

# Coordination of Virtual and Physical Multiple Vehicles: A Digital Twin Realization Method

1<sup>st</sup> Chunying Yang  
*School of Vehicle and Mobility*  
*Tsinghua University*  
Beijing, China  
ycyacademic@gmail.com

2<sup>nd</sup> Qing Xu  
*School of Vehicle and Mobility*  
*Tsinghua University*  
Beijing, China  
qingxu@tsinghua.edu.cn

3<sup>rd</sup> Jianghong Dong  
*School of Vehicle and Mobility*  
*Tsinghua University*  
Beijing, China  
18801380768@163.com

4<sup>th</sup> Mengchi Cai  
*School of Vehicle and Mobility*  
*Tsinghua University*  
Beijing, China  
cmc18@mails.tsinghua.edu.cn

5<sup>th</sup> Hongmao Qin  
*Wuxi Intelligent Control Research Institute, HNU*  
*Hunan University*  
Wuxi, China  
qinhongmao@hnu.edu.cn

6<sup>th</sup> Jianqiang Wang  
*School of Vehicle and Mobility*  
*Tsinghua University*  
Beijing, China  
wjqlws@tsinghua.edu.cn

7<sup>th</sup> Keqiang Li  
*School of Vehicle and Mobility*  
*Tsinghua University*  
Beijing, China  
likq@tsinghua.edu.cn

**Abstract**—In the last few years, theories and applications of digital twin (DT) have experienced explosive growth. To conduct field implementation of multiple vehicle coordination, we develop a system called DIRS under digital twin paradigm, which contains physical space, cyber space and cloud. Especially, the combination of physical and virtual vehicles enables field validation even under circumstance that the number of accessible physical vehicles is insufficient. To demonstrate system performance, we conduct several experiments in platoon scenario and analyze results in terms of safety and mobility.

**Index Terms**—Digital twin, multi-vehicle cooperative control, field validation

## I. INTRODUCTION

Digital twin (DT), an emerging representation of cyber-physical systems, has attracted increasing attentions recently. It opens the way to real-time monitoring and synchronization of real-world activities with the virtual counterparts [1]. In the field of simulation of Connected and Automated Vehicles (CAVs), related theories and applications have experienced explosive growth in the last several years because of its unique features. Compared to traditional simulation methods, DT paradigm has the following advantages. First, relying on reliable communication and sensor technology, DT system can access real-time data from physical world during running process. Second, since DT paradigm emphasizes two-way interaction between physical and cyber spaces, it is realizable for physical entities to receive feedback from cyber space at a very high frequency to achieve status updates. [1] proposed a general framework of DT system for connected vehicles, which is the first combination of DT and CAVs. [2] used a DT

approach to develop a cooperative ramp merging system and conducted field implementation. Also, they evaluated system effectiveness in terms of mobility and fuel consumption.

With the development of V2X communication, coordinated control of multiple vehicles has been widely studied. [4] proposed a unified multi-vehicle formation control framework for CAVs, and they combined SUMO and Matlab to verify its functions. [5] presented a graph-based vehicle cooperation method to guarantee safety and efficiency at unsignalized intersections. However, field experiment is rarely implemented due to high expenses and demand for extra-large space. Therefore, most researches of multi-vehicle coordination use simulation to deploy, validate and compare impacts of their algorithms on coordination tasks of CAVs.

Conducting field experiment of multi-vehicle coordination using a DT method seems to be a feasible solution. Relying on two-way interaction between physical and cyber spaces and several real vehicles, it is possible to perform coordination tasks even the number of real vehicles is insufficient. [3] proposed a trajectory-following guidance method to chase a virtual target, which inspired method proposed by this study that field experiment of multiple vehicles coordination can be conducted with several real vehicles (or physical vehicles) and a group of virtual vehicles existing in cyber space.

Compared to other recent study on DT system realization and field validation of multi-vehicle coordination algorithms, this study has the following major contributions:

- To our knowledge, this study is the first to propose implementing field validation of multiple vehicles coordination algorithms using a DT method. Moreover, this study

designs and develops a realization system to conduct field experiment even the number of accessible vehicles is insufficient.

- The system offers various ways to support human-machine interaction. For instance, a mixed reality device (Hololens) is leveraged to visualize cyber space and trigger sudden events. Besides, a driving simulator is incorporated to demonstrate driver's perspective.

The remainder of this paper is organized as follows. Section II introduces the realization system in detail, including architecture and components, interaction designing and hierarchical mechanism of vehicle control. In section III, we reveals a case study for platooning scenario and presents the experiment results. The last section concludes this article and shows a prospect of this work.

## II. METHODOLOGY

In this study, we designs and develops a CAVs-orientated digital twin realization system (DIRS) in Tsinghua university, which supports field validation of multiple vehicles coordination methods. To meet such a purpose, this study proposes using combination of physical and virtual vehicles to execute coordination tasks. Specifically, DIRS models two kinds of vehicles in cyber space. One of them is called twin vehicle, which is used to exhibit real-time running status of corresponding physical counterpart, and the other is called cloud vehicle, which is used to simulate behaviors of physical vehicles. Interactions between twin vehicle and cloud vehicle, in turn, can influence decisions and behaviors of physical vehicles. The fact that field experiments of coordination tasks are barely conducted is mainly attributed to financial and safety considerations. However, DIRS seems to be a solution. The combination of physical and virtual vehicles makes it possible to implement field testing even the number of accessible real vehicles is inadequate.

In the section that follow we describe DIRS in more technical detail, providing overviews of architecture and components, human-machine interaction designing and hierarchical mechanism of vehicle controlling.

### A. System architecture and components

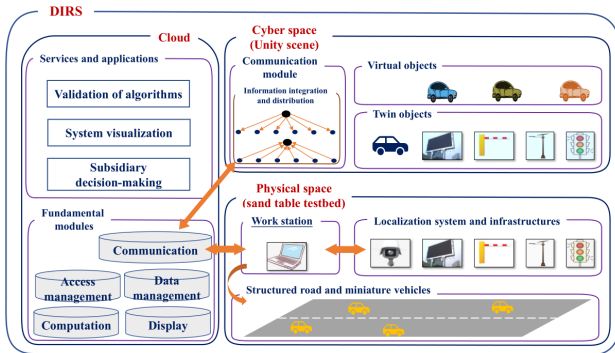


Fig. 1. System architecture

A schematic overview of system architecture is depicted in Fig. 1. DIRS is composed of three major components—physical space, which is in the form of sand table testbed, cyber space and cloud, among which cloud functions as information integration center that links the other two components.

#### Physical Space:

In digital twin paradigm, physical space of DIRS refers to a sand table testbed as shown in Fig. 5(a), which is equipped with miniature vehicles and traffic infrastructures and provides rich environment settings to support field testing.

To achieve mapping from physical to cyber space, vehicle, which is key element of the system and thus is selected to explain its realization process. It should be noted that we assume vehicles are running on a flat surface without height variation for purpose of simplifying computation. Therefore, state parameters of vehicles are composed of position, orientation and speed.

Vision system composed of four overhead cameras is adopted to implement localization and orientation measurement considering relatively stable light conditions of the sand table. Fig. 2(a) presents coverage of the vision system. Several steps are required to realize image-based status determination:

- **Distortion correction**
- **Identifying colors recognition**

Colored blocks with unique combination are attached on the top of individual vehicle and are used to determine real-time running status. Fig 2(c) demonstrates vehicles with identifying colored blocks.

- **Coordinate transformation**

Due to word limitation, more details about these steps are uploads to our system website [9].

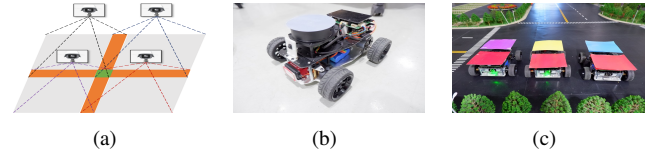


Fig. 2. Physical space. (a) Camera coverage. White, orange and green parallelograms represent area covered by one, two and four cameras respectively in order to guarantee seamless detection. (b) Miniature vehicle used by DIRS. (c) Vehicles with identifying colored blocks on the top

#### Cyber Space:

[6] proposed "Five-dimension model for the DT" and thought cyber space should support simulation, decision making and controlling of its corresponding physical space. When it comes to DIRS, we build cyber space with specific properties in mind. First, giving that open control parameters of miniature vehicles contain speed and steering only, which is insufficient in many cases. As a result, we model vehicles of cyber space with specific kinematics characteristics and other parameters. Second, we desire to visualize cyber space in a intuitional and immersive way. Since game engine is provided with excellent ability of scene rendering and high fidelity, it is adopted by many studies to construct cyber space recently, like [10], [11]. In this study, Unity3D (or Unity) is chosen because

TABLE I  
IMPORTANT PARAMETERS OF PHYSICAL SPACE

Parameters	Value	Unit
Length of sand table	9	m
Width of sand table	5	m
Mass of miniature vehicle	1.4	Kg
Chassis size	200×180×130	mm
Maximum speed	1	m/s
Positioning error of X-axis	±6	mm
Positioning error of Y-axis	±3	mm
Mapping error of X-axis	±3	cm
Mapping error of Y-axis	±5	cm
Mapping error of heading	±2	degree
Image resolution	1920×1080	Pixels per inch(PPI)
fps	30	

of its abundant asset resources and support for mixed reality development. Cyber space built with Unity is demonstrated as Fig. 5(b). Compared to [6], cyber space of DIRS functions as a data generation source and a visualization platform and some of its core functions defined in [6] are migrated to cloud. It is motivated by high access demands relative to game engine's fragile ability to cope with these issues.

To conduct field testing in the absence of sufficient real vehicles, we propose using combination of physical and virtual vehicles to execute coordination tasks. When it comes to modeling of cyber space's vehicles, **twin vehicle** and **cloud vehicle** are provided by DIRS for different purposes. The status-mapping model, called twin vehicle, can be regarded as a simplified particle model and mainly responsible for depicting real-time running status of corresponding physical counterpart. Therefore, it is theoretically sufficient to assign values to parameters of position and orientation. However, we notice that the particle model can not achieve smooth movement under control frequency of 10 Hz. To solve this problem, Kalman filter algorithm is adopted to improve smoothness of trajectory.

The other model named cloud vehicle is constructed to simulate motion characteristic of miniature vehicle. Bicycle model is selected to simplify computation. Corresponding mathematical expression is given by Equation 1 and physical parameters are listed in Table II. The state of the vehicle

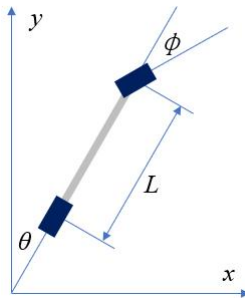


Fig. 3. Bicycle model.  $L$  represents the wheelbase,  $\phi$  represents the steering angle of the front wheel and  $\theta$  represents the yaw angle.

contains  $X$ ,  $Y$ ,  $\theta$  and  $\phi$ , where  $X$  and  $Y$  represent the

TABLE II  
VEHICLE PARAMETERS USED IN THIS PAPER.

$L$ (m)	$\phi$ (degree)	$\theta$ (degree)	$v$ (m/s)	$a$ (m/s <sup>2</sup> )
0.12	$[-45, 45]$	$[-180, 180]$	$[0, 0.2]$	$[-4, 4]$

horizontal and longitudinal position of the center of vehicle's rear axle. The inputs of the vehicle model are  $a$  and  $\phi$ , and the state is calculated as:

$$\begin{cases} \dot{X} = v \cdot \sin(\theta) \\ \dot{Y} = v \cdot \cos(\theta) \\ \dot{\theta} = \tan(\phi) \cdot v/L \\ \dot{v} = a \end{cases} \quad (1)$$

Two vehicle models built in cyber space are demonstrated in Fig. 4

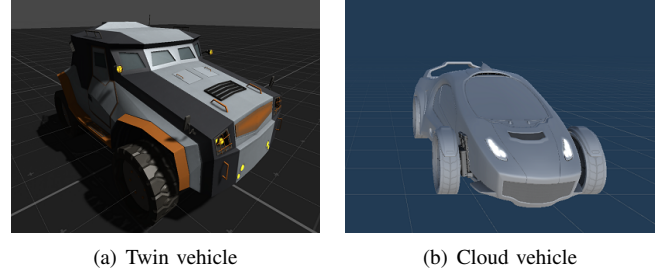


Fig. 4. Vehicle models built in cyber space

#### Cloud:

One of major goals in developing DIRS is to provide a platform where various algorithms can be evaluated without paying attention to internal implementation process. Designing of cloud for DIRS is motivated by high access demand of the system. Since cloud collects data from both physical and cyber spaces, it is feasible for external controller or program to access system data through establishing connection with cloud.

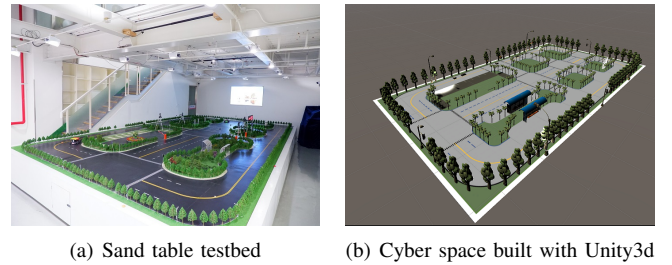


Fig. 5. Physical and cyber counterparts

#### B. System display and HMI designing

Many studies has been carried out to visualize cyber space and to enable human-machine interaction (HMI). [7] developed an augmented reality system to visualize digital twin and display real-time information. In this study, we desire to visualize cyber space in a novel way and to get people involved

for further study, for instance, driver behavior modeling. As a result, several devices are provided to support various ways of HMI: mixed reality device (Hololens), driving simulator and screen display, which can be seen in Fig. 6.



Fig. 6. Several HMI methods designed for DIRS

Scenes rendered by Hololens are in the form of realistic holograms and people wearing Hololens can manipulate objects of cyber space easily, for example, speeding up a virtual vehicle. In particular, it is achievable for Hololens to manipulate miniature vehicles through communication and information exchanges. Hololens is leveraged in DIRS mainly to render overlap of the sand table and its virtual counterpart as shown in Fig. 6(a). Through proper scripting, the **special** effect that cloud vehicles are running on the sand table together with miniature ones, has been achieved and can be visualized with Hololens. Driving simulator provides perspective of one of these vehicles and applies people's control intent onto it. Screen display provides one of the most intuitional presentations of cyber space and also provides convenient interface to adjust observation perspective.

### C. Hierarchical mechanism of vehicle controlling

One of the most salient characteristics of DIRS is that external controller or program can access real-time data of vehicle status and send control message backwards only if connection with cloud is established. Hierarchical mechanism of vehicle controlling is designed for cloud to manage system access and to guarantee safety.

When an external controller or program is added into DIRS for vehicle controlling, three modes are provided by cloud and each of them opens up different control parameters as follows:

- Speed and steering can be assigned to each vehicle directly in Mode 0. Once receiving these values, the sand-box mechanism of miniature vehicles will be launched to guarantee that these values will not exceed the valid range of parameters. When cloud vehicle gets control message, states of next time will be calculated according to Equation 2 and 1.
- Way points sequence, which refers to dispersion of the desired trajectory, is assigned to each vehicle in Mode 1. An example of way points sequence is presented in Fig. ??.
- Node, which denotes the divergence or merging point of vehicle platoon, is set as the target for each vehicle in Mode 2. Fig. 8 shows all nodes of the road map calibrated with unique coordinate.

Algorithm 1 explains the internal running process of the mechanism.

---

#### Algorithm 1: Hierarchical mechanism of vehicle controlling

---

**Input:** mode number (*mode*) and corresponding parameters (*set*  $\langle T \rangle$ ) at time  $k$

**Output:** status of each vehicle at time  $k + 1 : v_i^{k+1}, h_i^k, p_i^{k+1}$ , where  $v$  denotes speed,  $h$  denotes orientation and  $p$  denotes position

---

```

1 Initialization;
2 while true do
3   if mode=0 then
4     Acceleration, one of bicycle model's inputs,
       will be calculated according to Equation 2,
       where  $k_p$  refers to control gain and  $v_{i,ref}$ 
       refers to referential speed for vehicle  $i$  and
       states of cloud vehicle at time  $k + 1$  will be
       calculated according to Equation 1;
5   if mode=1 then
6     Preview control is adopted as the strategy,
       where  $w_{near}$  represents the nearest way point
       to vehicle  $i$  and  $w_{target}$  represents the target
       way point based on  $w_{near}$  and preview
       distance. Once  $w_{target}$  is determined,
       Equation 3 is leveraged to get acceleration
       and  $\phi$  and finally the mechanism traces back
       to Mode 0 to get outputs.
7   else
8     Dijkstra algorithm is leveraged to find the
       shortest path, and the output in this case is
       represented in the form of nodes sequence.
       Next, way points sequence between adjacent
       nodes are generated based on Bezier Curve
       discipline. Finally, the mechanism traces back
       to Mode 1 to get corresponding outputs.

```

---

$$acc = k_p \cdot (v_i^k - v_{i,ref}); \quad (2)$$

$$\begin{cases} v_{i,ref} = w_{target}.speed \\ acc = k_{p1} \cdot (v_i^k - v_{i,ref}) \\ h_{diff} = \arccos\left(\frac{(w_{target}.X - p_i^k.X)}{\sqrt{(w_{target}.X - p_i^k.X)^2 + (w_{target}.Y - p_i^k.Y)^2}}\right) \\ \phi = k_{p2} \cdot (h_{diff} - h_i^k) \end{cases} \quad (3)$$

### III. CASE STUDY AND RESULT EVALUATION

To present more practical details of DIRS, a vehicle level simulation is carried out in platooning scenario. Then the experiment results are analyzed in terms of safety and mobility.

#### A. Platoon management

As Fig. III-A depicts, we manage platoon of DIRS in a hierarchical way. Since DIRS contains three different types of vehicles—cloud, twin and miniature vehicles, we assign three managers to achieve centralized management. Manager



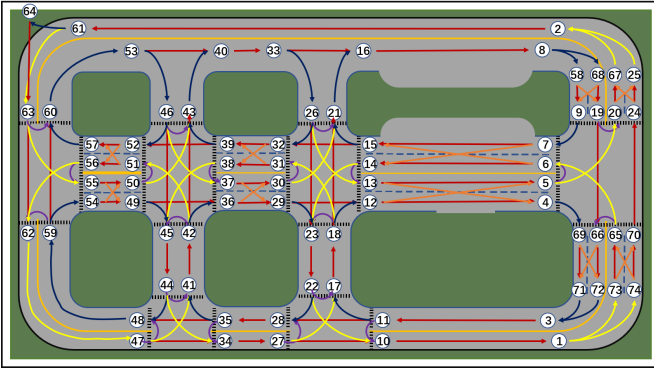


Fig. 7. Calibrated nodes of road map

is mainly in charge of vehicle status collecting and control message distributing. The top manager— *Platoon manager*, is composed of the three managers and functions as a delegate between platoon and external controller.

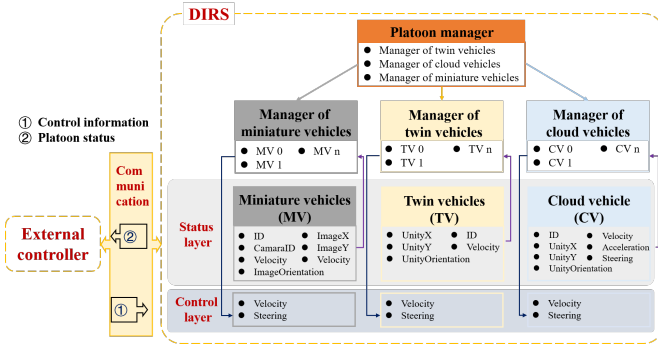


Fig. 8. Platoon management strategy

### B. Testing Scenario Designing

To analyze impacts of control modes on system performance and to present HMI features of DIRS, two scenario are designed to support experiments.

1) *Platoon following Scenario*: Scenario that platoon drives in a rectangular loop is chosen and is used to analyze impacts of control modes on system performance. Platoon's stability, which is indicated by the degree of amplified oscillations when the leading vehicle changes speed dramatically, is selected as an indicator. To meet such a purpose, we let velocity of leading vehicle fluctuate in form of square waves, where upper and lower bounds are set to 0.2m/s and 0.1m/s respectively and period is set to 4s. As Fig. ?? describes, briefly, there is a seven-vehicle platoon that keeps driving in a rectangular loop, and three of them are twin vehicles, 1, 3 and 5, respectively. The remaining are cloud ones. During running process, information flow topology of platoon is demonstrated in Fig. 10(c), where for each following vehicle, not only can it receive status from its front member, but from leading vehicle as well.

During implementation process, Matlab is selected as an external controller to regulate movement of all following

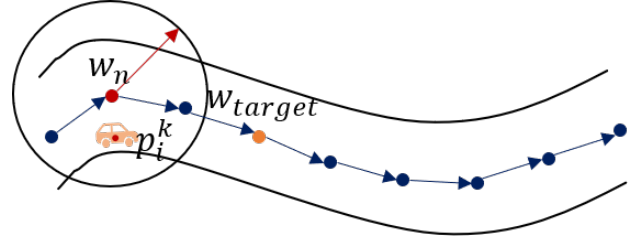


Fig. 9. Preview control, strategy adopted by Matlab in Mode 1, where red dot represents the nearest way point relative to  $p_i^k$ , and radius value of the black circle is equal to value of preview distance, and the orange and blue dots represent target and candidate way point respectively.

vehicles. Control strategy of Matlab in Mode 0 fuses the inputs of both spacing headway maintaining and desired speed keeping. The longitudinal controller is designed to form and maintain desired spacing headway and to keep expected speed simultaneously. The longitudinal feedback control law of vehicles is given as:

$$a_i = k_{pi}(X_i - X_{ref,i}) + k_{vi}(v_i - v_{ref,i}) \quad (4)$$

where  $a_i$  is the control input of vehicle  $i$  and  $k_{pi}$  and  $k_{vi}$  are feedback gains.  $X_{ref,i}$  and  $v_{ref,i}$  are referential longitudinal position and speed of the vehicle.

Control strategy of Matlab Under Mode 1 is influenced by preview control and Algorithm 2 explains it in detail.

#### Algorithm 2: Preview control

**Input:** Position ( $p_i^k$ ) of vehicle  $i$  at time  $k$   
**Output:** status of each vehicle at next time:  $v_i^{k+1}$ ,  $h_i^{k+1}$ ,  $p_i^{k+1}$ , where  $v$  denotes speed, and  $h$  denotes orientation and  $p$  denotes position

- 1 Find the nearest way point  $w_n$  among the set ( $W$ ) of all way points, and determine the target way point ( $w_{target}$ ) based on  $w_n$  and preview distance ( $d$ );
- 2 **begin**
- 3   **for**  $w_i \in W$  **do**
- 4      $dis = \text{Distance}(w_i, p_i^k)$ ;
- 5     **if**  $dis \leq d$  **then**
- 6       Index of  $w_i$  in  $W$  ( $I_i$ );
- 7       Index of  $w_n$  in  $W$  ( $I_n$ );
- 8       **if**  $I_i > I_n$  **then**
- 9           $w_{target} \leftarrow w_i$ , and Equation 3 is adopted to calculate outputs;
- 10       **break**.

2) *Obstacle Avoidance Scenario of Platoon* : We design scenario of platoon obstacle avoidance for three purposes. First, we wish to prove that decision-making and motion of miniature vehicles can be effected by inexistent cloud members. Second, we wish to reflect one of the most distinguishing features of DIRS that remote perception and centralized decision-making are achievable if cloud can integrate global information. Finally, obstacle can be generated by intent

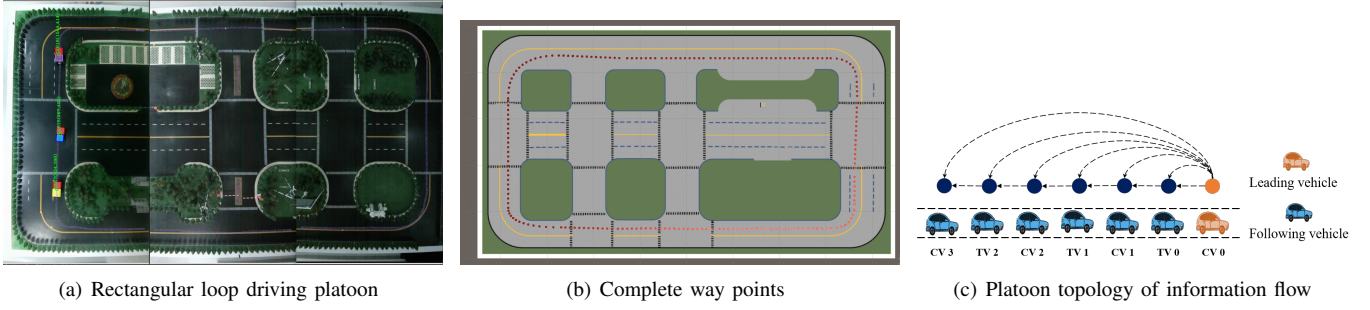


Fig. 10. Platooning scenario. From left to right, where 10(a) demonstrates camera perspective of way points set. 10(b) represents its cyber counterpart. 10(c) describes platoon's information flow topology, where PLF topology [12] is adopted.

TABLE III  
PARAMETER EXPLANATIONS OF FIG.??

Number	From	To	Frequency(Hz)	Main content
①	PS	Cloud	2	Real-time statuses of vehicle 1, 3 and 5
②	Cloud	PS	10	Control messages for vehicle 1, 3 and 5
③	CS	Cloud	10	Real-time statuses of vehicle 0, 2, 4 and 6
④	Cloud	CS	10	Control messages for vehicle 0, 2, 4 and 6
⑤	Matlab	Cloud	10	Real-time statuses of all vehicles
⑥	Cloud	Matlab	10	Control messages for all vehicles

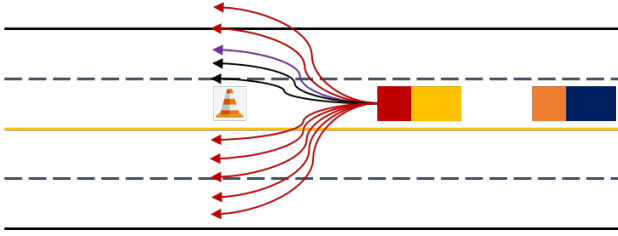


Fig. 11. Local trajectory candidates

captured on Hololens, which is one of the most important applications of Hololens in DIRS.

### C. Result analysis

Figure ?? presents the experiment results. As shown in Figure ?? that values of spacing headway oscillate around the expected value( in green) and that the larger the distance from leading vehicle, the more significant the fluctuation phenomenon is. While Figure ?? demonstrates that when speed of leading vehicle varies in the form of square waves, speeds of following members show the same trend. Similarly, the larger the distance from leading vehicle, the more significant the overshoot phenomenon was.

## IV. CONCLUSION AND FUTURE WORK

In this study, a CAV-orientated digital twin realization system(DIRS) is proposed and developed to facilitate experiments containing both field and simulation validation. Besides, several novel ways are designed to support HMI. To verify effectiveness of the system, we conducted an experiment under car-following scenario, which is simple but functions as the foundation of complex ones, like platoon formation control and collaborative intersection control that we would like

to conduct later. Simulation results demonstrated that DIRS is fully capable to support validation with various optional control parameters.

As a customized simulation platform, several problems need to be solved along its future application: 1): scenarios or experiments could be implemented should be further enriched and evaluated to demonstrate its capability. 2): model errors and impact of time-delay to the system should be quantified, since these factors would influence performance of the system directly. 3): DIRS provides various ways to support HMI, but how to get people more involved is still challenging and needs further considerations.

## V. ACKNOWLEDGMENT

### REFERENCES

- [1] Wang Z, Liao X, Zhao X, et al. A digital twin paradigm: Vehicle-to-cloud based advanced driver assistance systems[C]//2020 IEEE 91st Vehicular Technology Conference (VTC2020-Spring). IEEE, 2020: 1-6.
- [2] Liao X, Wang Z, Zhao X, et al. Cooperative Ramp Merging Design and Field Implementation: A Digital Twin Approach based on Vehicle-to-Cloud Communication[J].
- [3] Chen Q, Wang X, Yang J, et al. Trajectory-following guidance based on a virtual target and an angle constraint[J]. Aerospace Science and Technology, 2019, 87: 448-458.
- [4] Xu Q, Cai M, Li K, et al. Coordinated formation control for intelligent and connected vehicles in multiple traffic scenarios[J]. IET Intelligent Transport Systems, 2021, 15(1): 159-173.
- [5] Chen C, Xu Q, Cai M, et al. A Graph-based Conflict-free Cooperation Method for Intelligent Electric Vehicles at Unsignalized Intersections[J]. arXiv preprint arXiv:2103.14290, 2021.
- [6] Tao F, Zhang H, Liu A, et al. Digital twin in industry: State-of-the-art[J]. IEEE Transactions on Industrial Informatics, 2018, 15(4): 2405-2415.
- [7] G. Schroeder et al., "Visualising the digital twin using web services and augmented reality," in Proc. IEEE 14th Int. Conf. Ind. Informat., Poitiers, France, 2016, pp. 522-527, doi: 10.1109/INDIN.2016.7819217.

- [8] A. Stubbs, V. Vladimerou, A. T. Fulford, D. King, J. Strick and G. E. Dullerud, "Multivehicle systems control over networks: a hovercraft testbed for networked and decentralized control," in *IEEE Control Systems Magazine*, vol. 26, no. 3, pp. 56-69, June 2006, doi: 10.1109/MCS.2006.1636310.
- [9] <https://github.com/yangchunying123/DIRS>
- [10] Wang Z, Wu G, Boriboonsomsin K, et al. Cooperative ramp merging system: Agent-based modeling and simulation using game engine[J]. *SAE International Journal of Connected and Automated Vehicles*, 2019, 2(2).
- [11] Liao X, Zhao X, Wu G, et al. A Game Theory Based Ramp Merging Strategy for Connected and Automated Vehicles in the Mixed Traffic: A Unity-SUMO Integrated Platform[J]. *arXiv preprint arXiv:2101.11237*, 2021.
- [12] Li S E, Zheng Y, Li K, et al. An overview of vehicular platoon control under the four-component framework[C]//2015 IEEE Intelligent Vehicles Symposium (IV). IEEE, 2015: 286-291.