PERFORMANCE TEST OF V2X VEHICLE COLLISION WARNING ALGORITHM BASED ON COMBINATION OF VIRTUAL REALITY FUSION

Baojian HAN*, Yu ZHANG, Yunxiang LIU, Jianlin ZHU, Qing ZHANG

Proposing a performance testing methodology for vehicle collision warning algorithm at intersections, this approach addresses the issues of the key V2X technology being challenging to simulate with real-scene data in virtual simulations, alongside the high-risk factor, elevated testing cost and limited space during the real vehicle verification process. The proposed method involves constructing a simulation environment based on PreScan and Matlab/Simulink, to verify the algorithm's performance. The sensory data obtained from an on-board unit (OBU) installed in a test vehicle on actual roads is then collected and imported into two indoor OBUs to evaluate the algorithm's efficiency further. The test outcomes reveal that combining real and virtual data with simulated data substantially enhances the efficiency of the algorithm testing and ensures security while further validating its effectiveness in genuine scenarios.

Keywords: internet of vehicles; vehicle collision; warning algorithm; virtual reality fusion

1. Introduction

The fast-paced growth of intelligent networked vehicles has amplified the significance of V2X (Vehicle to Everything) technology, a vital component that has gained increasing attention in the industry, becoming a popular research direction both domestically and internationally. The Internet of Vehicles represents the Internet of Things' typical application in the transportation systems domain [1]. As a crucial enabling technology in the intelligent transportation field, V2X utilizes wireless communication to enable intelligent information exchange between vehicles and other vehicles, roads, people, and the cloud, thereby easing traffic congestion, lowering traffic accidents, enhancing road management efficiency, and providing users with comprehensive services that are safe, comfortable, energy-saving, and efficient [2-4]. Moreover, V2X is an essential technical support for attaining genuinely safe, high-level autonomous driving, and unmanned driving [5].

As urbanization accelerates worldwide, road traffic issues are becoming

* School of Computer Science and Information Engineering, Shanghai Institute of Technology, Shanghai, 201418, China, corresponding author's e-mail: 877535678@qq.com

increasingly severe, with approximately 40% of traffic accidents transpiring at intersections [6]. Among them, the encounter of vehicles at intersections is a relatively intricate and precarious driving scenario [7], where the utilization of V2X communication technology can effectively lower the incidence of traffic accidents [8]. By applying the collision warning algorithm based on V2X technology, the algorithm can warn the driver in a timely manner when the vehicle is in danger of collision, using the current position status of the vehicle and other information, consequently improving the efficiency and safety of vehicle traffic [9-10]. The collision warning algorithm's performance is particularly crucial. To ensure the algorithm's availability, effectiveness, and stability, it is necessary to test and evaluate the algorithm before its practical application. Currently, two primary test methods exist. The first is the virtual simulation test, which encompasses Hardware in the Loop (HIL) simulation; however, there is a gap between the simulated and real scenes, and the real test data cannot be entirely simulated, which diminishes the test results' confidence [11-12]. The second is the real vehicle test, which carries potential safety hazards and high scene construction costs, while also having relatively fixed parameter configurations. Additionally, problems such as long cycle times, low efficiency, and difficulties in reproducing test conditions may arise [13-14].

In light of the aforementioned issues, this paper has proposed an algorithm performance verification method that blends virtual and real elements. This method selects the intersection as a typical application scenario, initially employing simulation software to model and test the algorithm, followed by importing real scene data collected by the real vehicle into OBU to further verify algorithm performance. By combining the two test methods, the shortcomings of a single test method can be overcome. This not only meets the advantages of high security, low cost, and strong repeatability in the test but also can evaluate the performance of the algorithm in the real environment, providing a more reliable basis for the algorithm's practical application.

2. Experimental Rationale

2.1 Vehicle Collision Scenario at an Intersection

This paper has chosen the scenario of vehicles encountering each other at common intersections in urban areas. Specifically, it models the situation in which vehicles move straight or turn right at intersections that are controlled by traffic lights. The traffic lights for the north-south direction are set to green, while those for the east-west direction are set to red. The trajectory of the vehicles is illustrated in Fig. 1, and there is a risk of collision when two vehicles merge into the same lane.

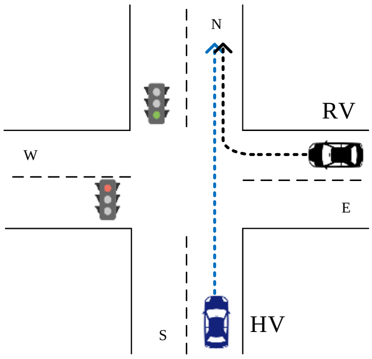

Fig. 1 Intersection crash scene

2.2 Intersection Collision Warning Algorithm

The algorithm initially converts the geodetic coordinate system in which the vehicle is situated into a two-dimensional plane coordinate system. For ease of calculation, the converted plane coordinate system is then rotated to a fresh plane coordinate system, with the heading angle of the host vehicle serving as the positive half-axis of the Y-axis. During vehicle collision detection, the algorithm takes into account the size of the vehicles and calculates the time difference between the two vehicles reaching the collision point using the new vehicle coordinates and heading angle. A danger threshold is established, and when the time difference between the two vehicles reaching the collision point falls below this threshold, the algorithm issues an alert.

2.2.1 Coordinate system transformation and coordinate system rotation

As the GPS coordinate information procured from the OBU is established on the longitude and latitude information of the geodetic coordinate system, for simplification of calculation, the geodetic coordinate system is transformed into a plane coordinate system where the true north direction acts as the positive semi-axis of the Y-axis. Due to the OBU device's limited communication distance, the earth's curved surface effect is considered negligible, and the earth's size can be regarded as that of a sphere with a radius of R. Assuming the HV longitude and latitude coordinates are (Lon1, Lat1), with a heading angle of θ, and the RV longitude and latitude coordinates are (Lon2, Lat2), with a heading angle of β, the HV coordinates after conversion are located at A (0, 0), and the RV coordinates are (m, n).

$$\mathrm{m} = \frac{(Lon1 - Lon2) \times R \times \pi \times \cos(Lat1)}{180} \tag{1}$$

$$\mathrm{n} = \frac{(Lat1 - Lat2) \times R \times \pi}{180} \tag{2}$$

To streamline the process, the plane coordinate system is further rotated into a system where the HV heading angle serves as the positive semi-axis of the Y-axis. Post-rotation, the HV coordinate A (0, 0) stands as the origin, and the RV coordinate B (x, y).

$$\theta_1 = \frac{\theta \times \pi}{180} \tag{3}$$

$$x = m \times \cos\theta_1 - n \times \sin\theta_1 \tag{4}$$

$$\mathrm{y} = m \times \sin\theta_1 + n \times \cos\theta_1 \tag{5}$$

2.2.2 The time when the two vehicles reach the collision point

As shown in Fig. 2, HV is located at the origin A (0, 0), the heading is the positive direction of the Y-axis, RV is located at point B (x, y), the heading angle is $360^{\circ} - \alpha$, and the intersection of the two vehicle headings is point C, then the distance from HV to the collision point C is AC, the distance from RV to collision point C is BC, and the distance between the two vehicles is $\mathrm{BA} = \sqrt{x^2 + y^2}$, and there is a relationship $AC = \frac{NA}{\sin\alpha}$, $BC = \frac{MB}{\sin\alpha} = \frac{x}{\sin\alpha}$, and the straight line equation of BC is:

$$Y - y = -\cot\alpha(X - x) \tag{6}$$

Then the length of NA is the distance from point A to line BC:

$$NA = \frac{\left|-x\cot\alpha - y\right|}{\sqrt{1+\cot^2\alpha}} \tag{7}$$

$$AC = \frac{\left|-x\cot\alpha - y\right|}{\sin\alpha\sqrt{1+\cot^2\alpha}} \tag{8}$$

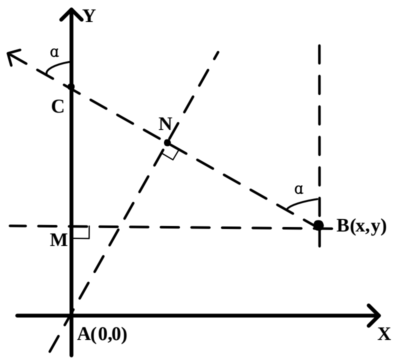

Fig. 2 Collision point time calculation

Let v_hv be the speed of HV, v_rv be the speed of RV, the time when the host vehicle HV reaches the collision point is t_hv, and the time when the remote vehicle RV reaches the collision point is t_rv.

$$t\_hv = \frac{AC}{v\_hv} \tag{9}$$

$$t\_rv = \frac{BC}{v\_rv} \tag{10}$$

2.2.3 Vehicle Collision Detection and Collision Avoidance Algorithms

In this paper's collision detection, the automobile is perceived as a circle with the OBU device's location as the center and a radius of 1.25m, as exhibited in Fig. 3. M and N denote the centers of the two circles. In the event of an intersection of the two circles, which means that the distance MN between the two vehicles is under 2.5m, it is deemed that a vehicular collision is impending. Assuming that the velocity of the automobile passing through the intersection is 5m/s, that is, when $|t\_hv - t\_rv| < 0.5\text{s}$, there is an imminent risk of a collision between the two vehicles.

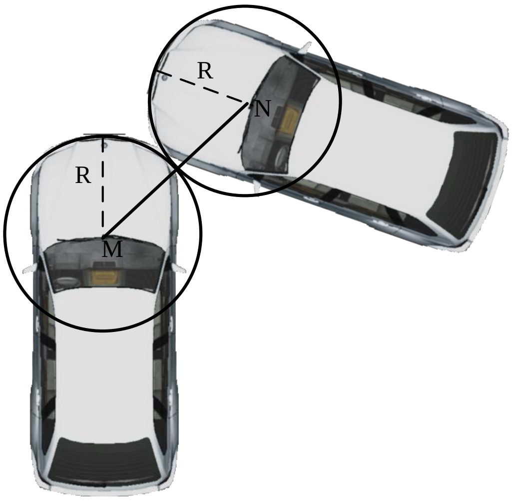

Fig. 3 Vehicle Collision Detection

According to the source cited in reference [15], alerting drivers of potentially hazardous situations, whether visible or not, can effectively decrease the incidence of collision accidents. The warning message should be conveyed promptly, allowing the drivers ample time to react, but not too early. As per literature [16], the safety benefit is greatest when the warning message is issued 4 to 8 seconds before the impending danger.

After comprehensive consideration, this paper sets the warning to the driver when $t\_hv < 5\text{s}$. The algorithm steps are as follows.

Algorithm 1 Collision avoidance algorithm for vehicles at intersections
Input Two-vehicle location status information
Output Algorithm warning
1. The speed of the host vehicle is v_hv, the heading angle is θ, and the latitude and longitude coordinates are (Lon1, Lat1).
2. The speed of the remote car is v_rv, the heading angle is β, and the latitude and longitude coordinates are (Lon2, Lat2).
3. Calculation formula (1) ~ formula (5)//Convert coordinates.
4. The coordinates of the host car are A (0, 0), and the coordinates of the remote car are B (x, y).
5. Calculation formula (6) ~ formula (10) // Calculate the time for the two vehicles to reach the collision point.
6. The time for the host vehicle to reach the collision point is t_hv.
7. The time for the remote vehicle to reach the collision point is t_rv.
8. if $|t\_hv - t\_rv| < 0.5$ && $t\_hv < 5$ then
9. Give an alert.
10. end.

3. Experimental design

3.1 Experimental procedure

(1) PreScan is a software simulation tool developed by TASS International [17-18] for the automotive industry. In this paper, PreScan simulation software is utilized in combination with Matlab/Simulink virtual simulation. The vehicle collision scenario illustrated in Fig. 1 is created within the PreScan software, and the vehicle trajectory, speed, and road conditions are all predetermined. The initial configuration of the two cars is depicted in Fig. 4. The leading car is moving straight, while the trailing car is making a right turn, thereby creating a potential collision risk as the two vehicles converge into the same lane. The experimental parameters are specified in Table 2, and the road parameters are aligned with the actual scene. This group of data is referred to as the Simulation Data Group (SDG).

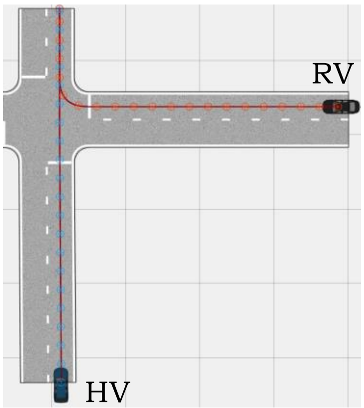

Fig. 4 PreScan Simulation Scenario

Table 1

PreScan Simulation Parameters

Parameters	HV	RV
driving direction	straight	turn right
initial velocity	5m/s	5m/s
acceleration	uniform speed	uniform speed
intersection width	8m	8m
vehicle length	4.79m	5.21m
vehicle width	2.17m	2.03m
lane width	3.5m	3.5m

(2) The algorithm described in this paper is implemented by combining individual modules within Matlab/Simulink. The PreScan software provides real-time vehicle coordinates, heading angles, and other status information that serves as input to the algorithm. The algorithm then computes warning results in real-time using Simulink and saves the outputs. The experiment was conducted 100 times, with the distances and driving trajectories of the two vehicles to the stop line being adjusted each time.

(3) To simulate actual vehicle driving, it is necessary to collect driving data. The On-Board Unit (OBU) is a critical node that forms the basis of the Internet of Vehicles. It is responsible for enabling information exchange among V2V (Vehicle to Vehicle), V2I (Vehicle to Infrastructure), V2P (Vehicle to Pedestrian), and V2N (Vehicle to Network). The OBU plays a central role in most information interaction activities in the Internet of Vehicles system and is poised to be the primary component of intelligent connected cars. An OBU device is comprised of communication chips, communication modules, terminal equipment, security chips, V2X protocol stack, and V2X application software. In this study, the OBU equipment was used to record the vehicle trajectory data of the straight and turning vehicles in the aforementioned scenarios. Fig. 5 displays the two OBU

devices employed in the experiment.

Fig. 5 On-board unit

The vehicle trajectory data was recorded on the road adjacent to the experimental building of the school, with the OBU equipment installed in the experimental vehicle. The same OBU equipment was utilized to record the data twice. The first vehicle traversed the intersection in a straight line from east to west, while the second vehicle turned right at the intersection from north to west. The NMEA (National Marine Electronics Association) data recorded by the OBU is displayed in Fig. 6(a). Through data analysis, information such as vehicle coordinates, speed, and heading angle can be extracted. The average speed of the first vehicle is 4.63 m/s, while the average speed of the second vehicle is 5.46 m/s. The two NEMA data were imported into the Google Earth Pro software to display the driving trajectories of the two vehicles, as illustrated in Fig. 6(b).

```
$GPGGA,055256.10,3050.648097,N,12130.628771,E,1,03,1.9,-1.3,M,10.0,M,,*47
$GPVTG,244.3,T,248.8,M,12.8,N,23.6,K,A*28
$GPRMC,055256.10,A,3050.648097,N,12130.628771,E,12.8,244.3,060621,4.5,W,A,V*6
B
$GPGSA,A,2,06,14,28,,,,,,,,,2.1,1.9,1.0,1*2D
$GPGGA,055256.20,3050.647964,N,12130.628318,E,1,03,1.9,-1.2,M,10.0,M,,*44
$GPVTG,246.3,T,250.9,M,13.3,N,24.6,K,A*2F
$GPRMC,055256.20,A,3050.647964,N,12130.628318,E,13.3,246.3,060621,4.5,W,A,V*61
$GPGSA,A,2,06,14,28,,,,,,,,,2.1,1.9,1.0,1*2D
$GPGGA,055256.30,3050.647799,N,12130.627955,E,1,03,1.9,-1.1,M,10.0,M,,*46
$GPVTG,244.9,T,249.5,M,13.1,N,24.2,K,A*25
$GPRMC,055256.30,A,3050.647799,N,12130.627955,E,13.1,244.9,060621,4.5,W,A,V*6
A
$GPGSA,A,2,06,14,28,,,,,,,,,2.1,1.9,1.0,1*2D
$GPGGA,055256.40,3050.647631,N,12130.627608,E,1,03,1.9,-1.2,M,10.0,M,,*46
```

Fig. 6 (a) NEMA data

Fig. 6(b) Vehicle trajectory

(4) Utilizing two OBU apparatuses indoors to emulate V2V communication. V2V communication is an inter-vehicular information exchange that is executed through the on-board terminal. The vehicle transmits its own details, such as speed, position, and driving status, to the outside via the on-board terminal, and simultaneously receives pertinent information transmitted by other vehicles [19-21]. This information determines whether there is a risk of a traffic accident and then prompts the driver to take proactive risk avoidance measures. Through V2V communication, vehicles are no longer isolated entities, but can communicate in real-time, significantly reducing traffic accidents and achieving vehicle monitoring and management. Introduce the two vehicle trajectory data into two OBU devices separately and start the two OBU devices indoors simultaneously for algorithm validation, which is utilized to simulate the scenario of actual vehicles passing by at intersections. As the OBU device routinely broadcasts its own data information and receives data information broadcasted by other OBU devices, the data information of the HV and the data information of the RV are employed as the input data for the algorithm. When the predetermined algorithm warning threshold is met, the algorithm will produce an alert at the terminal interface in the OBU device. This ensemble is defined as the True Data Group (TDG).

3.2 Analysis of simulation results

Upon the algorithm issuing its first alert, the distance between the two vehicles at that moment is calculated as the algorithm's evaluation index. As depicted in Fig. 7, out of the 100 experiments, the maximum first warning distance is 31.39 meters, the minimum first warning distance is 24.53 meters, and the mean first warning distance is 29.97 meters. When the initial velocity of the two cars is 5 meters per second, the car's braking distance is less than 10 meters [22]. Hence, the driver has sufficient reaction time to apply the brakes, which can prevent traffic accidents from occurring. This validates the efficacy of the algorithm proposed in this paper.

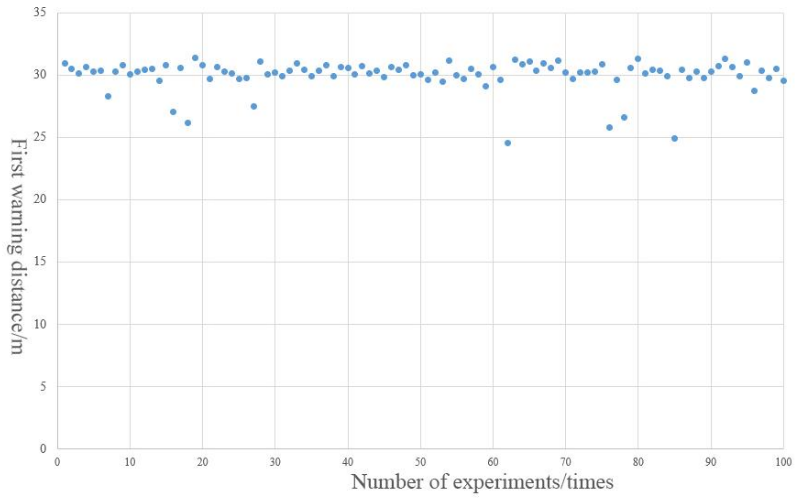

Fig. 7 Simulation results

3.3 Algorithm Verification on Real Vehicle Trajectory Data

The algorithm implemented in the simulation software is transplanted into the OBU, and the warning indicator is denoted as icw_alert. The warning indicator is expressed in two values, where 1 signifies that the algorithm has issued an alert, and 0 indicates that the algorithm has not issued an alert. The experimental outcomes of the actual vehicle trajectory data are illustrated in Fig. 8. The algorithm's first warning distance is 32.02 meters. At this point, the host vehicle's driving speed is 16.99 kilometers per hour, and the remote vehicle's driving speed is 19.22 kilometers per hour. The time T_rv for the host vehicle to reach the collision point is 2.37 seconds, and the time for the remote vehicle to reach the collision point T_hv is 1.93 seconds. The time difference between the two vehicles reaching the collision point is 0.44 seconds, which satisfies the algorithm alert condition. Since a vehicle cannot maintain a constant speed during the actual driving process, the TDG's first warning distance differs somewhat from that of the SDG. Nevertheless, the outcomes of the two are relatively close, thus validating the algorithm's practicality in real-world environments.

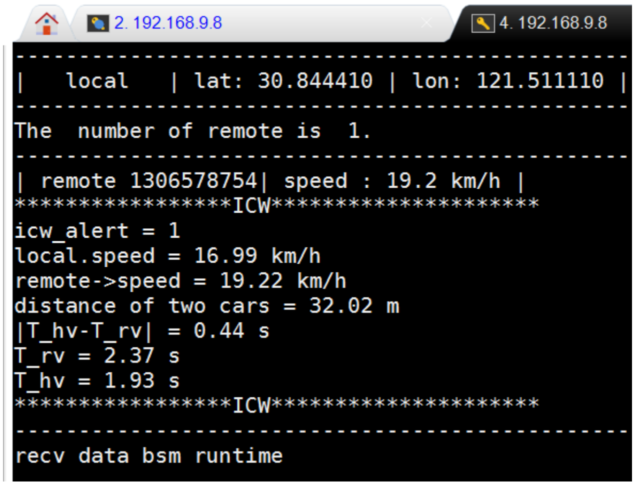

Fig. 8 Algorithm warning output interface

4. Conclusion

The validation of an algorithm with actual vehicles entails a number of cost-related factors, including vehicle cost - since at least two fully functional vehicles are needed for testing, which is generally a costly investment. This entails not only the purchase cost but also expenses related to maintenance and repair in the event of accidental damage during the testing period. Human costs also play a key role since numerous human resources, including test engineers, drivers, and data analysts, are required for actual vehicle testing. The associated expenses include personnel training and salaries. Additionally, various testing equipment and tools, such as sensors, recorders, and computers, are required for actual vehicle testing, which adds to the overall cost. An appropriate testing site, such as a test track or proving ground, is also required, and rental and maintenance costs must be taken into account. Finally, conducting actual vehicle tests can be time-consuming, encompassing preparation time, testing time, data analysis, and result reporting time.

In summary, verifying algorithms with actual vehicles presents several issues, including potential safety hazards, high construction costs, long test cycle times, and difficulty reproducing test conditions. However, the method employed in this paper, which combines virtual and real testing to evaluate algorithm performance, significantly reduces the costs associated with actual vehicle testing. By importing real-world data collected by actual vehicles into the OBU device, algorithm performance is simulated and verified indoors. This approach can overcome the limitations of a single testing method, meet high security standards, and offer low costs and strong test repeatability. Moreover, it can evaluate algorithm performance in real-world environments, providing a more reliable foundation for practical applications. Additionally, algorithm parameters can be optimized to improve algorithm accuracy and robustness.

To address the testing requirements of the alert algorithm in the intersection environment of the typical V2X technology application scenario, this paper proposes a test method that combines both virtual and real elements. The first stage involves testing the algorithm's performance through virtual simulation. In the second stage, the algorithm's performance is further verified using indoor simulation of actual vehicle driving data. By comparing the performance of the two-stage algorithm, the applicability of the algorithm in both virtual and real environments is verified. This method leverages the benefits of both virtual simulation testing and real vehicle testing, providing a feasible testing approach for evaluating the algorithm's effectiveness in practical applications.

In comparison to traditional algorithms (like blind spot / angle technology) that rely solely on calculating the distance between the centroids of two vehicles to determine a collision, the collision warning algorithm proposed in this paper can accurately compute the time when the two vehicles will reach the collision point in real-time. This can provide an effective technological means to enhance road traffic safety and reduce accidents.

Future research will delve deeper into the specific indicators for evaluating the algorithm and formulate a series of scientific experimental standards to explore the integrity of the algorithm in practical applications.

Acknowledgment

The research is sponsored by the Shanghai Natural Science Foundation of China (21ZR1462600). The authors would like to thank the editors, anonymous reviewers, and those who helped us for their valuable suggestions and constructive comments on this paper.

R E F E R E N C E S

[1] *Sehla, K., Nguyen, T. M. T., Pujolle, G., & Velloso, P. B.,* Resource Allocation Modes in C-V2X: From LTE-V2X to 5G-V2X. IEEE Internet of Things Journal, vol. 9, no. 11, pp. 8291-8314, 2022. DOI: 10.1109/JIOT.2022.3159591

[2] *Ding, Q., & Lv, Y.,* Research on resource scheduling method based on LTE-V2X direct connect communication. 2020 15th IEEE International Conference on Signal Processing (ICSP). Beijing, China, IEEE, vol. 1, pp. 679-683, 2020. DOI: 10.1109/ICSP48669.2020.9321062

[3] *Molina-Masegosa, R., Gozalvez, J., & Sepulcre, M.,* Comparison of IEEE 802.11p and LTE-V2X: An Evaluation with Periodic and Aperiodic Messages of Constant and Variable Size. IEEE Access, vol. 8, pp. 121526-121548, 2020. DOI: 10.1109/ACCESS.2020.3007115

[4] *Choi, J., & Kim, H.,* A QoS-aware congestion control scheme for C-V2X safety communications. *2020 IEEE Vehicular Networking Conference (VNC)*. New York, NY, USA, IEEE, pp. 1-4, 2020. DOI: 10.1109/VNC51378.2020.9318385

[5] *Moubayed, A., Shami, A., Heidari, P., Larabi, A., & Brunner, R.,* Edge-enabled V2X service placement for intelligent transportation systems. IEEE Transactions on Mobile Computing, vol. 20, no. 4, pp. 1380-1392, 2020. DOI: 10.1109/TMC.2020.2965929

[6] *Wu, Z., Yang, J., & Huang, L.,* Study on the Collision Avoidance Strategy at Unsignalized Intersection based on PreScan Simulation. Procedia-Social and Behavioral Sciences, vol. 96, pp. 1315-1321, 2013. https://doi.org/10.1016/j.sbspro.2013.08.149

[7] *Guan, Y., Ren, Y., Li, S. E., Sun, Q., Luo, L., & Li, K.,* Centralized Cooperation for Connected and Automated Vehicles at Intersections by Proximal Policy Optimization. IEEE Transactions on Vehicular Technology, vol. 69, no. 11, pp. 12597-12608, 2020. DOI: 10.1109/TVT.2020.3026111

[8] *Ni, Y., Wang, S., Xin, L., Meng, Y., Yin, J., & Sun, J.,* A V2X-based Approach for Avoiding Potential Blind-zone Collisions between Right-turning Vehicles and Pedestrians at Intersections. 2020 IEEE 23rd International Conference on Intelligent Transportation Systems (ITSC). Rhodes, Greece, IEEE, pp. 1-6, 2020. DOI: 10.1109/ITSC45102.2020.9294501

[9] *Thunberg, J., Bischoff, D., Schiegg, F. A., Meuser, T., & Vinel, A.,* Unreliable V2X Communication in Cooperative Driving: Safety Times for Emergency Braking. IEEE Access, vol. 9, pp. 148024-148036, 2021. DOI: 10.1109/ACCESS.2021.3124450

[10] *Hui, F., Xing, M., Guo, J., & Tang, S.,* Forward collision warning strategy based on vehicle-to-vehicle communication. Journal of Computer Applications, vol. 41, no. 2, pp. 498-503, 2021. DOI: 10.11772/j.issn.1001-9081.2020060773

[11] *Obermaier, C., Riebl, R., & Facchi, C.,* Fully Reactive Hardware-in-the-Loop Simulation for VANET Devices. 2018 21st International Conference on Intelligent Transportation Systems (ITSC). Maui, HI, USA, IEEE, pp. 3755-3760, 2018. DOI: 10.1109/ITSC.2018.8569663

[12] *Dong, X., Nan, Y., Chen, B., Feng, S., Liu, X.D.,* Research on Algorithm Evaluation Model Based on C-V2X Application Scenarios. Automotive Digest, no. 5, pp. 13-17, 2020. DOI: 10.19822/j.cnki.1671-6329.20190005

[13] *Guanetti, J., Kim, Y., & Borrelli, F.,* Control of connected and automated vehicles： State of the art and future challenges. Annual Reviews in Control, vol. 45, pp. 18-40, 2018. https://doi.org/10.1016/j.arcontrol.2018.04.011

[14] *Park, C., Chung, S., & Lee, H.,* Vehicle-in-the-Loop in Global Coordinates for Advanced Driver Assistance System. Applied Sciences, vol. 10, no. 8, pp. 2645, 2020. https://doi.org/10.3390/app10082645

[15] *Yan, X., Zhang, Y., & Ma, L.,* The influence of in-vehicle speech warning timing on driver collision avoidance performance at signalized intersections. Transportation Research Part C, Emerging Technologies, vol. 51, pp. 231-242, 2015. https://doi.org/10.1016/j.trc.2014.12.003

[16] *Wan, J., Wu, C., & Zhang, Y.,* Effects of lead time of verbal collision warning messages on driving behavior in connected vehicle settings. Journal of Safety Research, vol. 58, pp. 89-98, 2016. https://doi.org/10.1016/j.jsr.2016.07.003

[17] *Hidayatullah, M. R., & Juang, J. C.,* Centralized and distributed control framework under homogeneous and heterogeneous platoon. IEEE Access, vol. 9, pp. 49629-49648, 2021. DOI: 10.1109/ACCESS.2021.3068968

[18] *Son, T. D., Nguyen, L., & Van Der Auweraer, H.,* Learning Control Applications for Autonomous Driving in Extreme Maneuver Scenarios. 2021 21st International Conference on Control, Automation and Systems (ICCAS). Jeju, Korea, IEEE, pp. 974-979, 2021. DOI: 10.23919/ICCAS52745.2021.9649993

[19] *Akbar, R.Z.,* Performance Analysis FSR and DSR Routing Protocol in VANET with V2V

and V2I Models. 2020 3rd International Seminar on Research of Information Technology and Intelligent Systems (ISRITI). Yogyakarta, Indonesia, IEEE, pp. 158-163, 2020. DOI: 10.1109/ISRITI51436.2020.9315367

[20] *Luu, D. L., Lupu, C., Alshareefi, H., & Lupu, M.*, Performance Analysis of V2V and V2I Channels for Autonomous Smart Car. 2022 23rd International Carpathian Control Conference (ICCC). Sinaia, Romania, IEEE, pp. 382-386, 2022. DOI: 10.1109/ICCC54292.2022.9805862

[21] *Moshkov, V. V., Badin, A. D., & Guminskiy, O. A.*, Research of Characteristics of Radio Technologies of V2V/V2P Systems. 2022 Conference of Russian Young Researchers in Electrical and Electronic Engineering (ElConRus). Saint Petersburg, Russian Federation, IEEE, pp. 64-68, 2022. DOI: 10.1109/ElConRus54750.2022.9755670

[22] *Wu, M.*, The relationship between braking distance and speed. Highways & Automotive Applications, no. 3, pp. 46-49, 2010.