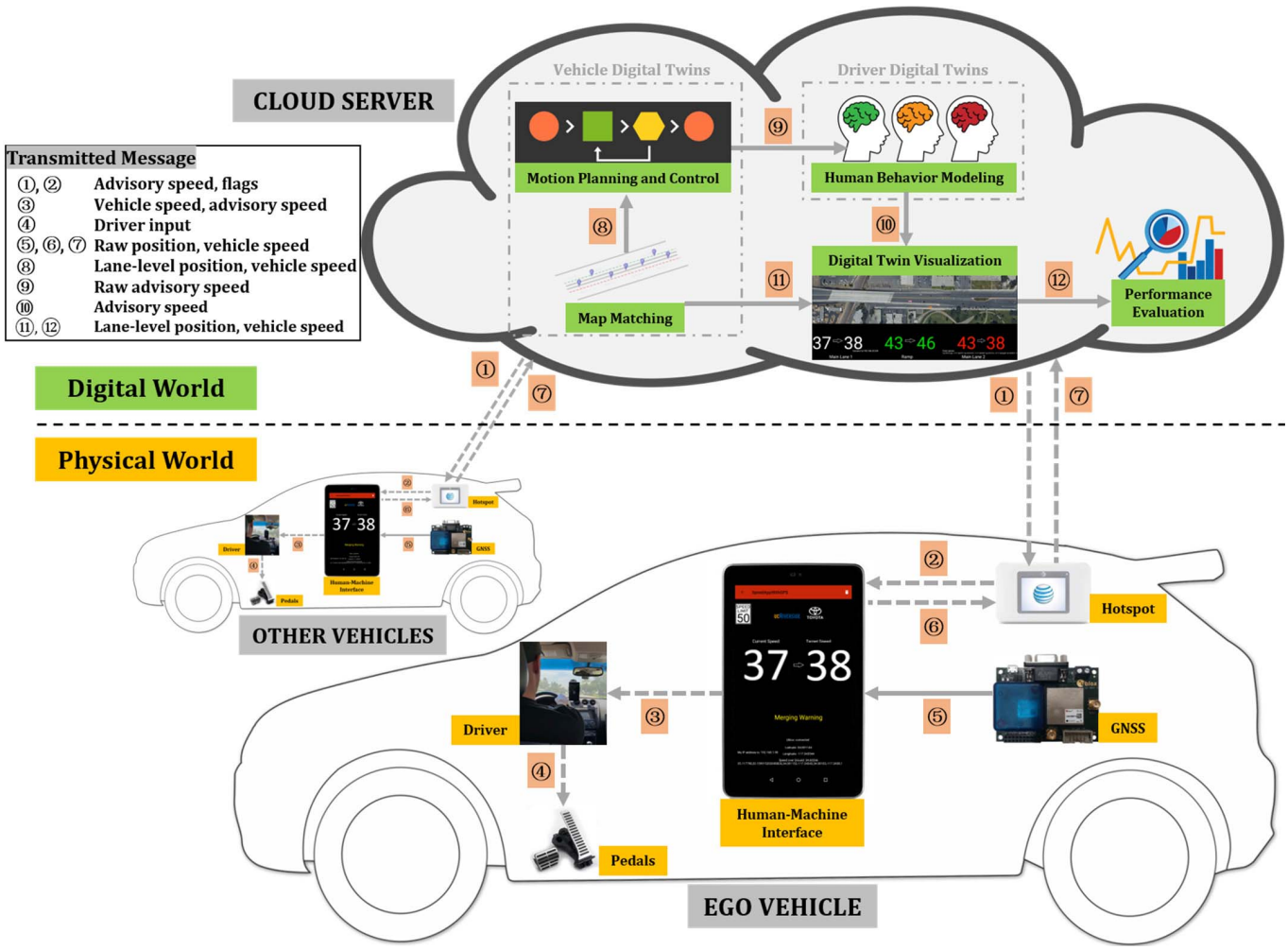


Fig. 2. General architecture of the vehicle-to-cloud based cooperative ramp merging system.

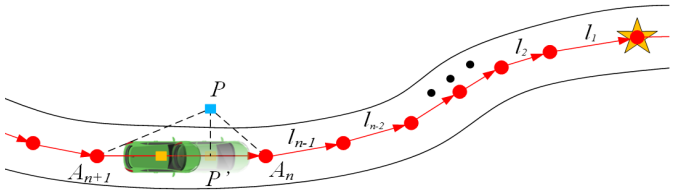


Fig. 3. Map matching algorithm. The "yellow square" and the "blue square" represent the actual and measured positions, respectively, of subject vehicle, and the "orange star" is the waypoint of interest for route distance estimation.

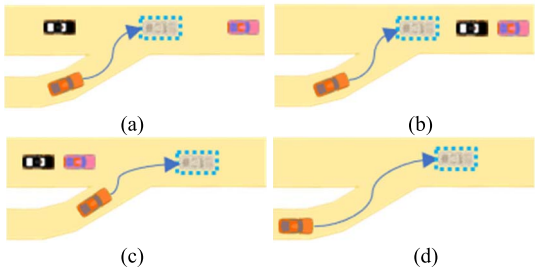


Fig. 4. Merging scenarios: (a) merging into a vehicle string, (b) merging after a vehicle or string, (c) merging in front of a vehicle or string, and (d) merging to an empty mainline.

hardware malfunction, the merging condition may not always be satisfied. For example, when RV is too slow to catch up the mainline vehicles, a higher advisory speed will be given by the control algorithm in the background, which may exceed the speed limit and be infeasible. In *Algorithm 3*, a counter is designed to record the occurrence of such infeasible advisory speed. Once the counter value exceeds a preset threshold, indicating that the merging requirement is not satisfied any longer, the system will deactivate the HMI.

C. Motion Control: Longitudinal Speed Generation

The motion controller on the cloud server generates the raw advisory speed of the ego vehicle. The inputs of the controller are the speed and lane-level position of the ego vehicle and its preceding (virtual) vehicle, where the preceding vehicle is decided by the aforementioned motion planner.

Based on our previous study [24], the lookup table-based consensus algorithm is developed as the motion controller to calculate the advisory speed of merging vehicles in this system, where the block diagram is shown as Fig. 5. The consensus algorithm is in the feedback control loop, while a parameter of this algorithm (damping gain) is calculated based on a feedforward lookup table.

1) Feedback Consensus Algorithm: In this study, we adopt a consensus-based algorithm from our previous study for the

Algorithm 2 Merging Process Conductor

Input: 1. Distances to Merging-point (D2M) and 2. Speeds (v_i) of the three vehicles. 3. Distances to Interacting-zone (D2I) of MV1 and RV.
Output: Actions to conduct the merging.
Variable: 1. flag1: indicator for location of mainline vehicle string. 2: flag2: indicator for merging condition.

```
Initialization:
flag2 = 1; flag1 = 0; MV2.leader = MV1;
while True:
    if (D2I(MV1) <= 0):// vehicle string in range
        flag1 = 1; // Merging stage, activate HMI on R
        if (D2I(R) <= 0):// RV in range
            RV.leader = MV1;
            RV.control gain = Lookup Table (D2M, v_r);
            flag2 = Algorithm3; // update
            if (flag2 == 1):// Rearrange Merging Sequence
                MV2.leader = R;
                MV2.control gain = Lookup Table (D2M, v_M2);
            else if (flag2 == 0):
                continue; // wait-and-see
            else if (flag2 == −1):
                break; // deactivate HMI
            end if
        else: // RV out of range
            continue;
        end if
    else: // vehicle string out of range
        continue;
        flag1 = 0;
    end if
end while
```

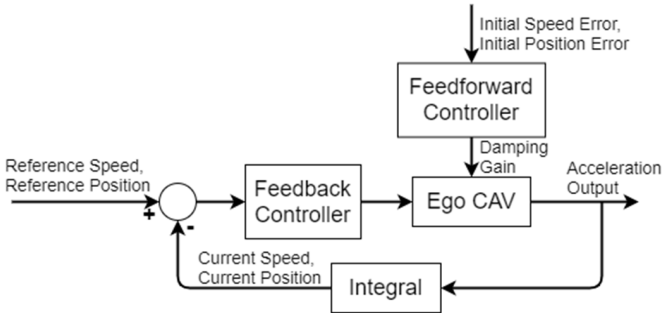


Fig. 5. Block diagram of the lookup table-based consensus algorithm [24].

acceleration control [12]. Basically, once the ego vehicle i is assigned with a preceding vehicle j to follow by the motion planner, it retrieves information of this preceding vehicle through V2C communication, which includes length l_j, longitudinal position r_j, and longitudinal speed v_j of the vehicle. Then, the proposed consensus algorithm takes those inputs as well as longitudinal position r_i and the longitudinal speed v_i of the ego vehicle, and computes a reference acceleration a_{ref} for the ego vehicle at the next time step by the following

Algorithm 3 Ramp Merging Condition Monitor

Input: 1. Distances to Merging-point (D2M) and 2. Current Speeds (v_{ci}) of the three vehicles. 3. Advisory Speed (v_{ai}). 4. Speed Limit (L). 5. Elapsed Time (t).
Output: The merging condition indicator: flag 2.
Variable: 1. D_counter: counter for deactivation. 2. A_counter: counter for activation.

```
Initialization:
D_counter = 0; A_counter = 0; flag 2 = 0;
acceleration threshold = P1; activate threshold = P2;
deactivate threshold = P3;
while True:
    A = v_ai − v_ci;// Advisory acceleration
    if (A>P1 or v2>L): // vehicle string in range
        D_counter ++;
    else:
        A_counter ++;
    end if
    if (A_counter/ t >= P2):// successful rate
        flag2 = 1;
    end if
    if (D_counter/ t >= P3): //higher priority
        flag2 = -1;
    end if
end while
```

algorithm:

$$a_{ref}(t+\delta t) = -\alpha_{ij}k_{ij}\cdot[(r_i(t)-r_j(t-\tau_{ij}(t))+l_j+v_i(t) \cdot(t_{ij}^{g}(t)+\tau_{ij}(t)))+\gamma_i\cdot(v_i(t)-v_j(t-\tau_{ij}(t)))] \tag{1}$$

where δt is the length of each time step, α_{ij} denotes the value of adjacency matrix, which is defined by the information flow topology graph of a distributed network around the vehicles. In the field experiment, the topology of the graph is associated with the communication within the group, such that $\alpha_{ij} = 1$, if vehicle i and j are connected, otherwise, $\alpha_{ij} = 0$. k_{ij} is a control gain, γ_i is a damping gain to be calculated by the lookup table, $t_{ij}^{g}(t)$ is the time-variant desired time gap between two vehicles, $\tau_{ij}(t)$ denotes the time-variant communication delay between two vehicles. In this study, to simplify the model, we set $\tau_{ij}(t) = 88$ms, and more detailed delay analysis is presented in Section V.C. The output of this motion controller, the raw advisory speed of the ego vehicle can then be computed as

$$v_i(t+\delta t) = v_i(t) + a_{ref}(t+\delta t)\cdot\delta t \tag{2}$$

where $v_i(t)$ is the current speed of the ego vehicle i.

2) Feedforward Lookup Table: Based on the sensitivity analysis of our previous study [25], it turned out that the damping gain γ_i of the consensus algorithm highly affects the convergence of the algorithm (i.e., the convergence of two vehicles' longitudinal positions and speeds). Therefore, a lookup table is built as a feedforward input to the feedback control loop, which calculates the ideal values of the damping gain γ_i in terms of different initial states of the vehicles.

Specifically, the initial states of the vehicles are defined as what the longitudinal positions and speeds are when the vehicles first activate the motion controller at time t_0. The initial states of vehicle i and its preceding vehicle j can be given as $(r_i(t_0), v_i(t_0))$ and $\left(r_j\left(t_0-\tau_{ij}(t_0)\right), v_j\left(t_0-\tau_{ij}(t_0)\right)\right)$.

The lookup table is built by running numerical simulations of all the possible combinations of the initial states of vehicles, and picking out the value of damping gain which best satisfies the three major constraints:

- Constraint 1 is the safety constraint, which ensures there is no overshoot of the position consensus term $(r_i(t)-r_j(t-\tau_{ij}(t))+l_j+v_i(t)\cdot(t_{ij}^g(t)+\tau_{ij}(t)))$ during the convergence process, namely no rear-end collision happens.
- Constraint 2 is the efficiency constraint, which ensures the convergence of the consensus happens as soon as possible, namely the ego vehicle adjusts its longitudinal position and speed with its preceding vehicle within a shorter period.
- Constraint 3 is the comfort constraint, which minimizes the maximum changing rate of the consensus algorithm (which is a function of the maximum acceleration and jerk) during the convergence process, namely making the ride as comfortable as possible.

The details of building the lookup table and searching the damping gain value in the lookup table, and the evaluation of the lookup table can all be referred to our previous study [24].

D. Human Behavior Modeling: Speed Tracking Error Compensation

Given the fact that a driver cannot track the advisory speed shown on the HMI perfectly, a speed tracking error is always generated at every time step. It has been concluded from one of our previous studies that, the speed tracking errors may contribute to as high as 12% degradation in system performance of the original advisory speed assistance design [26]. Therefore, it is beneficial to model each driver's behavior and compensate for the speed tracking error in a personalized manner.

In this study, the human behavior modeling module on the cloud server predicts the speed tracking error caused by the driver and compensates for the raw advisory speed in real time. The output of this module is the advisory speed sent to the ego vehicle and shown on the HMI, which already considers the speed error caused by the driver.

Based on the learning-based approach we proposed in our previous study [27], after clustering different drivers in the historical data with Hierarchical Clustering Analysis (HCA), Nonlinear Autoregressive Neural Networks (NARNNs) are trained with respect to different driver types. Therefore, in run time of the cooperative ramp merging system, once a driver is classified as a certain driver type by k-Nearest Neighbor (k-NN) algorithm, this type's NARNN predicts the speed tracking error of the driver at the next time step $\hat{y}(t+\delta t)$ based on its past values. Then the raw advisory speed can be compensated with this predicted speed tracking error as

$$\hat{v}_i(t+\delta t)=v_i(t+\delta t)+\hat{y}(t+\delta t) \tag{3}$$

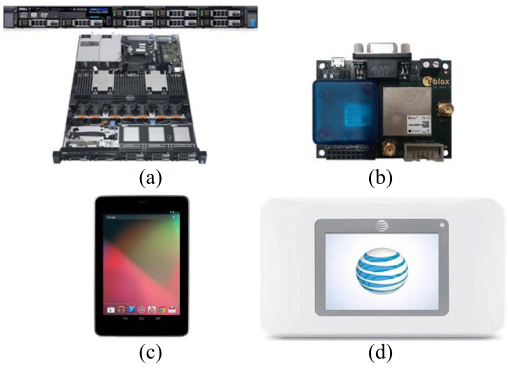


Fig. 6. Hardware in implementation: (a) Dell R630 server, (b) U-blox C94-M8P-2 GNSS unit, (c) Google Nexus 7, and (d) NETGEAR hotspot.

where $\hat{v}_i(t+\delta t)$ is the advisory speed sent to the ego vehicle i and be tracked by the driver on the HMI.

IV. Real Vehicle Field Implementation

In the field implementation of the proposed cooperative ramp merging design, three typical passenger vehicles were utilized as the test vehicles. All three vehicles are 2012 Corolla LE model, with 1.8-litre internal combustion engines. The drivers of these three vehicles are all graduate students in University of California, Riverside, with ages ranging from 23 to 25, and legal driving records between 3 to 5 years. Before the actual field implementation, everyone was familiarized with the developed system by taking a few test runs.

A. Hardware Setup

As mentioned earlier, the Digital Twin framework of the system enables the system to be highly portable, where there are only three vehicle on-board devices (one of them is optional) plus one cloud server. All hardware used in this field implementation (except for the cables) are shown in Fig. 6.

1) Cloud Server: This module coordinates the cooperative merging process, with the algorithms running on a self-build server. As shown in Fig. 6 (a), a Dell R630 server is adopted in this implementation, which is equipped with two Xeon 2.4GHz (6-core) CPUs, 64GB RAM, 1TB solid state drive and 14 TB hard disk drive. In its Windows Server 2012 system, the Python 3.x with python package "numpy" is needed to run the algorithms. Commercial cloud servers such as Amazon Web Services, Microsoft Azure, or Google Cloud can also work as alternatives in this proposed system.

2) GNSS: This module is equipped on the vehicle to measure its real-time raw position (latitude, longitude and altitude) and speed information and then sends them to the HMI device through a micro USB cable. As shown in Fig. 6(b), the u-blox C94-M8P-2 GNSS unit is adopted, with a maximum update frequency of 8Hz. According to a comprehensive field test [28], this GNSS unit has a 2-dimentional (horizontal) accuracy of 46.7 cm. When in the "RTK Fixed" mode, the 2D accuracy is 4.0 cm. Both modes of this GNSS unit can reach a lane level accuracy in an open sky environment.

3) HMI Device: This module shows the advisory information (via V2C communication) to the driver to conduct cooperative/intelligent maneuvers. Shown in Fig. 6 (c), the Google

(a)

(b)

Fig. 7. Perspectives of (a) ramp vehicle driver and (b) mainline vehicle 2 driver while conducting the cooperative ramp merging.

Nexus7 is adopted in this implementation. As shown in IV-B, the information displayed on the HMI include current speed (the left number), advisory speed (the right number), speed limit, and some other additional messages (e.g., latitude and longitude, IP address). Besides visual information, audio notification is also embedded in this HMI. For example, along with showing the yellow text in Fig. 7(b), an audio warning is played to driver of MV2 when the ramp vehicle is approaching.

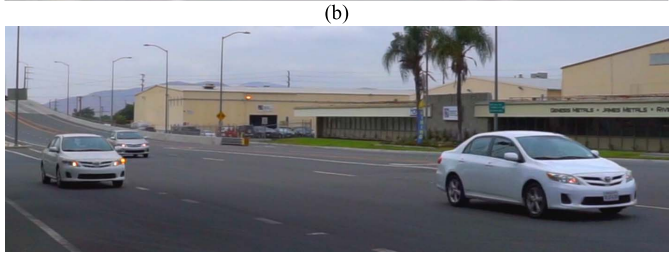
4) Cellular Hotspot (Optional): This module provides cellular access to the HMI device and therefore enables V2C communication in this Digital Twin model. The Wi-Fi hotspot is equipped with a 4G/LTE (potentially 5G) sim card and has the "share hotspot" feature. As shown in Fig. 6(d), the NETGEAR AirCard 770S mobile hotspot is adopted in this implementation. If the HMI device (such as a mobile phone or tablet) is equipped with a sim card, cellular hotspot is not a necessity.

B. Implementation Plan

The test track consists of a ramp and a mainline, where the mainline spans from the intersection of Columbia Avenue and Chicago Avenue to the intersection of Iowa Avenue in Riverside, California. The mainline is on an overpass while the ramp is under the overpass, which increases the difficulty of ramp merging because the visions of both mainline and ramp drivers are blocked by this overpass. The total length of the track is 780 meters, with a merging zone of 89 meters long.

(a)

(b)



(c)

(d)

Fig. 8. Cloud-based cooperative merging stages.

As shown in Fig. 8, the cooperative merging consists of four stages. On stage 1 in Fig. 8(a), MV2 is assigned to follow MV1 and enters the interacting zone with a constant speed. At the same time, RV receives the countdown information from the approaching MV1. On stage 2 in Fig. 8(b), RV is assigned to follow MV1 and starts to accelerate based on the speed suggestion. On stage 3 in Fig. 8(c), MV2 is assigned to follow RV when RV satisfies the requirement. RV is ready to merge, while MV2 is notified to slow down and generates a gap for the merge. On stage 4 in Fig. 8(d), given enough inter-vehicle gap, RV merges into the vehicle string. The implementation setups are listed in Table I, including the preset initial states, constraints, desired states, and the parameters in control algorithm. The control gain of each vehicle will be changed at certain time steps as the merging process goes along. For RV, K_r is given by the feedforward lookup table based on the initial condition when RV and MV1 are hooked up. For MV2, K_{m1} depends on the initial condition when MV1 and MV2 are bonded; K_{m2} depends on the initial condition when MV2 switches its leader from MV1 to RV.

At every time step, the localization coordinates and speed provided by the GNSS units will be read by the HMI device on each vehicle and transmitted to the cloud server. On the

TABLE I
PARAMETER SETUP OF THE COOPERATIVE MERGING FIELD IMPLEMENTATION

Parameters	Ramp Vehicle ("RV")	Mainline Vehicle 1 ("MV1")	Mainline Vehicle 2 ("MV2")
GNSS antenna to front bumper	2.2 m	2.2 m	2.2 m
GNSS antenna to rear bumper	2.3 m	2.3 m	2.3 m
Initial speed	0~4.5 m/s	17 m/s	17 m/s
Desired speed	-	17 ± 2 m/s	-
Desired acceleration range	±3 m/s^2	±1 m/s^2	±2 m/s^2
Initial distance to merging point	390 m	470 m	480 m
Initial inter-vehicle distance	75-102 m	10 m	
Initial time gap	6 s	0.6 s	
Desired time gap	0.6 s		
Control gain	K_r	-	K_{m1}, K_{m2}
Low speed minimum gap	2 m		
Time duration of merging	25 to 30 s		
Communication rate	10 Hz		
HMI update rate	3 Hz		
Interacting zone length	630 m		
Speed limit on HMI	20 m/s		

cloud, the map matching module utilizes the GNSS data to locate these three vehicles and compute the relative distance and speed among them. With the map matching result, the real-time advisory speed for each vehicle can be calculated. More specifically, to smooth the speed and facilitate the following of traffic rules in the test site, an internal acceleration limit for the ramp vehicle is set in the control algorithm with a range of $(-3, 3)$ m/s^2, and the speed display on HMI is bounded by the real-world's speed limit 45 mph (approximately 20 m/s). It should be noted that, when the advisory speed given by the consensus algorithm exceeds the speed limit, it will not be display on the HMI, but only goes to Algorithm 3 to increase the possibility of the HMI deactivation in the background.

V. RESULT EVALUATION

The results evaluation in this section is conducted by the "Performance Evaluation" module, which is executed on the cloud server as shown in Fig. 2 in real time. With a certain update frequency, this module evaluates the speed and distance trajectories, as well as the fuel consumption and pollutant emissions of all connected vehicles. The evaluation results are sent to the drivers to provide them with real-time feedback, and therefore recommend safer or more fuel-efficient driving behaviors. Show as Fig. 9, an app is designed to visualize vehicle Digital Twins running on the cloud server, which have the same parameters (position, speed, etc.) as the vehicles in the real world. Speed advisories for drivers of all three vehicles are also updated in real time. In field experiments, two inspectors, one at passenger seat, another one stay beside the server PC, monitor the whole process using this app.

A baseline implementation is designed and carried out to validate the benefit of the proposed cooperative merging. Since an unbiased comparison requires a same merging scenario, a low-grade HMI with only partial function is adopted in the baseline scenario. To ensure the three vehicles encounter with each other, the HMI provides speed guidance to RV only

Fig. 9. Digital Twin of vehicles running on the cloud server in real time.

before RV reaches a "deactivate" point. The point is set at 85 meters before the observable point, where the driver of RV drives "blindly" (without HMI or line-of-sight of other merging vehicles) within that 85 meters range. Without speed guidance, MV2 will not create gap for RV in advance.

In our field implementation, four cooperative merging trips, and four baseline trips are carried out, respectively. In this section, we present one typical cooperative merging trip that was carried out at 12:19:57 p.m. on September 27th, 2019. The proposed vehicle-to-cloud based cooperative merging system is compared with the baseline system in terms of safety, fuel and emission. Moreover, an implementation is conducted to investigate the V2C communication.

A. Safety Results

As can be seen from the cooperative merging implementation results, Fig. 10(a) shows the speed trajectories, and Fig. 10(b) shows the distance to merge of the three vehicles. The observable point is 100 meters before the merging point, where mainline and ramp drivers can see each other at the first time during merging. In this trip, drivers rely more on the HMI for longitudinal speed control because they have no visual information about the merging condition. After passing the observable point, when they have line-of-sight to other merging vehicles, they would usually trust more on what they see and make their own decision, instead of relying on the HMI.

During 0 – 8 seconds in V-B, RV accelerates to close its gap with the string on the mainline, and MV1 and MV2 keep a relative constant speed. During this period, both RV and MV2 consider MV1 as their leader. At 8 seconds, MV2 switches its leader from MV1 to RV and decelerates. During 8 – 15 seconds, the gap between MV1 and MV2 has been generated for RV to merge in. During 15 – 20 seconds, the speed of RV and MV2 converge to the speed of MV1, which means the cooperative longitudinal speed adjustment of three vehicles has already been completed before they actually conduct the lane change behaviors.

There are many approaches to measuring driving safety, such as time-to-collision, time-exposed time-to-collision, time integrated time-to-collision, crash index, headway, time-to-accident, and post-encroachment time to name a few [29]. However, these methods are not applicable in this implementation because they require all the vehicles to be driven on the same lane with potential conflicts. In this implementation, MV1 is not affected by the merging behavior happened behind, while MV2 and RV are initially on different lanes without direct confliction. Thus, we adopt the speed variance as an

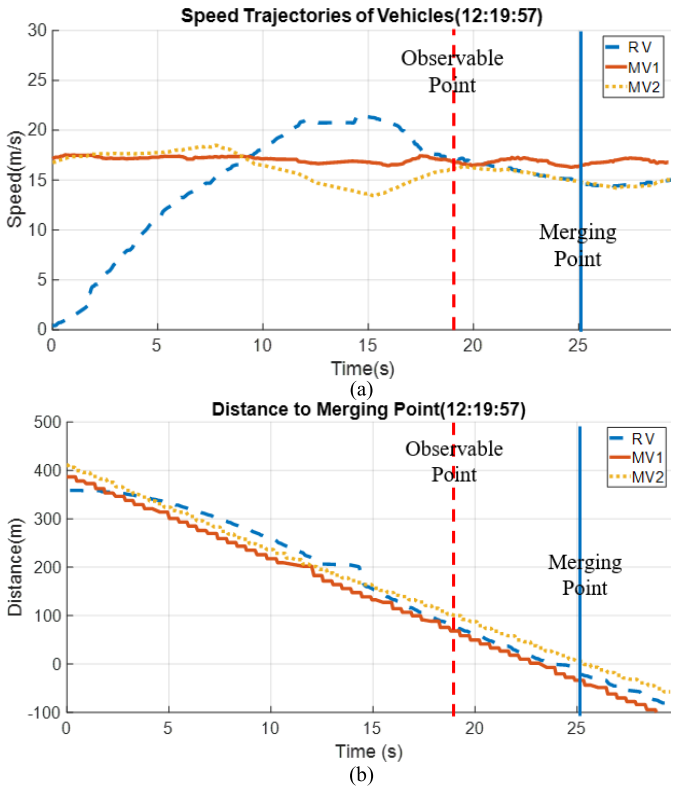


Fig. 10. Cloud-based cooperative merging scenario at on-ramp.

alternative safety measurement method. Since the accidence rates increase with increased speed variance for all classes of roads [30], a safe merging environment can be created by reducing the speed variance. The speed variance of MV2 between the "deactivate" point and the merging point is an indicator of how the upstream traffic safety is affected by the merging behavior.

During the normal merging process, both mainline drivers and ramp driver have to modify their speed, hence the speed variance of the involved vehicles will increase. The average speed variances of MV2 of during four baseline trips is 2.163, while the value is decreased to 0.705 in the four cooperative merging trips. The results show a reduction of 67.41% in terms of average speed variance, proving that the cooperative merging approach is safer than the baseline scenario.

B. Fuel and Emissions Results

For every trip, the fuel and pollutant emissions are analyzed by this Performance Evaluation module on the cloud server with the open-source "MOVESTAR" fuel and emission model [31]. To minimize the biases, MV1 is excluded in this estimation since it tracks a constant speed during the whole merging process and has no significant impacts.

The average fuel consumption and pollutant emissions of involved vehicles are calculated for both baseline and the proposed cooperative merging method, which is shown in Table II. A reduction up to 31.21% in pollutant emissions and a reduction of 7.45% in fuel consumption can be obtained after implementing the proposed model comparing to the baseline, respectively.

TABLE II
ENERGY CONSUMPTION AND POLLUTANT EMISSION RESULTS

	CO (g/km)	HC (g/km)	NO_X (g/km)	CO_2 (g/km)	Fuel (g/km)
Baseline	0.88	0.0088	0.031	242.8	76.038
Proposed	0.61	0.0056	0.025	224.7	70.375
Reduction	31.21%	35.71%	20.00%	7.46%	7.45%

TABLE III
STATISTIC RESULT OF COMMUNICATION DELAY

Average	Minimum	Maximum	75^{th} Percentile	99^{th} Percentile
80 ms	36 ms	3861 ms	88 ms	247 ms

C. Communication Results

A comprehensive test is conducted to measure the delay of 4G/LTE-based V2C communication. In the test, in order to measure only the communication time, all the algorithms are wiped out on the server with only the communication module remained to eliminate the computation time.

The communication delay is defined by a back-and-forth time differences of a "handshake" message. First, the ego vehicle sends the message to the target vehicle and records the current time t_1. Since the target vehicle is designed to reply immediately, when the ego vehicle receives the reply message and at time t_2, the communication delay is calculated as $t_2 - t_1$.

A total of 17717 communication delay samples are recorded during the field implementation from five different trips. Fig. 11(a) shows the distribution of communication delay under its 99^{th} percentile. Majority of the value (65%) locate in the range of 54-81 ms, where this 27 ms span only takes up 12.6% of the whole span. Shown in Table III, the 99^{th} percentile of communication delay is 247 ms, which is smaller than the update frequency of the HMI (3 Hz, 333 ms). Since the computation of the algorithm takes only 30 ms, 99% of the V2C communication is quicker than HMI update timeframe and will not undermine the performance of the algorithm.

According to Fig. 11(b), delay surges appear occasionally, and this 1% of high delay cannot be neglected since the highest delay reaches 3861 ms. Bringing in a Kalman filter or simply keeping the original speed suggestion are the solutions to handle this problem.

Additionally, we also intend to conduct a comparison test regarding the communication methods, i.e., 4G/LTE-based V2C communication versus DSRC-based V2V communication. We install two DSRC OBUs (Savari MobiWAVE 1000) on two of the vehicles (i.e., MV1 and RV), and allow them to send information to each other while the vehicles are conducting the cooperative merging. However, during the merging process, there is a major period of time (more than 5 seconds) that the information cannot be successfully transmitted between these two OBUs. That period mostly happens when MV1 is on the top of the overpass while RV is on the ground (which can be roughly shown as V(a)). As a similar phenomenon is reported by [32], a box trailer will likely "block" the DSRC signal broadcast since it requires line-of-sight for reception. Such performance show that DSRC-based V2V

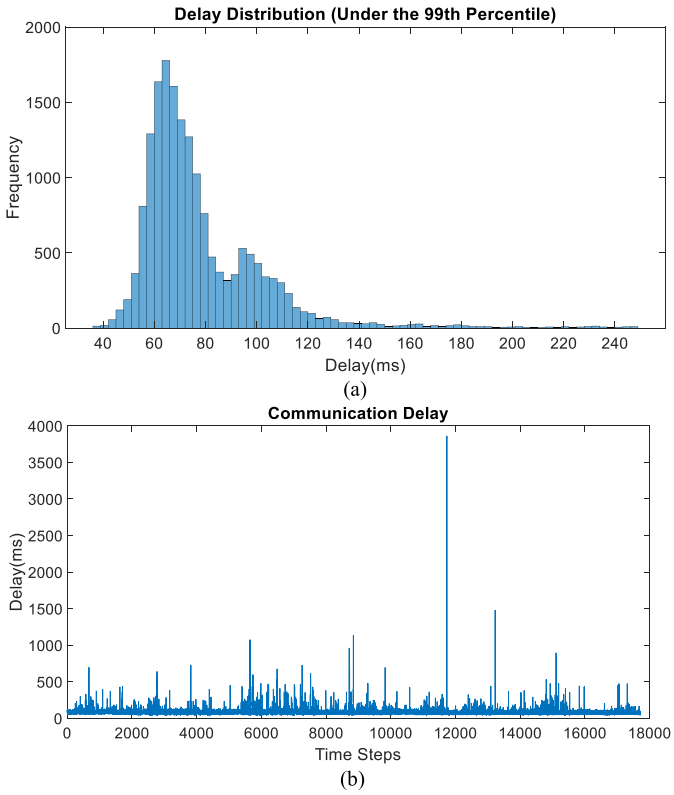


Fig. 11. Communication delay results: (a) Time series of communication delay and (b) Distribution of communication delay under its 99th percentiles.

might not be a good communication method for scenarios like this, when different vehicles are not on the same altitude, or there are major obstructions between two OBUs.

DSRC-based V2I communication might be a good alternative to solve the non-line-of-sight issue of V2V communication, but V2I communication with roadside units also have disadvantages of transmission range, computation power, and data storage compared to V2C communication with cloud computing. In this study, computation power and data storage are essential to conduct the human behavior modeling.

VI. Conclusion and Future Work

In this study, a cooperative ramp merging system has been developed using a V2C Digital Twin approach. Field implementation in the real world has been conducted with three typical passenger vehicles, which shows the proposed system improves current ramp merging scenario in terms of safety and environmental sustainability. Specifically, compared with the baseline scenario with no advisory information during the merging process, the proposed system reduces the average speed variance by 67.41%, reduces the pollutant emissions by up to 31.21%, and reduces the fuel consumption by 7.45%, respectively.

As the first few real-world implementations of Digital Twin in the automotive domain, numerous research questions need to be solved along its future development. To name a few, what is the required update frequency of the Digital Twins on the cloud server, so it can tolerate the uncertainties brought by V2C communication delay and packet loss. Additionally, what other modules can be designed and placed on the cloud server, so the functionality of the Digital Twin architecture can be fully utilized. Implementing the proposed V2C Digital Twin approach to other traffic scenarios besides cooperative ramp merging is also one of our future studies.

Acknowledgment

The authors are grateful to Pingbo Ruan, Shangrui Liu, Yejia Liao, Yu Jiang, David Oswald, Dr. Nigel Williams, and Ahmed Abdo for their contributions in the field implementation, as well as Dr. Ahmed Sakr and Dr. Sergei Avedisov for their knowledges on wireless communication.

The contents of this article only reflect the views of the authors, who are responsible for the facts and the accuracy of the data presented herein. The contents do not necessarily reflect the official views of Toyota Motor North America.

References

[1] U.S. Department of Transportation National Highway Traffic Safety Administration. (2019). *Early Estimate of Motor Vehicle Traffic Fatalities in 2018*. [Online]. Available: https://crashstats.nhtsa.dot.gov/Api/Public/ViewPublication/812749

[2] INRIX. (2018). *INRIX: Congestion Costs Each American 97 Hours, $1,348 a Year*. [Online]. Available: http://inrix.com/press-releases/scorecard-2018-us/

[3] U.S. Department of Energy. (2015). *Fuel Wasted in Traffic Congestion*. [Online]. Available: https://www.energy.gov/eere/vehicles/fact-897-november-2-2015-fuel-wasted-traffic-congestion

[4] J. B. Kenney, "Dedicated short-range communications (DSRC) standards in the United States," *Proc. IEEE*, vol. 99, no. 7, pp. 1162–1182, Jul. 2011.

[5] Z. Wang, 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 Intell. Transp. Syst. Mag.*, vol. 12, no. 1, pp. 4–24, Spring 2020.

[6] J. Rios-Torres and A. A. Malikopoulos, "A survey on the coordination of connected and automated vehicles at intersections and merging at highway on-ramps," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1066–1077, May 2017.

[7] R. Scarinci and B. Heydecker, "Control concepts for facilitating motorway on-ramp merging using intelligent vehicles," *Transp. Rev.*, vol. 34, no. 6, pp. 775–797, 2014.

[8] Z. Zhao, Z. Wang, G. Wu, F. Ye, and M. J. Barth, "The state-of-the-art of coordinated ramp control with mixed traffic conditions," in *Proc. 22th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2019, pp. 1741–1748.

[9] A. Uno, T. Sakaguchi, and S. Tsugawa, "A merging control algorithm based on inter-vehicle communication," in *Proc. IEEE/IEEJ/JSAI Int. Conf. Intell. Transp. Syst.*, Oct. 1999, pp. 783–787.

[10] X.-Y. Lu and K. J. Hedrick, "Longitudinal control algorithm for automated vehicle merging," in *Proc. 39th IEEE Conf. Decis. Control*, vol. 1, Dec. 2000, pp. 450–455.

[11] F.-C. Chou, S. E. Shladover, and G. Bansal, "Coordinated merge control based on V2V communication," in *Proc. IEEE Veh. Netw. Conf. (VNC)*, Dec. 2016, pp. 1–8.

[12] Z. Wang, G. Wu, and M. Barth, "Distributed consensus-based cooperative highway on-ramp merging using V2X communications," SAE Tech. Paper 2018-01-1177, Apr. 2018, doi: 10.4271/2018-01-1177.

[13] T.-S. Dao, C. M. Clark, and J. P. Huissoon, "Distributed platoon assignment and lane selection for traffic flow optimization," in *Proc. IEEE Intell. Vehicles Symp. (IV)*, Jun. 2008, pp. 739–744.

[14] J. Rios-Torres and A. A. Malikopoulos, "Automated and cooperative vehicle merging at highway on-ramps," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 4, pp. 780–789, Apr. 2017.

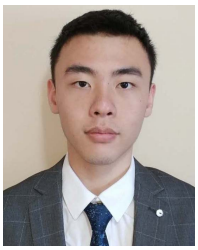
[15] J. Rios-Torres and A. A. Malikopoulos, "Impact of partial penetrations of connected and automated vehicles on fuel consumption and traffic flow," *IEEE Trans. Intell. Veh.*, vol. 3, no. 4, pp. 453–462, Dec. 2018.

[16] J. Ding, L. Li, H. Peng, and Y. Zhang, "A rule-based cooperative merging strategy for connected and automated vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 8, pp. 3436–3446, Aug. 2020.

[17] Z. Wang *et al.*, "Cooperative ramp merging system: Agent-based modeling and simulation using game engine," *SAE Int. J. Connected Automated Vehicles*, vol. 2, no. 2, pp. 1–14, 2019.
[18] V. Milanés, J. Godoy, J. Villagrá, and J. Pérez, "Automated on-ramp merging system for congested traffic situations," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 2, pp. 500–508, Jun. 2011.
[19] K. Raboy, J. Ma, E. Leslie, F. Zhou, K. Rush, and J. Stark, "Cooperative control for lane change maneuvers with connected automated vehicles: A field experiment," in *Proc. Transp. Res. Board 96th Annu. Meeting*, Jan. 2017, p. 21.
[20] S. Hussain, Z. Peng, and M. I. Hayee, "Development and demonstration of merge assist system using connected vehicle technology," Univ. of Minnesota, Minneapolis, MI, USA, Tech. Rep. CTS 19-06, Apr. 2019.
[21] M. S. Ahmed, M. A. Hoque, J. Rios-Torres, and A. Khattak, "Demo: Freeway merge assistance system using DSRC," in *Proc. 2nd ACM Int. Workshop Smart, Auton., Connected Veh. Syst. Services*, Oct. 2017, pp. 83–84.
[22] M. S. Ahmed, M. A. Hoque, J. Rios-Torres, and A. Khattak, "A cooperative freeway merge assistance system using connected vehicles," in *Proc. 97th Annu. Meeting Transp. Res. Board*, Jan. 2018, pp. 1–4.
[23] S. Sanchez-Mateo, E. Perez-Moreno, F. Jimenez, F. Serradilla, A. C. Ruiz, and S. F. Tamayo, "Validation of an assistance system for merging maneuvers in highways in real driving conditions," in *Proc. 16th Eur. Automot. Congr. (EAEC)*, Oct. 2019, pp. 525–531.
[24] Z. Wang, K. Han, B. Kim, G. Wu, and M. J. Barth, "Lookup table-based consensus algorithm for real-time longitudinal motion control of connected and automated vehicles," in *Proc. Amer. Control Conf. (ACC)*, Jul. 2019, pp. 5298–5303.
[25] Z. Wang, G. Wu, and M. J. Barth, "Developing a distributed consensus-based cooperative adaptive cruise control system for heterogeneous vehicles with predecessor following topology," *J. Adv. Transp.*, vol. 2017, Aug. 2017, Art. no. 1023654.
[26] X. Qi, P. Wang, G. Wu, K. Boriboonsomsin, and M. J. Barth, "Connected cooperative ecodriving system considering human driver error," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 8, pp. 2721–2733, Aug. 2018.
[27] Z. Wang *et al.*, "Driver behavior modeling using game engine and real vehicle: A learning-based approach," *IEEE Trans. Intell. Veh.*, vol. 5, no. 4, pp. 738–749, Dec. 2020.
[28] B. T. McCollum, "Analyzing GPS accuracy through the implementation of low-cost cots real-time kinematic GPS receivers in unmanned aerial systems," M.S. thesis, Dept. Syst. Eng. Manage., Air Force Inst. Technol., Wright-Patterson AFB, OH, USA, 2017.
[29] S. M. S. Mahmud, L. Ferreira, M. S. Hoque, and A. Tavassoli, "Application of proximal surrogate indicators for safety evaluation: A review of recent developments and research needs," *IATSS Res.*, vol. 41, no. 4, pp. 153–163, 2017.
[30] N. J. Garber and R. Gadirau, "Speed variance and its influence on accidents," AAA Found. Traffic Saf., Washington, DC, USA, Tech. Rep. ED312438, Jul. 1988.
[31] Z. Wang, G. Wu, and G. Scora, "MOVESTAR: An open-source vehicle fuel and emission model based on USEPA MOVES," 2020, *arXiv:2008.04986*. [Online]. Available: http://arxiv.org/abs/2008.04986
[32] U.S. Department of Transportation. (2020). *Wyoming DOT (WYDOT) Connected Vehicle Pilot Determines Appropriate Tractor-Trailer Antenna Placement and Equipment Configuration*. [Online]. Available: https://www.its.dot.gov/pilots/wyoming_antenna.htm

Xishun Liao (Student Member, IEEE) received the B.E. degree in mechanical engineering and automation from the Beijing University of Posts and Telecommunications in 2016 and the M.Eng. degree in mechanical engineering from the University of Maryland at College Park, College Park, in 2018. He is currently pursuing the Ph.D. degree in electrical and computer engineering with the University of California at Riverside. His research interest includes connected and automated vehicle technology.

Ziran Wang (Member, IEEE) received the Ph.D. degree in mechanical engineering from the University of California at Riverside in 2019. He is currently a Research Scientist with Toyota Motor North America, InfoTech Labs. His research interests include intelligent vehicle technology, including cooperative motion planning and control, driver behavior, and digital twin. He serves as an Associate Editor for *SAE International Journal of Connected and Automated Vehicles*.

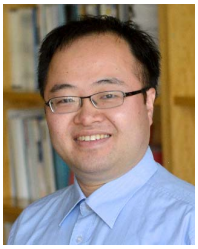


Xuanpeng Zhao received the B.E. degree in electrical engineering from Shanghai Maritime University in 2019. He is currently pursuing the master's degree in electrical and computer engineering with the University of California at Riverside. His research interests include embedded systems, computer vision, and connected and automated vehicle technology.

Kyungtae (KT) Han (Senior Member, IEEE) received the Ph.D. degree in electrical and computer engineering from The University of Texas at Austin in 2006. He was a Research Scientist with Intel Labs and also the Director of Locix, Inc. He is currently a Principal Researcher with Toyota Motor North America, InfoTech Labs. His research interests include cyber-physical systems, connected and automated vehicle technique, and intelligent transportation systems.

Prashant Tiwari received the Ph.D. degree in mechanical engineering from the Rensselaer Polytechnic Institute in 2004 and the M.B.A. degree from the University of Chicago in 2016. He held several leadership positions of increasing responsibilities at GE and UTC Aerospace Systems. He is currently an Executive Director with Toyota Motor North America, InfoTech Labs. He is highly active in Automotive Edge Computing Consortium (AECC) and SAE.

Matthew J. Barth (Fellow, IEEE) received the M.S. and Ph.D. degrees in electrical and computer engineering from the University of California at Santa Barbara in 1985 and 1990, respectively. He is currently the Yeager Families Professor with the College of Engineering, University of California at Riverside, USA. He is also serving as the Director for the Center for Environmental Research and Technology. His current research interests include ITS and the environment, transportation/emissions modeling, vehicle activity analysis, advanced navigation techniques, electric vehicle technology, and advanced sensing and control. He has been active in the IEEE Intelligent Transportation System Society for many years, serving as a Senior Editor for both the *Transactions of ITS* and the IEEE TRANSACTIONS ON INTELLIGENT VEHICLES. He served as the IEEE ITSS President for 2014 and 2015 and is currently the IEEE ITSS Vice President for Finance.



Guoyuan Wu (Senior Member, IEEE) received the Ph.D. degree in mechanical engineering from the University of California at Berkeley in 2010. He is currently an Associate Research Engineer with the Transportation Systems Research Group, Center for Environmental Research and Technology, Bourns College of Engineering, University of California at Riverside, Riverside, USA. His research interests include intelligent and sustainable transportation system technologies, optimization and control of transportation systems, and traffic simulation.