A SIMULATION FRAMEWORK FOR EXPLORING THE IMPACTS OF VEHICLE PLATOONS ON MIXED TRAFFIC UNDER CONNECTED AND AUTONOMOUS ENVIRONMENT

by

Dian Yuan

A dissertation submitted to the Faculty of the University of Delaware in partial fulfillment of the requirements for the degree of Doctor of Philosophy in Civil Engineering

Winter 2023

A SIMULATION FRAMEWORK FOR EXPLORING THE IMPACTS OF VEHICLE PLATOONS ON MIXED TRAFFIC UNDER CONNECTED AND AUTONOMOUS ENVIRONMENT

by

Dian Yuan

Approved: ____
Jack Puleo, Ph.D.
Chair of the Department of Civil and Environmental Engineering

Approved: ____
Levi T. Thompson, Ph.D.
Dean of the College of Engineering

Approved: ____
Louis F. Rossi, Ph.D.
Vice Provost for Graduate and Professional Education and
Dean of the Graduate College

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: ____________________

Ardeshir Faghri, Ph.D.
Professor in charge of dissertation

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: ____________________

Tracy DeLiberty, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: ____________________

Mark Nejad, Ph.D.
Member of dissertation committee

I certify that I have read this dissertation and that in my opinion it meets the academic and professional standard required by the University as a dissertation for the degree of Doctor of Philosophy.

Signed: ____________________

Shangjia Dong, Ph.D.
Member of dissertation committee

ACKNOWLEDGMENTS

I am eternally grateful to all the people who have supported and helped me not only in my research but in my daily life as well. My life and research would not have as many accomplishments without their help. Therefore, I am honored to acknowledge them here.

I am heartily thankful to my advisor Professor Ardeshir Faghri for his brilliant and patient guidance not only in my research but also in life. His enthusiasm and encouragement for research kept me motivated toward my commitment to my thesis work. Working under his guidance has constantly shaped me and helped me become a better scholar. I'd like to express deep gratitude for all his efforts and kindness.

I also want to thank my committee members, Professor Tracy DeLiberty, Professor Mark Nejad, and Professor Shangjia Dong, for giving profound scientific advice and for providing research insights from different perspectives for carrying out works in this paper. I appreciate every meeting with them that inspired me with this thesis.

Last but not least, I want to thank all of my family members. I want to express my appreciation to my wife, Dr. Jingya Qin, for encouraging me all the time. I would also like to thank my parents for supporting me with advanced education. Then, a big thanks to my sweetest baby daughter. May happiness and health be with her always. I am also grateful to my parents-in-law for helping me with taking care of my newborn daughter.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

ABSTRACT

Vehicle platooning, first studied as an application of Intelligent Transportation Systems (ITS), is increasingly gaining attention in recent years as autonomous driving and connected vehicle technologies advance. When being platooned, vehicles communicate within the platoon and operate with coordination to maintain a relatively steady state status with each other and with the outside. The major goal of this study is to build a conceptual simulation framework to help with exploring the impacts of connected and autonomous vehicle platoons on the existing traffic. The first part of this work effort is reviewing autonomous and connected vehicle technologies for depicting the functional structure of a platooning-ready connected and autonomous vehicle (CAV) platform. Then models and simulation tools are reviewed to break down the simulation framework into two levels – vehicle level and traffic level. The vehicle-level model provides in-depth modeling of CAVs and platooning modules. The traffic-level simulator provides the simulation of the existing traffic with the built CAV platoons. The simulation framework has been developed by integration and usage of GIS, MATLAB/Simulink, SUMO, and OMNeT++. GIS tools are used to gather the necessary traffic data. MATLAB/Simulink serves as the platform for vehicle-level modeling and simulation. SUMO and OMNeT++ are used to build the traffic and communication simulations, respectively. The completed model was used to conduct two case studies based on a section of the US Interstate Highway in order to explore the impacts of CAV platoons on existing traffic. The results indicate that, with the existing traffic pattern and infrastructure design, traffic can be improved after the introduction of CAV platoons, even after taking into consideration the rate of traffic growth. Moreover, deploying dedicated lanes and separating CAV platoon traffic from the non-

platooning traffic can benefit the traffic using such output as the travel speed/time and delay measures. However, using such new traffic patterns and infrastructure designs is not recommended for a low percentage of CAV platoon traffic.

Chapter 1

INTRODUCTION AND BACKGROUND

In the last decade, the development of autonomous vehicle technologies has become one of the most popular and studied topics in the transportation field. The safety benefits of autonomous vehicles and eliminating human errors have been vastly reported in literatures [e.g., 1, 2, 3]. Meanwhile, as the field of wireless communication advances, connecting vehicles in an optimized manner seems to become more achievable than ever before [4], [5], [6]. Instead of treating these technologies as two parallel paths to the future, researchers have started combining the two types of technologies to improve the overall performance of the traffic stream [7], [8], [9], [10]. Platooning connected and autonomous vehicles (CAVs) is one such combination. Some of the benefits of CAVs reported in the literature include improving safety, reduction in energy consumption and pollution, mitigating human errors, and increasing the capacity of the overall transportation system [11], [12], [13]. Back in the 1990s, the leading Intelligent Transportation System (ITS) program - the California Partners for Advanced Transit and Highways (PATH), sponsored many studies focusing on vehicle platooning [14], [15]. Nowadays, benefiting from advanced technologies, vehicle platooning under connected and autonomous vehicle environments has become an application with great potential.

The ultimate goal of an Automated Driving System (ADS) is to replace the human driver thoroughly. Prior to exploring the ADS technologies, a simple review of what a driver needs to handle while driving could help us to understand what functionalities the ADS needs to deliver. Assuming a trip is planned, after buckling up, drivers need to observe the surrounding environment of the vehicles as well as assess the status of the vehicles. Drivers need to keep their vehicles moving safely. By turning the steering wheel and applying the brake or gas pedal, the drivers operate the vehicles to complete the planned trips. From another perspective, no matter if it's based on personal experiences, signs or maps, the drivers need to understand what the traffic environment is, in order to make sure they can arrive at the destination at the expected time point, with some uncertainty, since the traffic can be much worse than the driver's expectation. However, besides gathering information visually or aurally, mobile apps or installed navigation functions on vehicles provide drivers with detailed information about the traffic.

Based on the scenario described above, the three main tasks of an ADS can be deduced. The first is to see and understand both the environment and the vehicle itself. The next task is to make decisions to guarantee the vehicle runs on the planned route with safety, comfort, and efficiency. The last task is to generate control commands according to the driver's decisions and execute them to maintain and change the speed and direction of the vehicle. Thus, this study adopts a functional architecture of AV technologies which commonly consists of three parts: i) perception, ii) planning, and iii) control [16]. A more detailed review of these technologies is presented in the second chapter.

Facilitating connecting vehicles and transmitting information among them requires many different technologies. Inter-vehicle communication (IVC) is one of those. IVC was considered an essential part of the ITS technologies back in the 1990s [17], [18]. It was believed that it could help drivers and vehicles with "seeing" further and broader. Recently, as the internet of things (IOT) concept has received a great deal of attention, connected vehicle technologies have advanced substantially in the last two decades. The three common phrases used in connected vehicle environments are vehicle-to-vehicle (V2V), vehicle-to-infrastructure (V2I), and vehicle-to-everything (V2X) communication. V2V communication indicates the data exchange among vehicles. It enables vehicles to perceive the driving environment with better extension. V2I communication means the vehicle is connected to devices installed on infrastructures and receives proper information [17]. For example, the traffic light with Infrastructure-to-Vehicle (I2V) devices can broadcast the time remaining on the green light phase, so approaching connected vehicles can make decisions more effectively depending on the remaining time for a safe passage. If the answer is no, the vehicle can smooth its deceleration and improve both safety and comfort. Reversely, the traffic light control system can be optimized by gathering states (e.g., position and speed) of approaching CAVs [19]. Both in the United States and Europe, V2V and V2I have designated licensed spectrums. The corresponding bandwidths for communications are from 5.850 to 5.925 GHz and from 5.855 to 5.905 GHz [20], [21], [22]. V2X communication is a broader concept that is used to describe the communications between a vehicle and all other entities that may influence mutually or one-sidedly [23], [24]. V2X communications include vehicle-to-pedestrian (V2P) communication, vehicle-to-grid communication (V2G), vehicle-to-cloud (V2C) communication, and others. V2G indicates the communication between electric-driven vehicles and the power grid. This technology is developed to optimize power storage and usage macroscopically [25], [26], [27]. V2C is an emerging concept. It emphasizes the communication between vehicles and extensive data resources. It can give vehicle access to more information and higher computing power [28], [29].

As essential technologies advance, the pace of realizing such applications are much more practical than ever before. Thus, for transportation planners, designers, engineers, and policymakers, how platooning CAVs will impact existing traffic and the system under different situations will be a crucial problem that needs to be explored. When new traffic patterns and/or new design patterns of infrastructure are implemented, it's necessary to estimate how CAV platoons will impact the traffic. Additionally, even though vehicle manufacturers and agencies have considered their plans to platoon CAVs [30] [31], it is still challenging to depict how the future will be with such components in the traffic. Also, establishing a real-life experiment may require great capital input and raise safety and ethical issues. Thus, a simulation framework can provide a solution to pre-assess the impacts without massive investment.

Software engineers and developers have released products that can simulate traffic with CAV traffic. But there aren't many for CAV platoons. Moreover, existing tools may provide limited adaptivity to be modified for exploring the impacts of the platoon with different CAV-related technologies or platooning algorithms. Thus, the adaptivity of the simulation framework is crucial.

Purpose and objectives

The main problem addressed in this dissertation is the lack of a reliable and flexible simulation tool that evaluates the impacts of CAV platoons on existing traffic.

The framework should have proper capability to adapt to the advancing technologies and changing circumstances. Meanwhile, to improve the precision of modeling CAV platoons, there should be two levels of simulation. The first is a lower level framework needs to simulate how a CAV, or a platoon will behave from a self-oriented perspective. The second is a higher-level simulation to help users explore the potential influences of the CAV platoons on the existing traffic stream, or to different traffic patterns with various infrastructure designs.

To address the main problem, this study is divided into four interrelated topics:

i. What is the functional architecture of a connected and autonomous vehicle? What types of technologies are essential when studying a platoon-enabled CAV?

ii. How will vehicles be platooned? How will a platoon work effectively in a connected and autonomous environment? What will the functional architecture be for platooning CAVs?

iii. How are platoon-enabled CAVs and CAV platoons modelled? How should the traffic with CAV platoons be simulated?

iv. What should the model do once platooned CAVs are introduced to the network? How are these impacts evaluated?

In order to solve a transportation problem properly, the temporal and spatial dimensions. For roadway transportation, a plan or a design is usually established to serve for about ten years. Therefore, in this rapidly changing period of roadway transportation industry, considering some issues priorly are necessary. On the other hand, CAVs are highly expected to dominate the roadway infrastructures in the future based on the trends of relevant technology development. Despite being a popular topic in multiple disciplines, the governments of major economic entities also released plans, regulations and standards on both developing and implementing CAV technologies from a real-world perspective [32] [33] [34] [35]. Vehicle platooning, one of applications of CAVs and cooperative driving has been discussed extensively in recent years. Developing a tool, or an approach to explore the influences of vehicle platoons on the traffic is not only interesting but is also meaningful in the development of ground transportation plans, designs and technologies.

As such, the main objectives of this study include:

i. Conducting a survey of CAV technologies and applications and depicting a functional architecture of both CAVs and the platooning of CAVs.
ii. Developing vehicular models representing different types of vehicles: a) human-driven vehicles, b) non-connected autonomous vehicles, c) connected and autonomous vehicles with platoon functions.
iii. Modeling platoon-enabled CAVs operations in a microscopic environment.
iv. Using traffic simulation tools to establish traffic-level simulation scenarios based on different shares of platoon-enabled CAVs and other types of vehicles and different roadway designs.
v. Integrating vehicular modeling results, traffic simulation models and testing scenarios to simulate the traffic with CAV platoons and exploring the impacts of CAV platoons on the traffic environment under different scenarios. Verifying simulation framework with sample tests.
vi. Validating the framework by conducting case studies based on real-life infrastructures and traffic aspects, and simulating the traffic with different shares of CAV platoons. Evaluating the outputs of the real-world based simulations.

In the next chapter, a comprehensive survey of literature is presented for reviewing the studies focused on Advanced Driving/Driver Assistant (ADA) technologies, Autonomous Driving, Inter-vehicle communication, and vehicle platooning. Modeling CAVs and simulating traffic algorithms are reviewed in Chapter 2. Moreover, the models and simulation theories utilized in this study are also introduced in Chapter 2. In the third chapter, two case studies are conducted to explore the impacts of CAV platoons on traffic flow. Potential changes on both traffic and infrastructures are considered for both cases. The last chapter of the main body of this dissertation summarizes this study and draws a conclusion. Limitations of this study and recommendations for future work are presented as well in the last chapter.

Scope

As was mentioned previously, the main goal of this study is to develop a simulation framework in order to evaluate the impact of CAV platoons on traffic. The studied scenarios use limited access and divided highways. In the USA, this category includes Interstates, Freeways, and Expressways [36]. Thus, roadways belonging to other functional classifications are not covered.

The studied vehicle types are passenger cars and trucks. Based on current studies, these two types of ground vehicles are widely studied as the participants of vehicle platooning. More specifications are given in the case study chapters. When exploring vehicle modeling, the most popular vehicle dynamic models found in the literature are used. The control inputs are limited to acceleration/deceleration and different steering angles. For traditional fossil-fuel-based engine vehicles, such controls are actuated by throttle/brake control systems and steering control systems [37] [38].

The tools for building and validating the simulation framework consist of MATLAB/Simulink SUMO and OMNeT++. In the Simulation chapter, the reasons for using these tools are provided. MATLAB/Simulink is used to model and microscopically simulate the CAVs and the single platoon's behaviors/maneuvers. SUMO is the tool that is used to conduct simulations of traffic macroscopically. OMNeT++ serves as the simulator of communication. Other assumptions are described in more detail throughout the upcoming chapters.

Chapter 2

CONNECTED DRIVING, AUTONOMOUS DRIVING AND VEHICLE PLATOONING TECHNOLOGIES

This chapter includes the following parts: Firstly, the functional architecture of CAV that has been adopted in this study is demonstrated. Then, an overview of implemented and developing technologies for connected driving, autonomous driving, and vehicle platooning technologies are presented. Finally, the overview provides descriptions and explanations of the components which compose the architecture.

Functional Architecture of Connected and Autonomous Driving System

After studying the technologies that have been implemented or are being developed, a functional architecture of connected and autonomous vehicles used in this study is demonstrated below in Figure 1.

As mentioned in the introduction chapter, a viable CAV system should be capable of operating the vehicle in a safe manner with limited human intervention or even without it and transporting the drivers/passengers and goods from the origin to the destination correctly. To achieve such goals, the following types of functions are deemed necessary: i) perception, ii) planning, and iii) control [16]. Perception functions can help with perceiving the events and objects that could influence vehicle driving. Planning functions are the functions that determine what the vehicle should do to complete the trip correctly. Control functions are the functions that guarantee vehicles are behaving as accurately as planned.

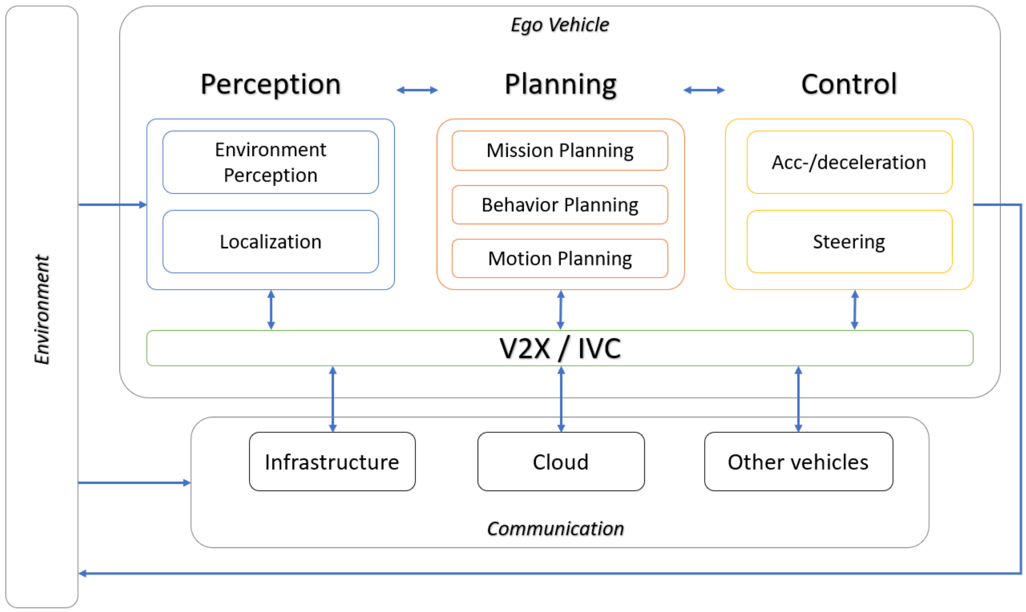

Figure 1. Functional Architecture of a CAV

Implemented technologies

Implemented technologies include popular technologies that are available in the market as driver assistance technologies. The Society of Automotive Engineers (SAE) defined vehicles that activate a single automated system as level 1 driving automation. Table 1 shows the levels of driving automation with the SAE J3016 standard [39]. An integrated system with multiple driver assistance technologies that can work simultaneously is considered a level-2 driving automation system.

As defined by The National Highway Traffic Safety Administration (NHTSA), the Driver Assistant Technologies can be split into four categories regarding their

Table 1. Levels of driving automation (SAE J3016) [39]

Level 0	No Driving Automation (DA)	Drivers are driving whenever driver support features are engaged. Drivers must constantly supervise and must take actions for driving safely.	Features are limited to providing warnings and momentary assistance.
Level 1	Driver Assistance		Features provide steering OR brake/acceleration support.
Level 2	Partial DA		Features provide steering AND brake/acceleration support.
Level 3	Conditional DA	Automated driving features are responsible for driving.	Features can drive the vehicle when all required conditions are met. Drivers must drive when requested.
Level 4	High DA		Features can drive the vehicle when all required conditions are met.
Level 5	Full DA		Features can drive the vehicle under all conditions.

functions [40]. They are collision warning, collision intervention, driving control assistance, and other systems. Table 2 shows commonly implemented technologies to perform such functions. Features listed in Table 2 are commonly integrated to perform a driving assistance. Some combinations of different technologies are implemented to improve the system with more capabilities. One example is the integration of Forward Collision Warning (FCW) and Automatic Emergency Braking (AEB) systems. The two systems work together to diminish the risk that a vehicle rear-ends the vehicle ahead, or mitigate the severity when a collision is unpreventable [41]. Once a potential forward collision is detected, the integrated system activates FCW to alert and remind the driver to take necessary operations to avoid a potential collision. If the driver does not take proper action in time, the system will automatically activate AEB to avoid a collision. Another example is the combination of Rear Cross Traffic Warning (RCTW) and Rear Automatic Braking (RAB). These two systems work jointly to help avoid collisions caused by overlooked objects while the vehicle is in reverse [42]. The mechanics are like the combinations of the FCW and AEB technologies. When cruising within a relatively stable situation, combinations of Adaptive Cruise Control, Lane Keeping Assistance/Lane Centering Assistance technologies can improve the comfort of the experience of both driver and passengers.

Table 2. Driver assistant technologies [40]

Category	Technology	Brief Description
Collision Warning	Forward Collision Warning	Warning driver of a potential crash with the vehicle ahead
	Lane Departure Warning	Warning driver while the vehicle is approaching or crossing lane markers
	Rear Cross Traffic Warning	Warning the driver of potential crash while driving backward.
	Blind Spot Warning	Warning the driver of objects in driver's blind spot
Collision Intervention	Automatic Emergency Braking	Appling brakes or enhancing braking when a forward collision is imminent
	Pedestrian Automatic Emergency Braking	Braking once a pedestrian(s) is detected ahead and a collision is imminent
	Rear Automatic Braking	Braking when a potential crash is imminent while driving backward
	Blind Spot Intervention	Braking or steering (either or both) when a vehicle is detected when ego vehicle is changing lanes
Driving Control Assistance	Adaptive Cruise Control	Adjusting speed automatically to maintain a safe distance from the vehicle ahead
	Lane Centering Assistance	Steering continually to keep the vehicle driving at the middle of its lane
	Lane Keeping Assistance	Steering automatically to keep the vehicle form departing the lane
Other Systems	Automatic High Beams	Activating and deactivating high beams depending on the distance from oncoming vehicle or the vehicle ahead
	Backup Camera	Providing a clear view behind the vehicle
	Automatic Crash Notification	Reporting automatically to an emergency responder when a crash occurred

Developing Technologies

This section reviews studies on the currently developing technologies (including algorithms, techniques, and equipment) related to level-3 or more advanced automated driving. Survey and review publications focusing on technologies of autonomous driving, connected driving, and vehicle platooning are reviewed as the major focuses. In addition, publications introducing innovative technologies are reviewed as supplementary.

Perception

Perception functions can be broken down into two major parts – environment perception and localization [43] [44] [45]. Environment perception functions refer to the functions that help the user/vehicle with perceiving the immediate surroundings. Localization functions help user/vehicle to locate itself regarding a reference. In this section, prevailing technologies are studied.

Environment perception technologies

The commonly studied technologies in the environment perception area are presented in this section. They include sensing technologies, sensor calibration and fusion technologies. Light detection and ranging (LiDAR), Camera, and Radio detection and ranging (Radar) are the external sensor technologies reviewed in this study.

a) LiDAR

LiDAR is an active sensing technology. By emitting lasers into the surrounding, distances toward surfaces are measured by the travel time of the emitted laser. The detection range of LiDAR sensors is commonly from 10 to 200 meters [46]. The common outputs of LiDAR sensors are point clouds. Each point indicates the relative position of a surface. The scanning strategies of LiDAR are 2D and 3D strategies. 3D scanning can be achieved by spinning the whole scanning device with multiple 2D sensors [44].

Defined by the basic mechanic, LiDAR is superior in ranging to other technologies. The physical information obtained by LiDAR technology is reliable [47]. On the other hand, the mechanics also limits this technology when there is a lack of reflectivity on the scanned surfaces. Adverse weather may affect the functionality of LiDAR as well. Moreover, since LiDAR can hardly provide texture information of scanned surfaces, the collaboration of LiDAR and Camera is usually needed to perform object recognition/tracking [16]. LiDAR technology's overall cost is relatively higher than other technologies [48].

b) Camera

In some studies [16] [48] [49], the camera technology applied in the self-driving area is also called vision technology. Cameras provide semantic description of the surroundings. The functions of camera technology in a connected and autonomous driving system include but are not limited to detecting on-road objects and recognizing lane geometry and traffic signs [16]. The advantages of camera technologies' capability of providing texture information about the scanned object are the foundation of those functions. Camera technology offers two systems for CAVs – monocular camera and stereo camera. Monocular camera solution is inexpensive. Stereo cameras can provide depth information. One of the most significant disadvantages of camera sensors is the dependency on sufficient illumination. Studies have shown that with advanced processing algorithms, a level of mitigation is achievable on resolution decrease under adverse illumination [44] [45].

c) Radar

Radar systems utilized for developing perception functions on CAVs can be classified depending on how far they detect. Two classes are mainly studied. They are short-range radar systems and long-range radar. The definitions of short- and long-range vary in different studies. In most cases, short-range automotive radars can detect up to 30 meters. On the other hand, the number of long-range radars is from 10 to 250 meters [50]. Radar technology is also a ranging technology. The mechanics are like LiDAR technology but emitting and receiving electromagnetic waves instead of lasers [45] [51]. For most applications of radar-based perception technologies, the most common frequencies of electromagnetic waves are 24 GHz, 60 GHz, 77 GHz, and 79 GHz [44] [51]. Radar techniques with such frequencies are known as millimeter-wave (mm-wave) radar technology.

The cost of radar systems is relatively low and provides robustness under inverse conditions. However, one critical disadvantage of radar technology is the difficulty of obtaining high resolution output when being compared to LiDAR technology.

d) Calibration

When multiple sensor modalities are employed, calibrations are necessary to ensure the integrated sensor system performs properly. Calibration methods include intrinsic calibration, extrinsic calibration, and temporal calibration. Intrinsic calibration focuses on sensor-specific parameters, such as the focal lengths of a vision camera [44]. Extrinsic calibrations are conducted to develop mapping among coordinate systems between sensor modalities. Temporal calibration is the process of synchronization of the data streams delivered by multiple sensor modalities.

e) Fusion

Sensor fusion integrates the acquired data from different sensing modalities. It can improve the performance of perception systems consisting of multiple types of sensing modality systematically [44] [51]. For instance, a sensor fusion of cameras and Lidar sensors can mitigate the limitations of vision cameras under inverse illumination conditions.

There are three groups of sensor fusion approaches. They are low-level fusion (LLF), mid-level fusion (MLF), and high-level fusion (HLF) [44] [52]. The LLF can also be called data-level fusion. It means data from sensors are integrated as raw as possible. Such fusion increases the cost and difficulty of data processing. But, as a result, the detection could be more accurate. MLF indicates that the fusion takes place at the feature level. The approach can be simplified as fusing the Information about one or multiple features of a detected object to perform recognition and classification. HLF refers to system-level fusion approaches. Each sensor system performs detection independently and fuses the output with others. HLF approaches have gained popularity due to their feasibility of application compared to the other two groups.

Localization

Localization is another critical component in a CAV's perception system. The main localization tasks are determining the position and orientation of the ego vehicle and estimating its motion. Commonly implemented or studied technologies include mapping, sensor-based, and cooperative technologies [53].

Satellite navigation technologies, such as Global Positioning System (GPS), can provide the global position of the ego vehicle. However, such technologies rely on signal strength. Inertial navigation technologies are sensor-based. The sensors are commonly called as Inertial Motion Units (IMU). By monitoring the motion of the ego vehicle, position information is obtained. A shortcoming of inertial navigation technologies is the error accumulation through the time when vehicles are operated continuously. Moreover, the approaches relying on infrastructure detection (markings or roadway detection) are also used for localizing vehicles [54]. However, such approaches require infrastructure to be built and maintained intensively.

Recently, map-aided localization technologies advanced significantly. One such technology, called Simultaneous Localization and Mapping (SLAM), is a hot topic in this area. The process of SLAM is building maps and using them for localization at the same time [16] [51]. In [54], Bresson categorized SLAM problems into two groups: full and online problems. The similarity between the two types of problems is that they both compute the vehicle's current position over previous positions, map, control inputs, and sensor data. Solving a full SLAM problem needs all previous inputs into account, but solving an online SLAM only considers the information from the last period.

Planning

With sufficient information about the environment, a comprehensive plan should be developed to ensure the vehicle completes transportation tasks safely and efficiently. The main responsibilities of the planning system of an AV can be summarized as: i) planning the route to transport from an initial location to the desired destination, ii) determining the selections of behavior as reactions to events from the environment, and iii) extracting trajectory-level plans from behavior selection. The planning systems of a CAV system are also termed the decision-making systems [16] [43] [55].

Route Planning

A route planning system indicates a system that plans for ensuring the CAV completes the desired transportation tasks at a road network level. Classical shortest path algorithms, such as Dijkstra or A* algorithm, are less practical when dealing with real-world routing problems [56]. But some practical algorithms derived from them with different trade-offs can achieve solving real-world problems in milliseconds with decent computing space requirements [46] [56]. Generally, time consumption, memory space usage, preprocessing, and simplifications of an algorithm are the aspects needed as trade off.

Bast et al. categorized route planning techniques within static road networks into goal-directed, separator-based, hierarchical, and bounded-hop [56]. Goal-directed algorithms are optimized routing algorithms compared to the basic Dijkstra algorithm. Such algorithms prune the map to avoid redundant options by leading the search toward the destination. Separator-based algorithms optimize the solutions by partitioning the map. Overlay maps may be deployed to achieve further optimizations during the process. Hierarchical techniques exploit the classification of the importance of each vertex/arch, and then mitigate the cost by skipping less important vertices/arches. Bounded-hop techniques focus on pre-computed distance between vertices and running multiple “hops” based on the precomputed results to find out the shortest path from the origin to the destination. Multiple algorithms could be utilized to improve the processing speed of route planning [46].

Behavior Selection

Selecting behaviors to maintain that autonomous vehicles are being operated safely and efficiently and mitigating risks when an inverse situation occurs is the main task of a behavior selection module. Recently, Schwarting et al. conducted reviews on autonomous system approaches and categorized behavior selection approaches into four groups [55]. They are game-theoretic, probabilistic, Partially-Observable-Markov-Decision-Process-based, and learning-based approaches.

The game-theoretic approaches, as the name suggests, are based on game theory. Behaviors are selected to maximize the ego vehicle's reward when considering both selected behavior and potential reactions that other objects (such as other road users) will take [57]. Unlike game-theoretic approaches, probabilistic approaches focus more on the estimating the probabilities of upcoming situations and executing the responding behaviors to maximize the benefits of the ego vehicle. Partially Observable Markov Decision Processes (POMDPs) are typically generalizations of Markov Decision Processes (MDPs) [55]. MDPs are discrete-time stochastic control processes. The core of an MDP is a state-action-probability-reward control mechanism [58]. Actions yield probabilities to transition from one state to another. Corresponding rewards are gained after the transitions between states. POMDPs employ observation sets and observation probability sets [55]. Those sets are used to update the present state since the states are not guaranteed as expected and planned in the real world.

Local Planning

This study categorizes local planning systems into two parts – path planning and trajectory planning [16]. Path planning components break down the planned routes into a sequence of configuration states of the vehicle. Trajectory planning is defined as a time-parametrized function that reflects the state changing in time.

Path planning algorithms, according to Gonzalez et al.'s research, can be split into three classes [59]. The first class is a graph-based algorithm. Those algorithms are commonly developed upon the Dijkstra algorithm, the A* algorithm, and the state lattice algorithm. They deploy grids or coordinates and calculate the feasible and optimal solution for transiting the vehicle from one state to the next. States represent not only position, but also information like velocity, angle, acceleration, and heading. Another class of path-planning algorithms is sampling-based algorithms. These algorithms are developed to provide solutions under temporal constraints. One of the robotics methods for path planning, the Rapidly-exploring Random Tree, is employed in some studies as a path planning algorithm for AV [16]. The last class is the interpolation curve algorithms. Interpolation algorithms rely on given waypoints. By inserting new data points, the output results in smoothed trajectories with considerations of environment and constraints.

Control

The control modules, or controllers, in a connected and autonomous driving system are in response to maintaining the vehicle movement according to desired ways planned by planning modules [16]. Most controls aim at two aspects of the vehicle – steering and acceleration/deceleration. By calculating the error among reference trajectory/path and vehicle state, control actuators take corresponding operations to events, e.g., applying emergency braking to avoid collision with an object ahead. Figure

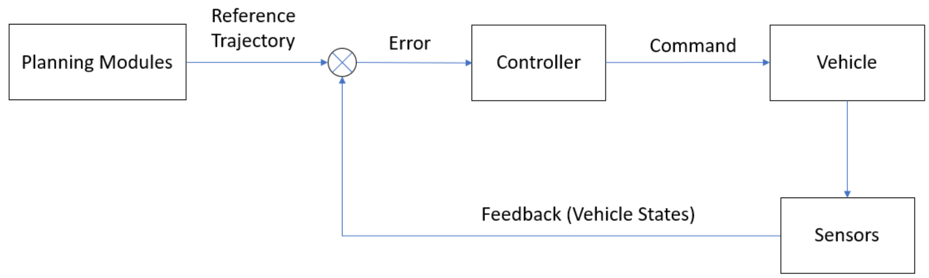

Figure 2. Single-input Single-output Control System

2 briefly demonstrates a single-input-single-output control system, which is a basic schema of a control system. In [60], Amer et al. reviewed control approaches of autonomous vehicles and categorized them into seven classes. They are geometric approaches, kinematic approaches, dynamic control approaches, optimal control approaches, adaptive and intelligent control approaches, model-based approaches, and classic approaches.

Classic control approaches are common approaches deployed for solving real-world control problems. Proportional integral derivative (PID) controller and sliding mode controller (SMC) are two instances of the application of this class of control approaches [60].

Geometric approaches are based on the vehicle's geometric model, velocity, and acceleration [60]. Thus, geometric approaches are popular because of their simplicity. Pure pursuit controller is one of the most popular applications of geometric approaches. Vehicles remain running as planned by "chasing" a moving point in a time step. Kinematic controllers are developed based on the vehicle's kinematic models. This type of control approach can add velocity, acceleration, and even slip into consideration. But forces are not explored. Dynamic approaches are developed regarding forces in vehicles. Thus, in most cases, dynamic approaches are more complex than geometric and kinematic approaches and require additional computational power or data processing time as well. In the Simulation chapter, more contents about the vehicle kinematic and dynamic model are presented.

Optimal control approaches find control laws for a system to achieve an optimal result. The results are usually defined by an objective function. Linear-quadratic-regulator (LQR) control is one of the optimal control methods. Adaptive and intelligent control approaches are expected to be able to change parameters regarding the changing conditions. Intelligent control approaches utilize various artificial intelligence approaches to achieve such goals. Fuzzy logic, machine learning algorithms, and neural networks are commonly chosen for intelligence features.

Model-based control approaches differ from vehicle model-based approaches from those mentioned previously [60]. Model-based control approaches refer to the approach's performance, controlling by assessing and modeling control output. A Model Predictive Controller (MPC) is one of the most known applications of model-based control approaches. With models that reflect the correlations between dependent variables and independent variables, MPCs take action according to the anticipated changes in the dependent variables. Such predictions are commonly taken for a finite time horizon to obtain optimized performance with acceptable processing time.

Learning-Based and End-to-end Autonomous Driving System

Previously mentioned components are commonly deployed in modular ADSs. Due to the modularized structure, such ADSs could be more robust when malfunctions or under-expected performances occur with modules. However, end-to-end ADSs, as an alternative approach to designing ADSs systems and functional components, have been studied in great detail [61] [62] [63]. End-to-end algorithms usually rely on learning-based algorithms. Such algorithms replaced perception, planning, and control components partially or entirely.

Tampu et al. conducted a survey of end-to-end driving approaches [64]. The results show that with full end-to-end driving systems, inputs (e.g., perception data) are directly used to generate control outputs (e.g., steering commands). Each task is completed as an individual learning task, instead of a sequence of tasks processed by different modules. Popular learning methods in end-to-end driving include imitation learning and reinforcement learning. Inputs for end-to-end driving include inputs gathered by aforementioned perception methods, HD maps, and navigation information or input fusions. The outputs could be control commands, waypoints, cost maps, affordances, or multitask deliveries. Figure 3 demonstrates an example of end-to-end driving systems.

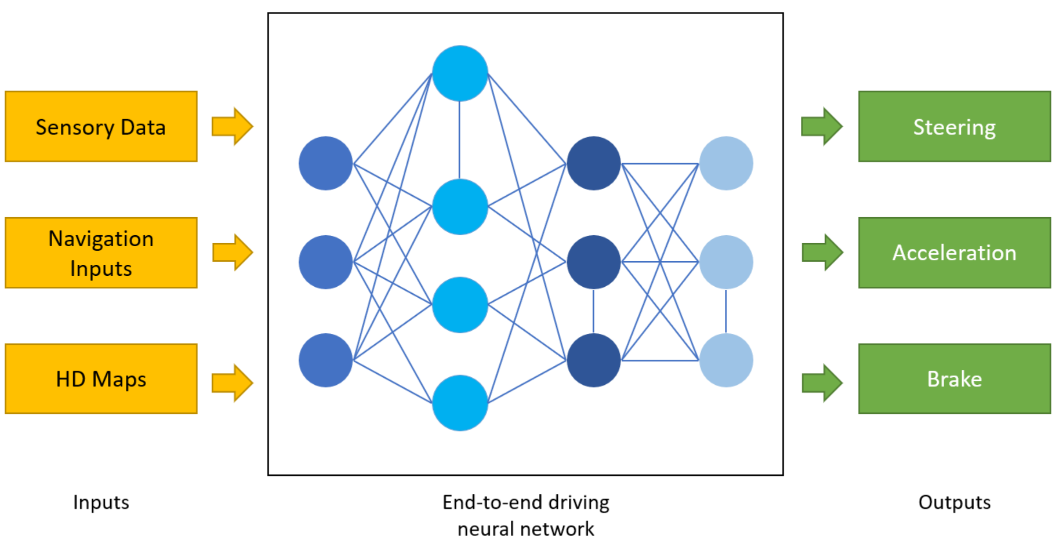

Figure 3. Abstract End-to-end Autonomous Driving System

Inter-vehicle communication

When discussing inter-vehicle communications, the Vehicular Ad Hoc Network (VANET) is one of the inevitable topics. It was developed based on Mobile Ad Hoc Network (MANET) [65] and was introduced at the beginning of the 21st century. VANETs' principle is enabling vehicles to create wireless networks spontaneously without any administrator involvement [66], [67]. The components of a VANET system are Onboard Units (OBUs), Application Units (AUs), and Roadside Units (RSUs). OBUs indicate the devices mounted on a vehicle and used for exchanging information within the vehicle, among vehicles, and with RSUs. AUs are devices that utilize information gathered by OBUs. RSUs are usually fixed in locations. They help the vehicles to communicate with infrastructures. Figure 4 shows an example of information flow among these types of devices. The signal tower is considered an instance of RSUs. The green vehicles are OBUs and IVC-enabled. The applications of VANETs include but are not limited to intersection collision avoidance, approaching

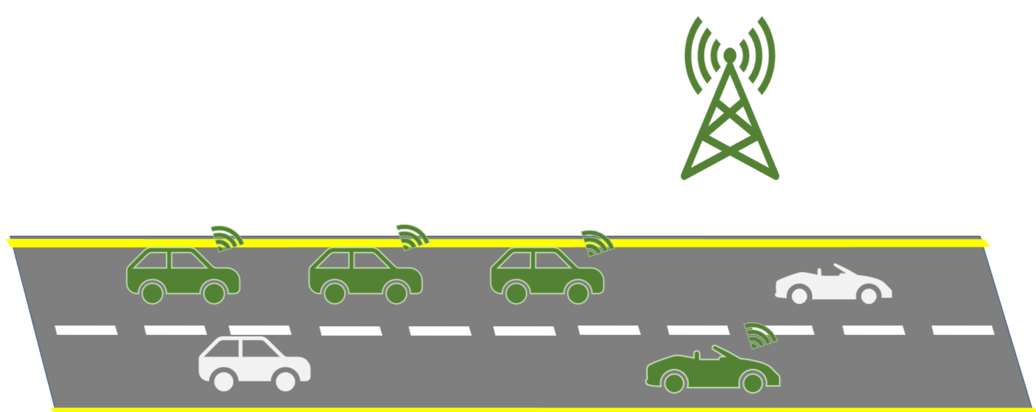

Figure 4. IVC Through RSUs and OBUs

emergency vehicle warning, sign extension, vehicle diagnostics and maintenance, lane change warning, and speed harmony [65], [68]. These applications are expected to improve not only the safety but also the overall performance of the road network.

The two most competitive candidate access technologies for V2X are dedicated short-range communication (DSRC) and cellular V2X (C-V2X). Other candidates include Bluetooth, Satellite Radio and Visible Light Communications (VLC) [69]. Figure 5 shows the illustrative protocol stack of DSRC. As illustrated, DSRC is rooted on IEEE 802.11p standard and extended by 1609 family standards.

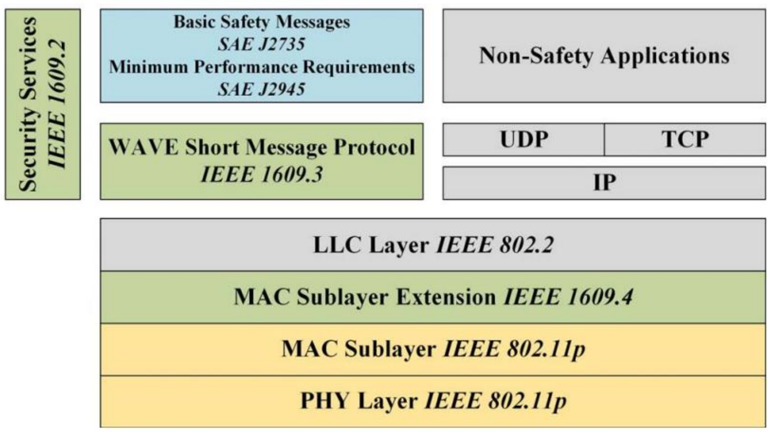

Figure 5. Protocol Stack of DSRC (Retrieved from [154])

SAE J2735 and SAE J2945 regulate the messaging application layer of this method in the US. Defined by SAE J2735, message types include but are not limited to Basic Safety Messages, Emergency Vehicle Alerts, Signal Phase and Timing Messages, Map Data [70]. In EU, message types for C-ITS are Cooperative Awareness Message (CAM) and Decentralized Environmental Notification Message (DENM) [71] [72]. CAMs are defined as the exchanged messages to maintain the awareness of each participant and support the cooperative performance of the road network [72]. DENMs are messages informing road users of detected events [71].

Clustering

Vehicle clustering techniques aim to solve vehicle "grouping" problems regarding the vehicle's geographical information under a VANET environment. A cluster refers to a group of vehicles. In some rules, vehicles in a cluster play three roles – cluster head (CH), cluster member (CM), and cluster gateway (CG) [73] [74]. A cluster head is a vehicle that coordinates with its CM(s) and other CH(s). A cluster gateway is a vehicle linked with other CG(s) that provides connections between its cluster and others. The rest vehicles are called CMs. Figure 6 shows a simplified scene with two vehicle clusters.

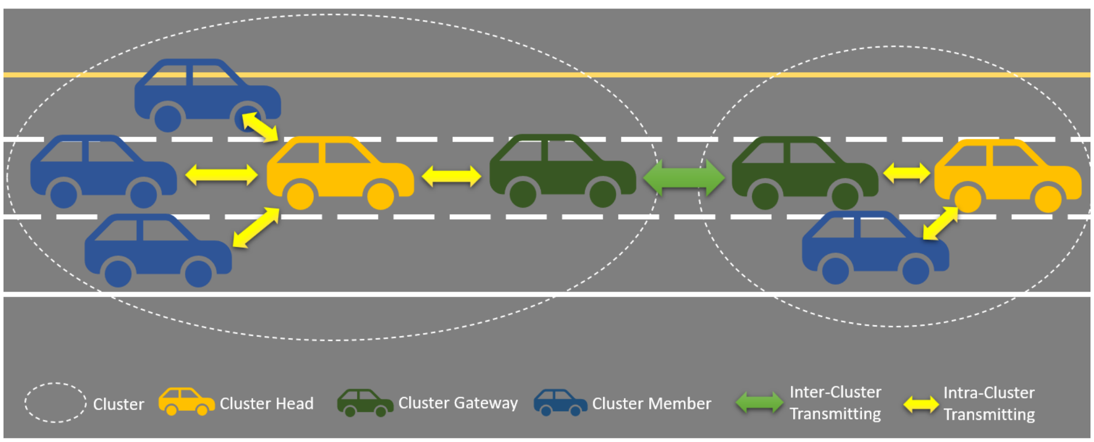

Figure 6. Simplified Vehicle Clusters

The fundamental procedure for clustering under VANET environments is summarized in Cooper et al. study. The procedure includes neighborhood discovery, cluster head selection, affiliation, announcement, and maintenance [75]. Once a vehicle joins a network, it starts broadcasting its existence and gathers similar information from its neighbors. Then a process will be conducted to select a CH within a neighborhood. CH may play centralized information processor and decision-maker roles. Such roles vary depending on clustering algorithms. The vehicles that are assigned to be CHs send announcements to selected vehicles that are selected to be its CMs. Affiliation indicates a vehicle is attempting to build a connection with its optimal CH and becomes a member of that cluster. To maintain a cluster, the CH needs to check the members' status periodically. Reversely, a CM must also check its link with its CH. If a CH finds no connection between it and its member(s), it will return to the neighborhood discovery step. Both CHs and CMs can change their roles on demand. Merging or nesting clusters in another cluster are achievable with some algorithms [75].

Clustering algorithms in VANET are developed to achieve various goals. Some are generalized, and others are developed for specific applications [75]. Generalized clustering algorithms in VANET are mainly aimed at establishing stable and reliable clusters for connected vehicles. Application-dependent clustering algorithms are developed to provide clustering solutions to improve the performance of the applications. Such applications include but are not limited to event management, routing, channel access management, security, traffic safety, etc. [73].

Platooning

Platooning vehicle indicates the technique that enables vehicles to travel along as a single unit. It is also known as convoy driving. Vehicles maintain a constant short spacing within a platoon with coordinated longitudinal and lateral control algorithms. Successful tests on truck platooning in the last two decades provide an experimental perspective to explore the deployment of platooning on other types of vehicles, especially passenger cars. Bhoopalam et al. reviewed publications on truck platooning and summarized the two common objectives of a truck platooning service [76]. They are minimizing the system-wide fuel cost and maximizing the number of trucks platoon. Since the aerodynamic drag can be effectively mitigated by platooning trucks, minimizing fuel consumption is one of the main benefits of platooning trucks. Due to the aerodynamic drag reduction, the size of a truck platoon could extensively benefit from an energy-saving perspective [77]. However, coordinating and platooning trucks with different Origin-Destination OD pairs may lead to extra travel that increases the overall cost. Moreover, platooning trucks may also significantly raise the regional density of large vehicles. That could cause congestion to traffic that increments fuel consumption not only for platooned trucks but also for other users in that area. To manage such issue, more efforts are needed in global planning for platooning and setting the maximum number of trucks in a platoon for some vulnerable areas.

When compared to tuck platooning, there is less research that asserts that platooning passenger cars can potentially lower the fuel cost of traveling [78]. However, autonomous driving without cooperation has led to non-systematic improvement of the performance of the roadway network since such driving is usually based on information that was not collected cooperatively and the decisions made to maximize ego's benefits [79]. Thus, one of the main purposes of platooning passenger cars could be to improve the throughput of the traffic and thereby reduce the time of traveling systematically.

In last couple of decades, tests on platooning vehicles have been conducted as a part of the California Program of Advanced Transit and Highways (PATH). In experiments, platoons are formatted prior to entering the highways. Being guided by magnets embedded in the road, an eight-vehicle platoon with 6.5-meter spacing performed basic maneuvers [80]. Information is transmitted among those platooned vehicles.

Jia et al. conducted a review of the platoon-based vehicular cyber-physical system (VCPS). This review gives a good insight into developing systems for platooning vehicles [81]. Cyber-plane refers to the network and communication plane. The Physical plane indicates the physical constraints or environment that could influence a vehicle's behavior. Under the autonomous driving environment, vehicles observe the environment with their perception system and react to it with planning and control systems. Such actions happen on the physical plane. Transmitting information, both intra-vehicle and inter-vehicle, is related to the cyber plane. Thus, inter-vehicle/platoon communications are critical for the cyber-plane behavior of vehicle platooning as well as the autonomous driving system toward the physical plane.

An inter-platoon CPS is expected to improve the performance of the traffic. On the other hand, an intra-platoon CPS should be able to manage the single vehicle's performance [81]. In this study, intra-platoon CPS is considered as the integration of connected and autonomous vehicle technologies. Such technologies have been discussed in previous sections of this chapter. On a platoon-based inter-vehicle cyber plane, challenges such as coordination and management protocols are popular subjects of study.

Before exploring vehicle platooning technologies, a relevant technology that has partially realized the functions of vehicle platooning is reviewed. This technology is the Cooperative Adaptive Cruise Control (CACC).

CACC (Cooperative Adaptive Cruise Control)

Recently, CACC has been extensively studied, leveraging both autonomous driving and connected driving technologies. CACC is an increment of V2V communications to adaptive cruise control technologies (ACC). It aims to provide automated longitudinal control. An ACC-enabled vehicle can cruise with a preset speed in traffic unless its preceding vehicle runs slower [79]. If a slow preceding vehicle is detected, the ACC system will decelerate the vehicle smoothly and maintains a fixed gap between the vehicle and the preceding one. CACC vehicles are virtually loosely coupled to maintain strings on the road with a fixed gap among them when suitable situations are met. With V2V communication, CACC vehicles can respond more rapidly to speed changes of preceding vehicles than conventional ACC vehicles [82]. Eventually, when increasing shares of CACC-enabled vehicles in traffic, the throughput increases significantly since CACC vehicles can maintain shorter gaps between each other. The main objectives of implemented CACC are improving roadway capacity and mitigating fuel consumption. When utilizing infrastructure-to-vehicle (I2V) communication, CACC has the potential to mitigate traffic bottleneck problems.

CACC system can be separated into two parts – communication and automated control. CACC communication has distinct characteristics. For instance, due to the mobility of vehicles, the routing topology changes dynamically when traffic consists of both conventional human-driven vehicles and CACC vehicles. Thus, communication routing protocols for achieving CACC are critical. Dey et al. reviewed communication routing protocols and categorized them into three classes [83]. They are unicast, multicast, and broadcast protocols. Unicast protocols indicate the strategies that one or multiple copies of an information bundle is/are forwarded to a single recipient. On the contrary, with multicast protocols, the bundle is sent to multiple recipients. Sending multiple copies of one bundle can help with increasing the success rate of delivery. Broadcast protocols cast information bundles toward all users without specific paths. Different from unicast and multicast, with broadcast protocols, bundle casters and recipients spend a shorter time building connections. Since the characteristics vary among those three types of communications protocols, protocols are deployed depending on the situations.

CACC systems are expected to accomplish the following objectives – perceiving and interpreting preceding vehicle's actions, making decision responses to such actions, and taking corresponding reasonable actions. CACC vehicles can effectively achieve those objectives since perception responsibilities are handed over to onboard perception systems and V2V communications. However, based on the definition given by SAE, CACC systems are considered L2 Autonomous Driving Systems since they only perform longitudinal control features.

Collaborative driving system

In 2021, Malik et al. conducted a survey of collaborative autonomous driving [84]. The collaborative driving system (CDS) has gained growing attention in recent years as a research area of ITS. Collaborative driving focuses on improving the system performance of the traffic instead of the performance of a single vehicle or a group of vehicles. Additionally, the layered architecture for CDS is also demonstrated. Figure 7 shows the layered architecture. As demonstrated, the vehicle control layer is responsible for perceiving the vehicle's state information and surrounding events, as well as keeping the vehicle under control for running safely and properly. The vehicle management layer determines each vehicle's behavior in an optimized manner cooperatively. This layer makes decisions based on integrating information from the other two layers and the IVC system. The traffic management layer oversees casting traffic-related information to the vehicle management layer. Such information includes but is not limited to traffic signal schedules, reduced speed limits, roadwork zone information, etc. Vehicle platooning is one of the potential applications within the collaborative driving environment. In the survey, one problem that focused on vehicle platooning has been addressed, which is the platoon leader election. A few questions have been brought up. They are about platooning heterogeneous vehicles, communication degradation in a long platoon, accurate estimations of vehicle characteristics, trustworthiness, and equal profit distribution.

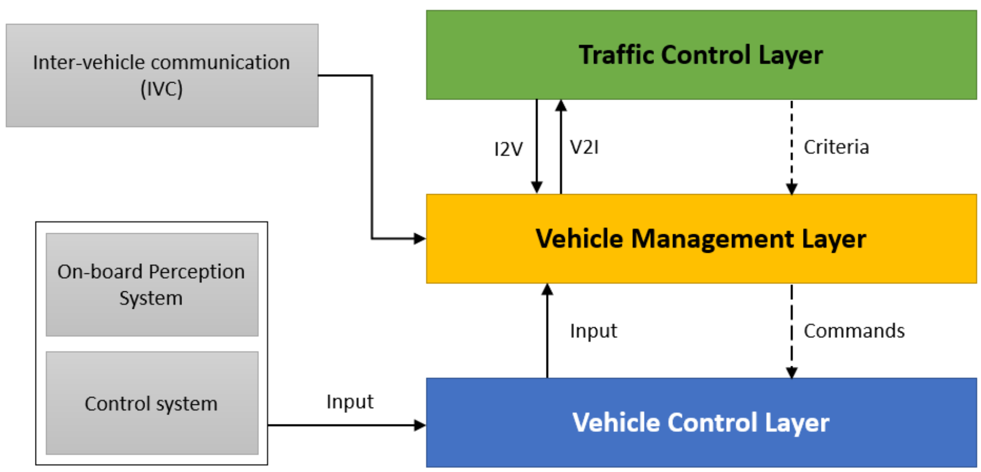

Figure 7. Layered Structure of Collaborative Driving System [84]

Vehicle Platooning System Architecture

After studying existing connected and autonomous vehicle protocols and technologies, an abstract architecture of the vehicle platooning system is depicted. In this study, the fundamental architecture of the vehicle platooning system is demonstrated in Figure 8.

As shown in Figure 8, platooned vehicles are assumed to be level-3 and more advanced autonomous-driving-enabled. Thus, vehicles should be able to operate themselves effectively (with less latency among perception, planning, and control) even without human involvement.

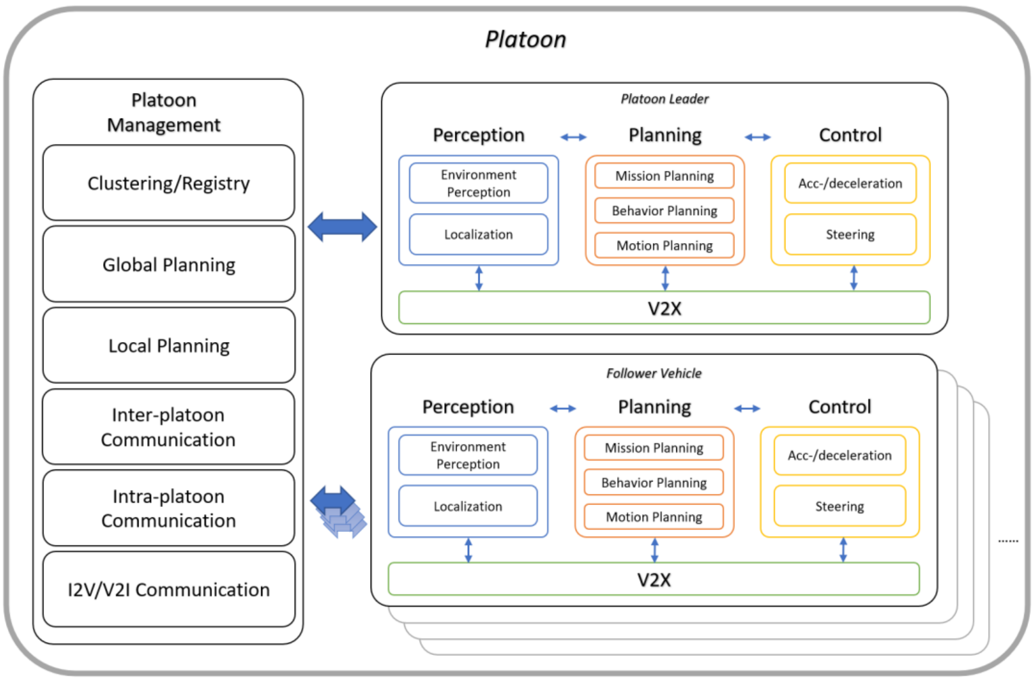

Figure 8. The Functional Structure of CAV platoons

Platooning Vehicle

Effective perception components are necessary for a connected and autonomous vehicle to detect an object or an event that may trigger the CAV to change its state. For instance, when detecting a slowing-moving vehicle ahead, the CAV chooses to decelerate or change its lane. Different from utilizing all perception data solely for ego operation benefits, data is shared via an intra-vehicle communication network with all members of the platoon.

Vehicles may platoon when a section of the planned route is shared by them. This means exact origin and destination are not required for forming a platoon. Thus, each vehicle may have its own mission for transport. Behavior planning modules are responsible for deciding which behavior, such as lane changing, should be selected regarding the current or predicted situation. Even a platoon management system may take over the decision-making from a platoon perspective, but maintaining behavior planning for each vehicle can improve the robustness of performance.

With the vehicle platooning system, vehicle's control systems may receive not only commands from on-board planning modules but also from the platooning management system. In platooning systems, similar to CACC, vehicles are under obligation to inform their control actuation to others in the same platoon through IVC or other communication approaches.

Onboard V2X modules indicate the integration of on-board communication units. In a connected driving environment, V2X modules perform different roles in the majority of the time. In previous sections, some examples of vehicle-to-everything communication have been provided.

Platoon Management System

The platoon management system is in charge of maintaining a platoon in a safe, stable, and optimal manner. The main tasks of the management system include but are not limited to maintaining string stability, determining intervehicle gaps, coordinating platoon formation, path planning, collision avoidance, synchronizing, establishing intra- or inter-platoon communication, etc. In Figure 5, modules of platoon management include clustering/coordination, global planning, local planning, inter-platoon communication, intra-platoon communication, and I2V/V2I communication.

Clustering modules help the system determine which vehicle should be chosen to be included in the communication group. Coordination modules are the modules focusing on guaranteeing cooperation among vehicles within each cluster that was established for platooning. Coordination modules play a critical role in speed synchronization, platoon formation, vehicle joining and leaving, platoon merging and splitting, cooperative lane changing, and other tasks.

Global planning modules of a platoon are similar to the mission planning modules of a vehicle’s onboard system. By integrating information from exterior sources, the platoon may need to update the route dynamically regarding traffic condition changes. For example, when traffic congestion is perceived, and the potential impact is significant, the platoon may reroute to avoid congested areas to mitigate travel time increase. Local planning modules are responsible for developing behavior plans for platooned vehicles. Platooned vehicles include the vehicles that are running in the platoon and vehicles that intend to join the platoon. Local planning also includes inter-vehicle gap maintenance, determining the in-platoon position of joining vehicles, maintaining string stability, and others.

Intra-platoon communications modules are responsible for information exchange among platooned vehicles. Such information may cover the states of the vehicle (velocity, yaw rate, position, etc.), specs (max speed, max deceleration, etc.), missions (origin and destination), and intentions (following, lane changing, leaving, joining).

Inter-platoon communications are not limited to communications among platoons but also include communications between non-platoon cooperative vehicles. States of vehicles and maneuver intentions are essential when exchanging information for inter-vehicle communications. Event awareness information is also important, but such information may not be transmitted via IVC only but also via I2V communications. I2V communications mostly help the platoon management systems with gathering road-network-level information, such as work zone notifications or adjustable speed limits. In some scenarios, platoon management systems may be implemented by utilizing the modules of the platoon leader. On the other hand, V2C communication may rely on cloud computing for implementing centralized management.

Chapter 3

SIMULATION

In this section, modeling and simulation algorithms in both vehicle and traffic scopes are reviewed. As introduced in previous sections, the main goal of this study is to develop a simulation framework to explore the impacts of CAV platoons on traffic. The whole simulation framework and modeling will be computational. The entire simulation has been divided into two parts: microscopic simulation and macroscopic simulation. Different from general terminologies used to categorize traffic simulation software with different scopes, the meaning of these two terms will be explained in the following sections.

The first section of this chapter is a review of studies that focus on CAV platooning algorithms and technologies. Besides the algorithms and technologies, the tools and approaches adopted in those studies for modeling and simulating are presented. The next section explains the vehicle's kinematic and dynamic models, communication topology, and control algorithms related to autonomous and connected techniques. In the third section, traffic simulation tools/platforms are reviewed by going through official documents and related publications.

Research Review

In 2011, Schünemann developed a runtime infrastructure for coupling communication and traffic simulators and conducting sophisticated V2X simulations

[85]. The infrastructure is called V2X Simulation Runtime Infrastructure – abbreviated as VSimRTI. The infrastructure was designed to manage a federation of simulators. The major components of the VSimRTI include a core infrastructure and federates. A Federate consists of a simulator, a federate ambassador, and a VSimRTI ambassador. The simulator could be coupled with VSimRTI, which includes SUMO, VISSIM, JiST/SWANS, OMNeT++, etc. VSimRTI has been upgraded to Eclipse MOSAIC [86].

Segata et al. developed and verified a simulation tool for automated platooning in mixed highway scenarios with the combination of a communication simulation platform and a traffic simulation platform, the OMNeT++ and SUMO [87], in 2012. The proposed platooning protocol in the study was CACC-based. The results showed that with 30% penetration of platoon-enabled vehicles, over 80% of platoons have a size of less than three. The recorded largest size of a platoon during the simulation was 8. In 2014, Segata et al. published the works done based on this simulation tool and named the completed tool PLEXE [88].

In 2013, Zhao and Sun developed a simulation framework that explores the vehicle platooning and car-following behavior under the CV environment [89]. A six-vehicle CACC platoon was simulated on a microscopic traffic simulation platform – VISSIM. Examined platoon maneuvers include forming, adjusting, splitting, dismissing, and joining. They concluded that the lane capacity is positively correlated with the market penetration of CACC-based platoons. On the contrary, platoon size impacts the capacity subtly. The minimum desired headway for ACC vehicles was set to 1.4 seconds as well as 0.5 secs for CACC vehicles.

In [90], Jia et al. presented a disturbance-adaptive (DA) design for VANET-based vehicle platoons. The design aimed to improve the platoon's stability of communication and operation. Within a DA platoon, four roles were played by vehicles. The first is the Leader. Similar to the role in a common platoon system, a Leader makes decisions on platoon-level operations (formatting, splitting, merging, broadcasting the existence of the platoon, etc.). The second role is the Tail. As the name indicates, the last vehicle in a platoon plays this role, and oversees the communication with the leader of next platoon. The third role is called Relay. The relay vehicles perform data-forwarding in a multi-hop VANET environment. The last role is Member, who simply follows the plan decided by the Leader. Based on such theory, authors developed a driving strategy for Leaders of DA-platoon. The driving strategy adopts a sliding mode controller and determines the desired inter-platoon gap by gathering information from the Tail of the preceding platoon. The principles guaranteed the gap to be small enough for not breaking the connectivity with adjacent platoons and large enough for avoiding collisions. The principles also helped with determining the platoon size. Then, simulations were conducted for algorithm validation.

In 2014, a distributed framework was developed for coordinating HDVs for platooning [91]. The core of this framework is a virtual controller system that coordinates HDVs at each vertex (e.g., each major intersection) for platooning. The objective is to maximize the earnings (energy savings minus costs). A case study was conducted based on the German Autobahn network. 0 to 7,000 HDVs were initialized in the network. The results showed, as the number of HDV increase, more fuel saving can be achieved.

Artery, a simulation framework based on OMNeT++, was introduced in 2015 as an extension of Veins [92]. Recently, an extension called Artery-C to adapt C-V2X communication approaches were released. [93] gave an overview of the Artery-C. The pros of Artery-C include 5G selected features, open-source, and capabilities of modeling facilities.

In 2015, Santini et al. presented a consensus-based approach for vehicle platooning under IVC environments [94]. The consensus algorithm was implemented to achieve equal inter-vehicle gaps within a platoon. Validation simulations were conducted. And the results showed that the approach performed better than the classic CACC approach by improving the stability and convergence of the platoons. Later on, in another publication [95], Santini et al. presented their work on validating the consensus-based platooning approach when vehicle communication topology changes due to platoon maneuvers. Simulations were developed and run upon PLEXE. The results proved the reliability of the approach, even while a platoon with heterogeneous vehicles is maneuvering. Tested maneuvers include join-at-tail, leave-at-tail, join-at-middle, and leave-at-middle. Since the communication topology is a typical leader and predecessor following topology, vehicles in a platoon communicate with the platoon leader and the two adjacent vehicles (the one ahead and the one after).

In 2015, Li et al. reviewed relevant studies and presented a four-component framework of the vehicle platoon system [96]. The four components of the framework are 1) Node dynamics, 2) information flow topology, 3) distributed controller, and 4) formation geometry. Node dynamics describes the behavior of every platooned vehicle and all others involved. Information flow topology means how the vehicles exchange information with each other. Feedback controllers handle feedback control with neighboring information. Formation geometry indicates the desired inter-vehicle gap within a platoon.

Deng published a simulation framework for modeling and analyzing heavy-duty vehicle (HDV) platoons [97]. The simulation framework was built upon the commercial simulation platform – VISSIM. Within the framework, a fuel consumption model was embedded to estimate the influences of HDV platoon on fuel saving. Input data for fuel consumption estimation is obtained by recording the state of each vehicle. The truck's state data includes motion state (lane, longitudinal position, speed, acc-/deceleration), platooning information (in-platoon position, inter-vehicle distance), and temporal data (time instance). Three cases were studied to figure out the influences of HDV platoon and platoon formation maneuvers. The results showed that, with HDV platoons, even the aggregate highway velocity dropped, but the traffic flow rate increased on a two-lane highway section. Otherwise, under medium/high traffic scenarios, the interest in reducing platoon formation time conflicted with the arrival time of the HDVs.

Ribeiro et al. proposed a platooning management protocol [98]. In the study, the protocol covers the procedures of platoon's creating, merging, and dissolving operations as well as vehicle's joining and leaving maneuvers. Simulation and testing were conducted on a combination of SUMO, V2X Simulation Runtime Infrastructure (VSimRTI), and ns-3. Two-truck platoons were introduced into the simulation network. The results showed that, with the protocol,

a) vehicle joining the platoon from rear has been completed in 69 seconds(s) on average, or in 114.2 seconds when joining from the adjacent lane.

b) one average, a vehicle leaving the platoon took 23.3 s to change to the adjacent lane, or 1 s if it's the last vehicle in the platoon and leaves by increasing gap to its preceding vehicle;

c) adjusting the gap took 12.6 s for a joining maneuver or 17.4 s for a leaving maneuver on average.

d) Merging two platoons was similar to the joining maneuver, but with 62.7 s adjusting on average.

e) dissolving maneuver took 3.8 s on average.

Additionally, message latency was limited to 100 milliseconds (ms), which meets the required maximum delay. The capacity of a lane was proven to be better with platoons.

In Bang and Ahn's 2017 publication, a platooning strategy for CAVs based on the spring-mass-damper system is described [99]. With the system, the longitudinal platoon control of CAVs by controlling the spring constant and damping coefficient is presented. The maximum acceleration/deceleration, mass, and length were used to determine the controlling parameters. Moreover, different relations between the spring constant and traffic flow were considered when developing simulation scenarios. The results proved that, with the critical damping coefficient, the maximum efficiency of completing platoon formation could be achieved when the maximum relation between flow and spring constant was selected.

In 2017, Jain et al. published their work on developing a prediction-based framework for vehicle platooning [100]. An MPC-based control algorithm, which solely relied on V2V communication, was developed and implemented for vehicle platoon control. Both simulations and experiments were conducted to test the algorithm. Simulations were performed on the Dominion framework developed by German Aerospace Center. A two-vehicle platoon with both 5G-V2X and 802.11p communications was used for the experiments. Both simulation and experiment results showed the excellent potential of controlling a vehicle platoon with pure V2V communication. The authors also concluded that 5G performed better than 802.11p in providing V2V/V2I communication because of its larger data rate and communication range.

In 2017, Liu et al. published their research on the platoon system engineering process that considers safety and cyber security issues [101]. The engineering process has four steps - 1) defining the safety goal, 2) defining the attack model, 3) deriving the security goal, and 4) deriving functional security requirements. Due to the tight coupling relation among vehicles in a platoon, the impacts of a cyber-attack against the platoon system could lead to decreasing stability, platoon dissolution, and even collision. Functional security requirements of a platoon system included but were not limited to detecting false messages, ensuring the timeliness of messages and responses, and keeping messages intact from attackers. The general approaches to developing a platooning system with the proper capability to maintain safety and security include optimizing the gap between vehicles with security consideration, enabling cyber-attack detection, and deploying fail-safe mechanisms to eliminate harm when attacks are encountered. In the paper, a proactive platooning approach was presented. The approach calculated the optimal acceleration difference threshold based on the desired acceleration under CACC and pure ACC situations. Moreover, the desired gap was determined based on the desired acceleration. Simulations with PLEXE and MATLAB validated the algorithm.

In 2018, Ramezani et al. developed a simulation model for exploring the influences of CACC-based truck platooning operations on traffic [102]. Aimsun Next Micro-SDK was used for developing the simulation platform. The aspects of trucks, including desired acceleration speed, were computed explicitly and implemented in the simulation. A case study was conducted based on a 15-mile urban section of the I-710 Northbound – an Interstate highway in Southern California. The results showed that truck CACC platoon could increase the speeds of cars by reducing congestion when penetration rates reach a high level (over 80%). However, in on-ramp areas, since truck platoons used the rightmost rule, the merging traffic had to wait longer when penetration rates were low. Generally, when penetration rates of CACC trucks reached 100%, the benefits included easing congestion propagation and increasing the average speed of traffic at uncongested areas. Otherwise, similar benefits were not found with CACC car's platoons.

In a publication by Ibrahim et al., a co-simulation framework developed by the authors for vehicle platooning was presented [103]. The framework consists of ns-3, SUMO, and MATLAB. Ns-3 simulated the packet broadcast of vehicles. SUMO simulated the traffic. Control algorithms were developed in MATLAB and replaced the algorithms that were given in SUMO. MATLAB also performed as the interface between SUMO and ns-3. The developed algorithms focused on longitudinal acceleration control. Model predictive control and state-feedback control algorithms were implemented for the upper- and lower-layer control. Upper-layer controller determined acceleration or deceleration regarding the gap to the preceding vehicle. And lower-layer controller worked to eliminate errors in acceleration/deceleration. Tests were conducted for framework validation. Simulations based on realistic highway scenarios were conducted as well. The results showed that both the severity and frequency of platoon vehicle's speed fluctuations increased when packet losses grew.

Vieira et al. developed a realistic simulation framework for vehicular platooning based on Robotic Operating System (ROS) framework [104] and published a paper in 2019. ROS is a popular framework for designing robotics applications. An integration of Gazebo (ROS robotic simulator) and OMNeT++ was presented. ROS publish/subscribe mechanisms played a critical role in data delivery and simulator synchronization. In a later Vieira et al. publication, after more work had been done, this realistic simulation framework was named COPADRIVe [105].

In 2019, Gerrits et al. developed a study exploring the influences of opportunistic truck platooning matchmaking algorithms [106]. One is First-Viable Match (FVM), and the other is Best-Match (BM). The FVM takes waiting time into account as the cost. Once a match can lead to a positive earning (subtracting savings over costs), the match is selected. The BM selects the match with the highest earnings within a searching area. Properties of the truck (hourly wage, urgency, brand, destination, matching locations) were added as factors into the simulation model. The results showed that BM performed better than FVM on saving. The wage savings are significant when platooning trucks.

Sethuraman et al. developed a simulation to evaluate the impacts of bus platoons on traffic [107]. The simulation scenario was developed based on a 16-kilometer (KM) section of an urban roadway in Singapore with two major signalized intersections. The simulation was run on the VISSIM platform. Both the quality of services of the bus and the performance of the traffic were assessed. Numerical analysis showed that, generally, the simulation results showed that bus platooning increased the operational speed of buses and other cars and, as a result, the overall delay was reduced for both types of vehicles. Moreover, output data indicated similarity with platooning trucks since platooning buses reduced the aerodynamic drag and fuel consumption. Correlation between the number of buses in the platoon showed a positive coefficient with energy savings.

In 2019, Hoef et al. published their predictive framework for dynamic HDV platoon coordination [108]. The presented framework aimed to coordinate the in-route formation of platoons. The core of the framework is a platoon coordinator. A layered control system architecture for coordinated platooning was also presented. The layers are the service layer, strategic layer, tactical layer, and operational layer. On the first layer, transport tasks are managed. On the second layer, strategies, such as coordinating platoons, are performed. For the remaining two layers, platoon management systems and vehicle control systems are deployed.

In 2019, Hyun et al. published a paper that overviews a statistical verification framework for a platooning system of systems (SoS) [109]. The framework, called StarPlateS, consists of three modules – scenario generation module, simulation module, and verification module. The first module performed platoon configuration generation and scenario generation. The simulation module was handed over to a SUMO/OMNeT++ integrated extension – Vehicular Network Open Simulator (VENTOS). The verification module checked the achievement rates of goals with the statistical model checking (SMC) algorithms. The two checked properties of the SoS were the throughput within a specific time horizon and the rejection rate of operation.

Since 2006, Sommer et al. have started working on a model library for OMNeT++. It's named Veins. Recently, a publication in 2019 provided an overview of the developments of the library [110]. Now, Veins support the simulations not only relating to the IEEE 802.11p family but also LTE and Visible Light Communication (VLC) are covered by the library as well. The Veins does not manage road vehicle simulations. However, by bidirectional coupling with SUMO through Traffic Control Interface (TraCI), users may customize vehicular mobility models on demand. Models on the communication layer are created by Veins in the OMNeT++ simulator to represent vehicles. Then, by establishing mapping with the mobility models in SUMO, the cyber-physical system of a connected vehicle environment is simulated.

Quadri et al. published one of their works on the MEC-based vehicle platoon control framework for vehicle platooning in 2020 [111]. MEC is the abbreviation of multi-access edge communication. Being different from some distributed controllers, an MEC-based controller offers a centralized approach to platoon control. Simulations relying on SUMO and Python-based applications were conducted for two types of scenarios – sinusoidal and real-trace vehicle movement patterns. According to the results, with an MEC-based controller, inter-vehicular can be shortened to 5 meters or less. However, since the round-trip time (RTT) of cloud computing could hardly be achieved below 150 ms on average, deploying the controller onto the cloud was not suitable for such centralized control approach.

In 2021, Hidayatullah and Juang published their study on the centralized and distributed control framework under homogeneous and heterogeneous platoons [112]. In the paper, they used PreScan and MATLAB/Simulink to establish simulations to investigate the string stability of both centralized and distributed control frameworks comprehensively. Features that include vehicle dynamics, sensing, and V2X communication are added to the simulations. The performance index integral square error (ISE) was used to evaluate two frameworks. The results showed that, with 0.05 latency and 30% packet loss probability, a distributed framework achieved slightly less ISE of mean spacing error than a centralized framework.

In 2021, Miekautsch et al. published their study on a situation-dependent communication topology for platooning heterogeneous vehicles. In that publication, platoon systems were studied via the four-component framework [113]. A heterogeneous vehicle platoon indicates that a platoon was formatted by vehicles with various configurations (power-train time constant, max acc-/declaration, max speed). Two situations were simulated – emergency braking and count-in. For the emergency braking situations, the author presented a flexible reversed Leader-predecessor-follower (LPF) communication topology. With such topology, the vehicle's configuration and the order within its platoon were considered. By deploying this topology, a platooned vehicle with a higher maximum deceleration rate can brake harder compared to where the traditional LPF topology was deployed, and the vehicle was limited to use the lowest deceleration rate of all vehicles within its platoon.

In 2021, a publication by Xu et al. revealed an opensource simulation tool for cooperative driving automation CDA [114]. It's called OpenCDA. The key features of OpenCDA were summarized as IFMBC – integration, full-stack platform, modularity,

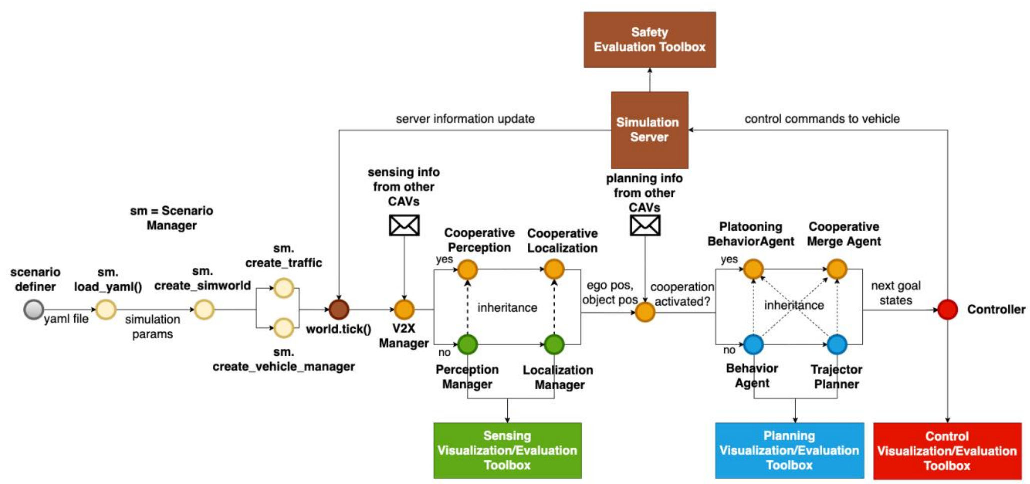

Figure 9. OpenCDA Logic Flow (Retrieved from [114], Oct 26, 2022)

benchmark, connectivity, and cooperation. OpenCDA selected CARLA and SUMO as the simulator for traffic. The hierarchical architecture of the tool was demonstrated. The top layer was the "PlatoonManager" class. Then is the "VehicleManager" class. The bottom layer includes "PerceptionManager", "LocalizationManager", "BehaviorManager", "ControlManager", and "V2XManager". The following figure (Figure 9) demonstrates the logic flow of the simulation process in OpenCDA. Simulations based on various scenarios were reviewed through the safety, stability, and efficiency perspectives.

In 2022, an integration of ROS2, OMNeT++, and MATLAB was presented by Teper et al. as a simulation framework for cooperative driving [115]. ROS2 is an improved version of ROS, which provides a powerful platform for developing robot applications. In this integration, the Gazebo plugins of ROS2 were used to simulate the robot objects (vehicles). Controllers are coded in MATLAB/Simulink and alter the control modules in ROS2 through a toolbox embedded in a recent release of MATLAB. OMNet++/Veins, as in other studies, managed the communication events that may change the vehicle's state.

In a publication by Liu et al., a five-layered vehicle platoon control system was adopted. The top of this hierarchical architecture is the cloud-based decision layer [116]. In this layer, all information was gathered by OBUs and RSUs involved in a system and utilized to make optimized decisions for the system (traffic and network). The second layer was the communication layer. All the information exchange happened in this layer, including messages for the upper layer, V2V message, and environment perception information. In the middle layer, the management layer, the tasks of choosing maneuvers and assigning roles to vehicles were accomplished. The fourth layer was the control layer. In this level, behaviors were broken down into trajectory and control commands, and then commands were passed down to the next layer – the Physical layer. The physical layer indicated the execution of control commands. Based on that, a two-dimensional framework for the management layer was developed. The two dimensions were the roles of vehicle and platoon maneuvers. Under this framework, each vehicle would take corresponding strategies when one maneuver was selected.

Simulation Software and Tools

In this section, a review of selected simulation tool and software were conducted. Such tools performed one or more of the following tasks – traffic simulation, network simulation, application development, and simulation. The information is obtained from review/survey studies and the official website of those tools.

MATLAB/Simulink

MATLAB is one of the most popular programming and numeric computing platforms for modeling. It has been widely used for vehicle modeling, controller development, data analyzing, and visualization not just in the CAV and vehicle platooning areas but in various fields. Simulink provides a block diagram environment for designing and developing a system with models. It's commonly used for control algorithms designing, simulating sensing approaches, and vehicle dynamic modeling in the CAV domain. The common usage of MATLAB/Simulink is to develop vehicle-level models and simulations. Such models are usually modules and applications related to the three functional components of autonomous systems – perception, planning, and control. Existing packages in Simulink were utilized in this study include the vehicle body dynamic block and other basic blocks.

GIS

Since the end of the 20th century, the application of Geographic Information System (GIS) in traffic management has been discussed [117] [118]. As presented on the U.S. Department of Transportation website, the GIS provides a tool to display, analyze and manipulate traffic data with inherent spatial information [119]. GIS is commonly used to present information that is highly bonded with spatial information. The strength of GIS, from a transportation perspective, is the binding between numerical data and graphical lines and shapes that represent infrastructures or objects in which transportation researchers are interested. Such a strength helps with not only data presentation but also information gathering, spatial-correlated analyses, traffic management, and planning. In this study, a GIS public database managed by the Delaware.gov is used for retrieving essential traffic data.

SUMO

Simulation of Urban Mobility is a microscopic and continuous multi-modal traffic simulation software with the capability of simulating extensive networks. Due to its open-source characteristic, recently, a vast amount of library and package was developed by researchers to explore the algorithm, application, and influences of CAV and vehicle platooning. Modules, like Traffic Control Interface (TraCI), help users with developing applications or functions with different languages (C++ and python) and platforms (MATLAB). After reviewing publications related to CAV platoon simulation, it was found that SUMO was the most popular traffic simulation tool for this topic.

OMNeT++

Objective Modular Network Testbed in C++ (OMNeT++) is commonly used for developing vehicular network simulators. It has features such as extensible, modular, and component-based. The integrations of OMNeT++-based communication simulators and some traffic simulators are popular when conducting simulations to explore IVC, V2V, and V2X communication issues. OMNeT++ is the communication simulation in this study.

Microscopic Simulation

Microscopic simulation consists of two major parts: modeling vehicles and simulating the communication environment. Modeling vehicles includes developing or applying the following models: a) individual vehicle models, b) V2V and V2I models, c) CAV platoon's stability model, and d) CAV platoon's operation models. Communication environment simulation will be based on existing studies' results. Communication factors such as topologies, latency, and reliability will be reflected by variables and functions.

Vehicle Kinematic Model

As shown in Figure 10, a bicycle model and relevant notations are presented.

Based on this scenario, the basic kinematic model of a moving rigid-body bicycle-like object uses the following differential equations.

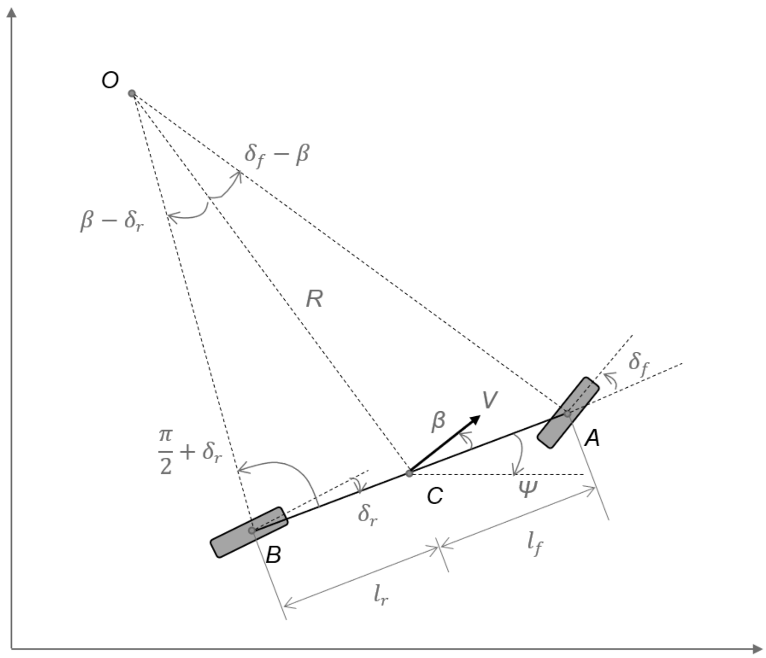

Figure 10. Bicycle Kinematic Model

$$\dot{X} = Vcos(\psi + \beta) \tag{1}$$

$$\dot{Y} = Vsin(\psi + \beta) \tag{2}$$

$$\dot{\psi} = \frac{Vcos(\beta)}{l_f + l_r}\left(\tan\left(\delta_f\right) - \tan(\delta_r)\right) \tag{3}$$

When considering angle of the rare wheel to be subtle, Eq. (3) can be adjusted to Eq. (4). Where δ stands for the steering angle.

$$\dot{\psi} = \frac{V}{L}\tan(\delta) \tag{4}$$

Vehicle Dynamic Model and Control Laws

For the individual vehicle's dynamic models, many models express the vehicle dynamic through different perspectives. With higher vehicle speed, assuming that each wheel is in the same direction as the velocity is no longer valid from the kinematic perspective. Thus, dynamic-based vehicle body models are developed to take a sideslip angle into account.

The lateral dynamic model in terms of error with respect to the chosen path is a proper model for this study [120], [121], [122], [123], [124], [125]. Since navigation correction based on error is one of the core functions in most autonomous driving systems [126], [127], [128], expressing the vehicle's dynamic in terms of error can help with smoothly converting variables between the navigation functions and the vehicle's dynamic models. The following table shows the original equations for which they serve as the foundation of the dynamic model that will be used in this study.

Table 3. Basic dynamic model [120], [121], [122], [123], [124], [125]

Symbol	Nomenclature	Equation
s	State space vector	$s = [e_1\ \dot{e}_1\ e_2\ \dot{e}_2]^T$
		$\dot{s} = Ax + B_1\delta + B_2\dot{\psi}_{des} + B_3 sin\ (\phi)$
e_1	Lateral position error with respect to road	$\ddot{e}_1 = \ddot{y} + V_x(\dot{\psi} - \dot{\psi}_{des})$
e_2	Yaw angle error with respect to path	$e_2 = (\psi - \psi_{des})$
δ	Front wheel steering angle	
$\dot{\psi}_{des}$	Desired yaw rate determined from road radius R	$\dot{\psi}_{des} = \frac{V_x}{R}$
ϕ	Bank angle with sign convention	

In the second row of Table 1, A, B_1, B_2, B_3 are defined as:

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & -\frac{2C_{af} + 2C_{ar}}{mV_x} & \frac{2C_{af} + 2C_{ar}}{m} & -\frac{2C_{af}l_f + 2C_{ar}l_r}{mV_x} \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{2C_{af}l_f - 2C_{ar}l_r}{I_zV_x} & \frac{2C_{af}l_f - 2C_{ar}l_r}{I_z} & -\frac{2C_{af}l_f^2 + 2C_{ar}l_r^2}{I_zV_x} \end{bmatrix} \quad (5)$$

$$B_1 = \begin{bmatrix} 0 & \frac{2C_{af}}{m} & 0 & \frac{2C_{af}l_f}{I_z} \end{bmatrix}^T \quad (6)$$

$$B_2 = \begin{bmatrix} 0 & -\frac{2C_{af}l_f - 2C_{ar}l_r}{mV_x} - V_x & 0 & -\frac{2C_{af}l_f^2 + C_{ar}l_r^2}{I_zV_x} \end{bmatrix}^T \quad (7)$$

$$B_3 = [\ 0 \quad g \quad 0 \quad 0\ ]^T \quad (8)$$

C_{af} and C_{ar} are the cornering stiffness of the front and rear tires. l_f and l_r indicate the distance from the center of gravity to the centers of front and rear tires. m is vehicle mass. V_x represents vehicle longitudinal velocity. I_z is the yaw moment of inertia. g is the gravitational constant.

For vehicle platoon longitudinal dynamic modeling, there is substantial amount of research works that have been conducted for autonomous driving and vehicle platooning in the last twenty years [129], [130], [131].

However, encapsulated vehicle dynamic models including longitudinal and lateral motions are provided in popular vehicle modeling tools. Such integrated vehicle dynamic models vary by the degrees of freedom (DOF). The most basic but also widely used model is based on two DOF bicycle models. The 2-DOF vehicle dynamic model is also known as a linear bicycle model, commonly used to calculate vehicle's yaw rate and lateral acceleration. In 2-DOF model, the lateral tire-road forces are considered to be a function of sideslip angle. 3-DOF models add longitudinal velocity into account. Moreover, when elements are added, such as steering dynamics and suspension movement in the vertical direction, the vehicle dynamic models are presented with more DOFs.

However, conducting a comprehensive review of vehicle dynamic models is not a major objective of this study. With MATLAB/Simulink, multiple preset vehicle body models based on vehicle dynamic models are provided. In this study, the 3-DOF vehicle body model is deployed for simulating the vehicle's lateral, longitudinal and yaw motions. The model accounts for vehicle mass, aerodynamic drag, and weight distribution due to motions. The following equations are used to calculate the rigid body dynamics.

$$\ddot{y} = -\dot{x}r + \frac{F_{yf} + F_{yr} + F_{yext}}{m} \tag{9}$$

$$\dot{r} = \frac{aF_{yf} - bF_{yr} + M_{zext}}{I_{zz}} \tag{10}$$

$$r = \dot{\psi} \tag{11}$$

$\ddot{x}$ and $\ddot{y}$ stand for the accelerations along x- and y-axis. r and $\dot{r}$ demonstrate the vehicle angular velocity and changing rate. F_{xf}, F_{xr}, F_{yf}, F_{yr} are the forces that are applied on front and rear wheels along the vehicle's x- and y-axis. F_{xext} and F_{yext} are external forces along vehicle's x- and y-axis. M_{zext} is the moment on vehicle's z-axis. I_{zz} is the vehicle body moment of inertia about the vehicle's z-axis. Thus, this model is a dual-track vehicle dynamic model. m refers to the mass of the vehicle.

The basic upper-level longitudinal control law for cruise control, or adaptive cruise control under free driving conditions (no object is detected ahead within a range) is proportional-integral (PI) control using error in speed as the feedback signal [120]. The following equation shows the control function.

$$\ddot{x}_{des}(t) = -k_p(V_x - V_{ref}) - k_I \int_0^t (V_x - V_{ref})dt \tag{12}$$

Where k_p and k_I stand for gains of proportional difference and integral of the difference. V_x and V_{ref} are current longitudinal speed and reference speed. t indicates the control horizon.

With adaptive cruise control (ACC), the vehicle needs to guarantee a safe gap with vehicle driving ahead of it. In this study, constant time-gap spacing policy is adopted. The desired inter-vehicle desired spacing distance and spacing error can be formulated as [132]:

$$L_{des} = l_{i-1} - T_s \dot{x}_i \tag{13}$$

$$\delta_i = \varepsilon_i - T_s \dot{x}_i \tag{14}$$

Where ε_i stands for inter-vehicle spacing

$$\varepsilon_i = x_i - x_{i-1} + l_{i-1} \tag{15}$$

x_i and x_{i-1} indicate the longitudinal location of the i th and $i-1$ th vehicle. l_{i-1} is the length of the preceding vehicle. Derived from 12 and 13, a control law for determining desired longitudinal acceleration of the i th vehicle with adaptive cruise control under conditions where crucial objects were detected ahead can be defined as:

$$\ddot{x}_{des_i} = -\frac{1}{T_s}(\dot{\varepsilon}_i - \lambda\delta_i) \tag{16}$$

$\dot{\varepsilon}_i$ indicates the relative speed between the i th and $i-1$ th vehicle. δ_i refers to space error of the of the ith vehicle. λ is a gain parameter needs to be tuned. More information is given in next chapter.

String stability theory has been proven to be a remarkable tool for stability analysis of vehicle platoon. It is based on preventing spacing error propagation from one vehicle to another in the platoon [133]. The following equations were used to define a string stable system [120].

$$\left\|\widehat{H}(s)\right\|_{\infty} \leq 1 \tag{17}$$

Where $\widehat{H}(s)$ is the transfer function relating the spacing errors of consecutive vehicles.

$$\widehat{H}(s) = \frac{\varepsilon_i}{\varepsilon_{i-1}} \tag{18}$$

Under connected environment, ACC enhanced by wireless communication, can be developed to ensure the string stability of a platoon with the constant spacing policy. One algorithm, presented in Rajamani's book [120], is sliding surface control approach. The control law for the i th vehicle in a CACC-based platoon can be formulated as:

$$\begin{aligned} \ddot{x}_{i_{des}} = (1 - C_1)\, \ddot{x}_{i-1} + C_1\, \ddot{x}_l - \left(2\xi - C_1\left(\xi + \sqrt{\xi^2 - 1}\right)\right) \omega_n\, \dot{\varepsilon}_i \\ - \left(\xi + \sqrt{\xi^2 - 1}\right) \omega_n C_1 (V_i - V_l) - \omega_n^2\, \varepsilon_i \end{aligned} \tag{19}$$

Where C_1 is a gain ranging from 0 to 1. ξ indicates the damping ratio when considering a platoon's longitudinal alignments as a spring system. And ω_n means the bandwidth of the controller. ε_i is defined in equation 14. V_i and V_l represent the longitudinal velocity of the i th vehicle and the leader vehicle.

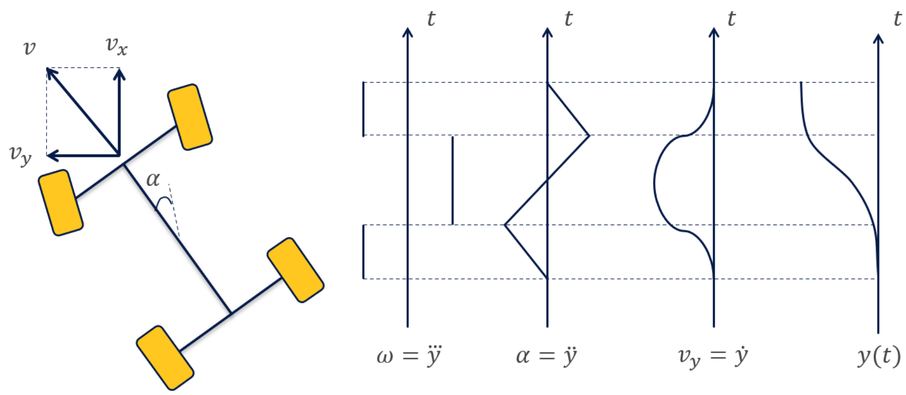

Figure 11. Cubic Polynomial Trajectory Generation Curves

Trajectory Generation

In this study, platoon merging and splitting involving multiple lanes have two essential parts. The first is the preparation period. During this period, the merging vehicle/platoon needs to synchronize its speed with the target platoon. Then, the target platoon needs to adjust the headway between some vehicles to create space for the merger. During the second period, the merging takes place. The core task of this part is to model the lane-changing paths. The operation could be assumed to be complete by giving the vehicle a constant steering rate. By assuming the steering rate is linearly related to the vehicle yaw rate, the curvature of the path will be a cubic polynomial function. Figure 11 shows how the formula is achieved. The corresponding functions representing the relationship of position, velocity, and acceleration of variables versus time are shown below.

$$s(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 \quad (20)$$

$$\dot{s}(t) = a_1 + 2a_2 t + 3a_3 t^2 \quad (21)$$

$$\ddot{s}(t) = 2a_2 + 6a_3 t \quad (22)$$

However, the trajectory planning may be modified for different optimization purposes, such as minimizing execution time, minimizing energy consumption, and even minimizing the maximum actuation forces and moments [134]. Minimum Jerk trajectory planning is widely used for generating trajectories for robot control. Jerk indicates the time derivative of acceleration. It was proved to be critical for generating smoothness trajectory [135]. Minimum Jerk trajectory planning is optimized for minimizing the force for control actuation. The following equations represent the relationship of position, velocity, and acceleration versus time [136].

$$s(t) = a_0 + a_1 t + a_2 t^2 + a_3 t^3 + a_4 t^4 + a_5 t^5 \quad (23)$$

$$\dot{s}(t) = a_1 + 2a_2 t + 3a_3 t^2 + 4a_4 t^3 + 5a_5 t^4 \quad (24)$$

$$\ddot{s}(t) = 2a_2 + 6a_3 t + 12a_4 t^2 + 20a_5 t^3 \quad (25)$$

Where a_0 to a_5 are parameters need to determine by inserting the initial and final state information of the trajectory. Additionally, the following equation represents the correlation between control input u and states S.

$$\dot{S} = A_{Tr} S + B_{Tr} u \quad (26)$$

Where A_{Tr} and B_{Tr} refer to gain matrices.

$$A_{Tr} = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \quad and \quad B_{Tr} = \begin{bmatrix} 0 \\ 0 \\ 1 \end{bmatrix} \tag{27}$$

And by inserting initial and final states $S(0)$ and $S(T)$, parameters a_0 to a_5 can be determined.

$$S(0) = \begin{bmatrix} s(0) \\ \dot{s}(0) \\ \ddot{s}(0) \end{bmatrix} = \begin{bmatrix} x(0) \\ 0 \\ 0 \end{bmatrix} \quad and \quad S(T) = \begin{bmatrix} s(T) \\ 0 \\ 0 \end{bmatrix} \tag{28}$$

Platoon Maneuver

Modeling common platoon operations are also critical for completing the microscopic simulation in this study. Besides cruising in one lane, operations such as merging, splitting, and lane changing could challenge the whole traffic more extensively and intensively. As a result, they are more complex. Merging means one or more

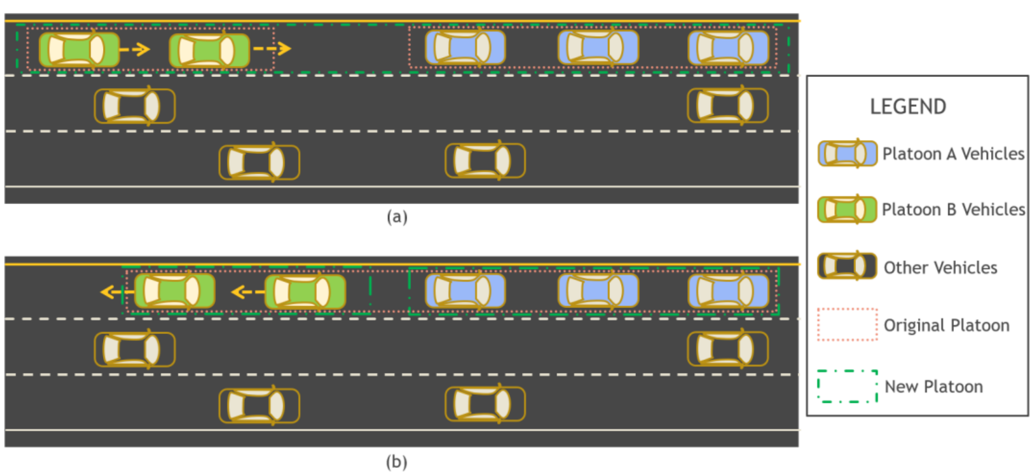

Figure 12. Platoon (a) Merging and (b) Splitting Happen in the Same Lane (Direction of Travel -->)

vehicles joining an existing platoon (one or more vehicles) to form a new one. Splitting indicates a platoon with at least two vehicles splitting into two platoons (one or more vehicles). Both merging and splitting could involve one or more lanes. The other operation, which is lane-changing, means the whole platoon leaves the current lane and changes to another one.

Merging and splitting could happen in one lane. For instance, when a platoon intends to merge into a platoon running right in front of it in the same lane, and no other vehicle is between them, that platoon could accelerate and chase the other platoon. Splitting a platoon in the same lane could happen, for example, when a platoon is cut by a non-connected vehicle. One platoon has to operate temporarily as two separate platoons. Figure 12.a and Figure 12.b gives examples of platoon merging and splitting in the same lane.

When merging and splitting involves multiple lanes, such an operation may impact more vehicles. Figures 13.a and 13.b show platoon merging and splitting that involves multiple lanes. If a vehicle or a platoon is merging from a heterogenous lane,

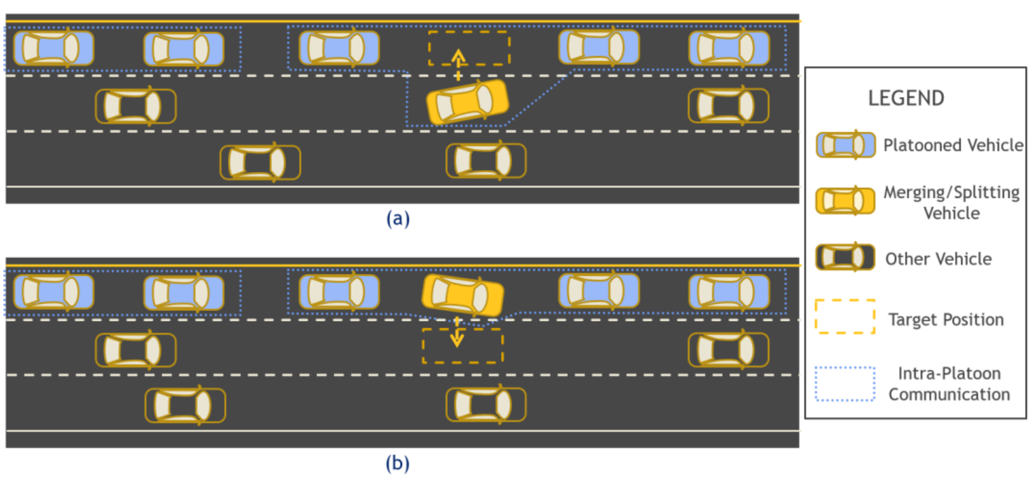

Figure 13. Platoon (a) Merging and (b) Splitting involve Multiple Lanes (Direction of Travel -->)

vehicles behind the merger and the target platoon could both be impacted. On the other hand, the merging and splitting platoons may encounter more challenges under such situation. For example, when a vehicle just leaves its platoon and enters another lane, it will be exposed to an impact from both vehicles driving ahead on its original and target lanes. Figure 14 gives an example of why the situation will be complex when merging happens on multiple lanes. When merging is taking place, if either the vehicle in front of the platoon or the merging vehicle (black and red vehicles) brake, vehicles in the platoon, especially the two followers, would be behind the merging vehicle when the

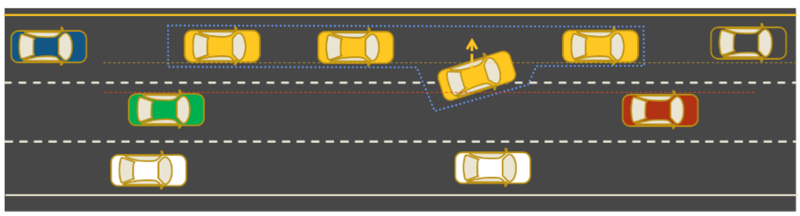

Figure 14. Merging from Other Lane Leads to Complex Situations (Direction of Travel -->)

new platoon is complete. Moreover, if the platoon detects an obstacle and decides to decelerate drastically, both the vehicle running behind the merging vehicle and the platoon may have to take the same maneuver.

Thus, modeling merging and splitting in the same lane is easier than modeling those in multiple lanes. The merging in the same lane can be done by decelerating the vehicle/platoon that drives ahead of others and accelerating the following vehicle/platoon. Otherwise, one of them could also retain their speed and wait for others to approach until the new platoon is formed.

IVC/V2X Communication Simulation

The main task of modeling the V2V connection is determining the communication delay in this study. However, modeling communication delay involves extensive studies. Researchers are working on mitigating that value to millisecond levels. Thus, the communication delay will be set as constant. But the value varies according to different information flow topologies (IFT). The following figure shows several basic IFTs. Figure 15.a shows the Predecessor Following approach. Figure 15.b shows the Bidirectional approach. Figure 15.c shows the Leader approach. Figure 15.d shows a mixed approach called the Predecessor-Following. More complex IFTs can be achieved by combining these basic IFTs. In this study, a leader-predecessor communication topology was used.

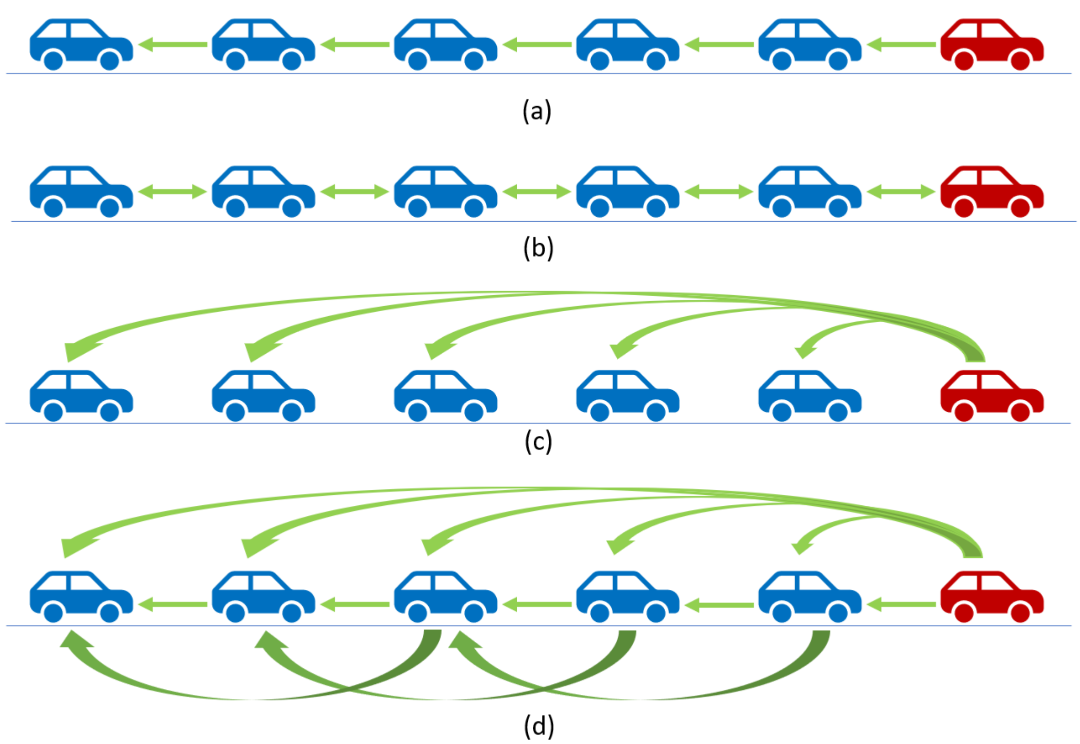

Figure 15. Basic Information Flow Topologies

In this study, OMNeT++ will be used as the communication simulator. As introduced above, OMNeT++ is a component-based simulator. An OMNeT++ model is built based on modules. Communication messages are sent between modules directly or through other modules [137]. An OMNet++ simulation model consists of module structure description, message definition, and other modules [138]. Modules are basically programmed in C++. With configuration files, files define how simulations are executed and with parameter values, the simulations can be conducted via the GUI interface or command-line interpreters. Moreover, benefiting from supporting parallel simulation execution, OMNeT++ is capable of being deployed to develop CAV-related simulations with traffic simulators. Integration with SUMO is one example of those, which was deployed in the developed system in this study.

Macroscopic simulation

The macroscopic simulations were developed on a popular traffic flow simulation software – Eclipse Simulation of Urban Mobility (SUMO).

SUMO is an open-source microscopic and continuous simulation package. Its capability includes but is not limited to simulating multi-modal traffic and handling large networks. The term - microscopic simulation means the simulation of traffic was conducted upon vehicle-level modeling. Such traffic simulation approaches differ from the simulation that explores the aspects of the flow of traffic rather than each vehicle. The strengths of SUMO include a) open-source, b) low CPU and Memory usage, c) popularity, d) accessibility to the developed multiplatform simulations, and e) adaptation to microscopic traffic simulation tools [139] [140] [141].

Due to its open-source traffic simulation tool, SUMO is inexpensive for users who develop simulations of traffic with CAVs. It could also be one of the reasons for the popularity of this simulation platform. The following table shows the number of search results of different traffic simulation tools on Google Scholar.

Additionally, SUMO has great portability to be integrated with other simulators. For instance, SUMO was originally developed based on C++ language. But with TraCI, a TCP-based client-server interface, users can develop applications by using Java and Python [142]. The mechanism of TraCI is establishing a portal and connecting with other applications/platforms through the portal. Moreover, also leveraged by TraCI, users can establish inter-platform simulations. With such integration between SUMO and MATLAB, users can develop vehicle-level modules in MATLAB and build traffic-level simulations in SUMO. Such approaches also enable SUMO to adapt to communication simulators, e.g., with OMNeT++. SUMO also accepts road networks drawn by other software.

Table 4. Numbers of publication found by searching (searching conducted by Nov. 2022, Google Scholar)

Traffic Simulation Software	Search Query			
	{NAME} + "autonomous vehicle"	{NAME} + "Inter vehicle communication"	{NAME} + "CAV"	{NAME} + "vehicle platoon"
SUMO	12,300	13,600	4,230	2,580
PTV Vissim	4,390	3,490	1,050	3,330
Aimsun	1,580	1,560	319	1,150
Paramics	1,360	1,830	169	1,190
MATSim	2,050	1,370	111	236
POLARIS*	3,970	7,300	902	520

Polaris is not exclusively used by POLARIS Transportation System Simulation Tool, so the numbers may not be accurate accordingly.*

By inserting the information that was extracted from vehicle-level models and developing traffic simulation scenarios, such as determining the proportions of different types of vehicles and roadway geometric information, scenarios for macroscopic simulations are built.

In the case study section, simulations are run repeatedly under similar environmental setups but with different traffic compositions and traffic patterns. By establishing basic statistical analyses, how some parameters impact the system will be examined. Input variables and factors include shares of different types of vehicles, number of lanes, speed limit, density, and average lag of communication. The output will include different traffic and pollution measures of effectiveness. Correlations among input and output variables will also be studied.

Inter-platform Simulation

Thus, according to the review conducted in previous sections of this chapter, most simulation frameworks are developed based on integrations of simulators that were focused on different topics. The combination of communication network simulation platforms and traffic simulation platforms can effectively reduce the requirement of substantial efforts in programming for exploring CAV's impact on traffic management or infrastructure design. In this study, such a strategy is deployed as well. The integration tool used is Veins [107].

Veins is one of the most utilized frameworks that manages the integration of communication simulation software and traffic simulation software. The basic integration mechanism of multiplatform simulation provided by Veins is rooted in OMNeT++'s module-based characteristic. A corresponding module will be created in

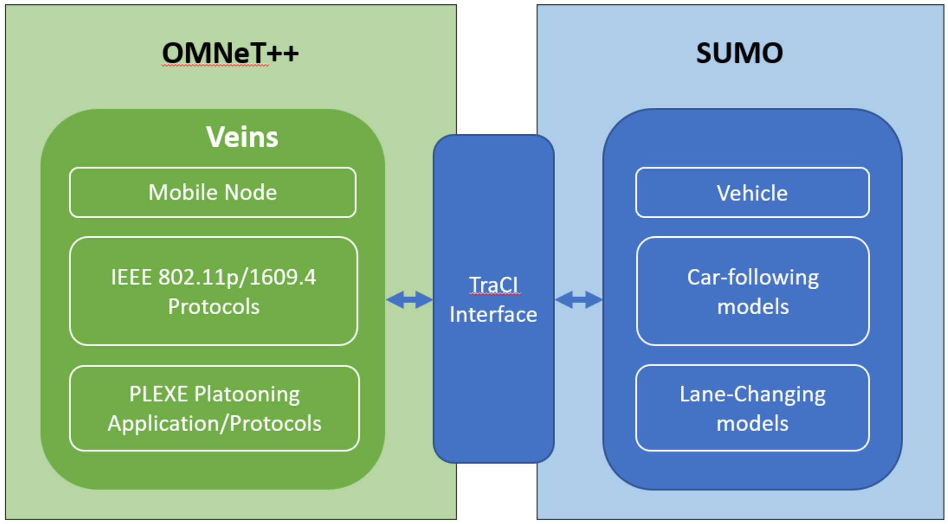

Figure 16. Architecture of Integrated Simulation based on PLEXE [88]

the OMNeT++ simulation model once a vehicle (mobile node) is emitted into the SUMO simulation network. The architecture of a vehicle's communication module under Veins is APP-MAC-PHY. The MAC and PHY layers explicitly follow IEEE 802.11p and IEEE 1609.4 protocols. Those are the core of IEEE WAVE and ETSI ITS-5G.

Based on Veins, the extension called PLEXE provides a simulation framework for CAV platooning upon SUMO and OMNeT++. PLEXE follows the basic integration mechanism of Veins. But the vehicle's mobility information from SUMO are gathered and used not only for updating corresponding module's status in OMNeT++ but also for platooning protocols and applications [88]. In the meantime, platooning maneuvers are actuated and simulated in SUMO. In the following case study chapter, platooning applications developed based on PLEXE are explained in detail. Figure 16 demonstrates the architecture of simulation developed upon PLEXE. Figure 17 shows the logic flow of the instance of the simulation framework built in this study.

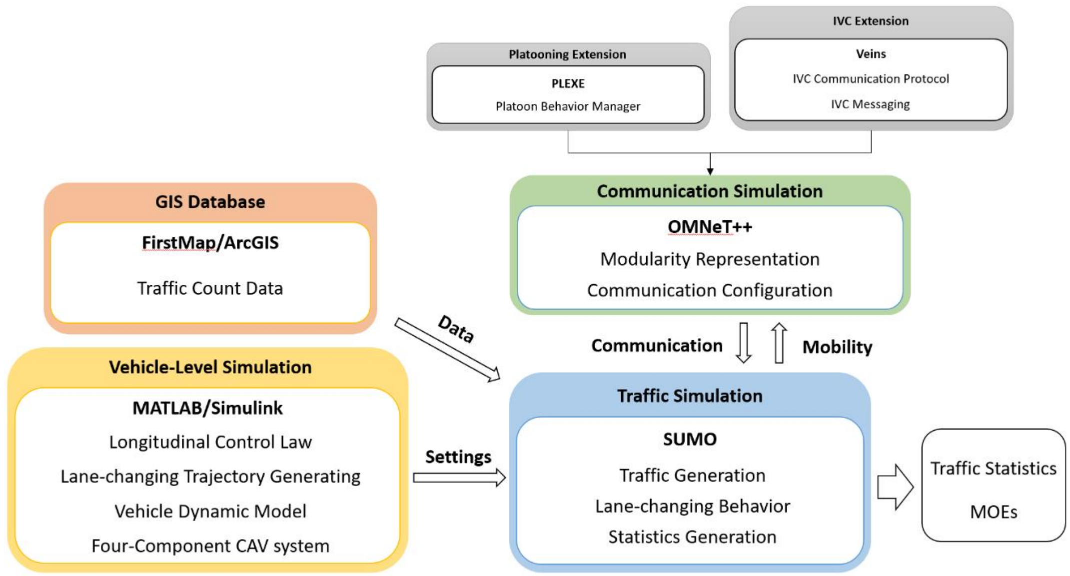

Figure 17. Logic Flow of the Instance of the Proposed Simulation Framework in This Study

Chapter 4

CASE STUDIES

In this chapter, two case studies are conducted. The purpose of these works is to determine the impacts of CAV platoons on traffic. The scenarios are set to represent the impacts of the shares of CAV platoon change. However, before the case studies, calibrations are conducted to determine the basic setting-ups of models, including the vehicle dynamic model, car-following model, and lane-changing models. Calibrations are conducted by the microscopic simulations built on MATLAB/Simulink.

Calibration

The first step of conduct calibration for finding a proper set of parameters for macroscopic simulation (traffic-level simulation) is to develop a model to represent different types of vehicles. In this study, three types of vehicles are modeled. They are human-driven vehicles (HV), autonomous vehicles without/deactivated IVC functions (AVs), and platooning-enabled CAVs (P-CAVs). The first part of calibration is to determine the parameters for the longitudinal control model regarding the different types of vehicles.

Longitudinal control model

In this study, three longitudinal control laws are implemented to represent the longitudinal control strategies of different types of vehicles. They are cruise control

(CC), adaptive cruise control (ACC), and cooperative adaptive cruise control (CACC). The basic control laws of these three longitudinal control strategies were introduced in the last section 3.3.2. However, since the calibrations are conducted to determine the parameters from a macroscopic simulation viewpoint, the selected models are modified models which are adopted by the macroscopic simulation platform – PLEXE [88]. The control law to determine the acceleration rate of a vehicle with CC (human-driven vehicles) is:

$$\ddot{x}_{des}(t) = -k_p(\dot{x} - \dot{x}_{des}) - \eta \tag{25}$$

Where k_p is the gain of proportional controller, $\dot{x}$ and $\dot{x}_{des}$ indicate the ego speed and desired speed of the vehicle respectively. η is a random number that reflects the imprecisions of speed measurement and control actuator.

For ACC vehicles or platoon leaders, the selected longitudinal control law is:

$$\ddot{x}_{des} = -\frac{1}{T_h}(\dot{\varepsilon} - \lambda\delta) \tag{26}$$

T_h stands for the time headway in seconds. $\dot{\varepsilon}$ is the relative speed between the ego vehicle and its preceding vehicle. δ indicates the distance error between the ego vehicle and its preceding vehicle. λ is a design parameter needed to be determined. The following two equations (Eq. 27 and Eq. 28) show the formula for determining $\dot{\varepsilon}$ and δ. l_{pre} is the length of the preceding vehicle.

$$\dot{\varepsilon} = \dot{x}_{ego} - \dot{x}_{pre} \tag{27}$$

$$\delta = x_{ego} - x_{pre} - l_{pre} + T_h\,\dot{x}_{ego} \tag{28}$$

Vehicles' longitudinal motions are controlled with CACC strategy. Typically, for the follower vehicles in a platoon, the following longitudinal control law is deployed.

$$\ddot{x}_{i\_des} = (1 - C_1)\,\ddot{x}_{i-1} + C_1\,\ddot{x}_l - \left(2\xi - C_1\left(\xi + \sqrt{\xi^2 - 1}\right)\right)\omega_n\,\dot{\varepsilon}_i - \left(\xi + \sqrt{\xi^2 - 1}\right)\omega_n C_1(\dot{x}_i - \dot{x}_0) - \omega_n^2\,\varepsilon_i \tag{29}$$

$\ddot{x}_{i\_des}$ means the desired acceleration of the i th vehicle in a platoon. $\ddot{x}_{i-1}$ and $\ddot{x}_l$ stand for the acceleration of the $i - 1$ th (the preceding vehicle of the i th vehicle) and the leader vehicle of the platoon. $\dot{x}_i$ and $\dot{x}_0$ are the velocities of the i th vehicle and leader vehicle. C_1 is a ratio parameter representing the importance of leader vehicle's acceleration versus the preceding vehicle's acceleration. ξ stands for the damping ratio of the platoon string.

As mentioned before, platooning CACC vehicles follow the fix spacing strategy as defined. Thus, the method to calculate the distancing error sightly differs from the one to calculate ACC. Eq. 30 shows how to calculate the distance error ε_i of the i th vehicle in a platoon for CACC longitudinal control law.

$$\varepsilon_i = x_i - x_{i-1} - l_{i-1} + \text{gap}_{des} \tag{30}$$

$\dot{\varepsilon}_i$ indicates the speed difference between the i th and the $i - 1$ th vehicle. Eq. 31 shows the way to calculate the difference within a CAV platoon. This is the same for ACC vehicles (Eq. 27).

$$\dot{\varepsilon}_i = \dot{x}_i - \dot{x}_{i-1} \tag{31}$$

In MATLAB/Simulink, the longitudinal control functions are coded as MATLAB functions and implemented into Simulink vehicle mobility models. The following figures (Figure 18 - 20) show the screenshots of the three control law modules introduced above.

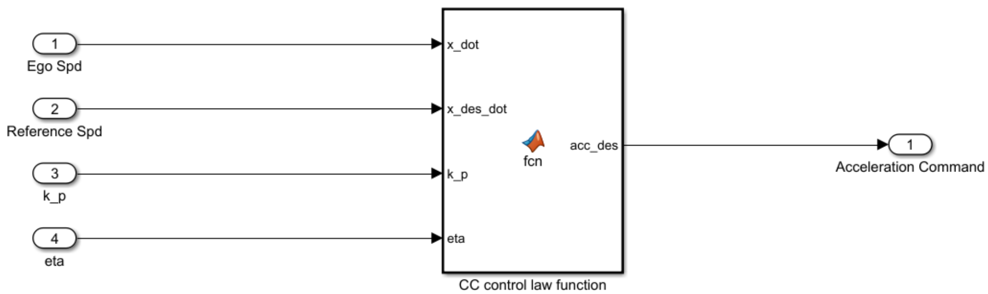

Figure 18. Cruise Control (CC) Longitudinal Control Module

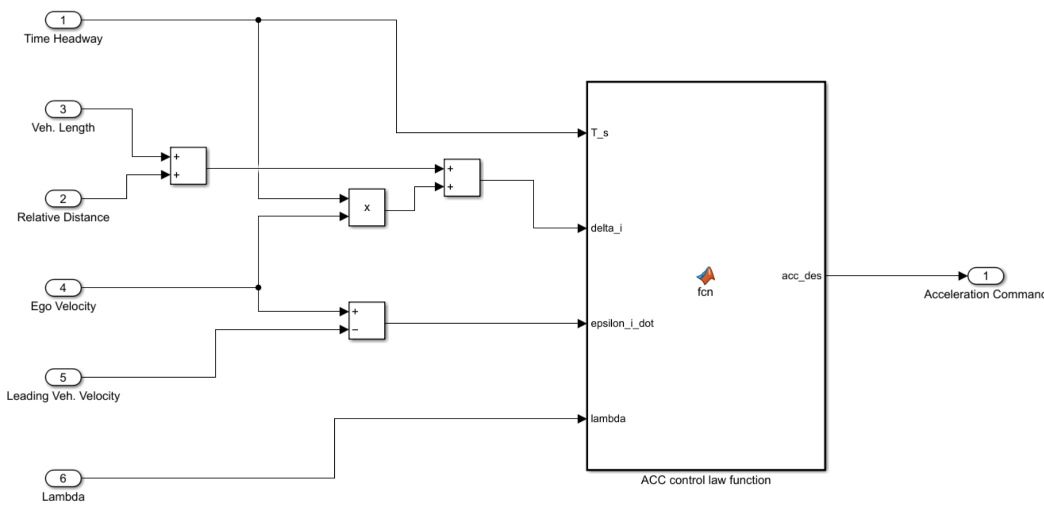

Figure 19. Adaptive Cruise Control (ACC) Longitudinal Control Model

The longitudinal control modules introduced above are combined to perform the vehicle's longitudinal control under different conditions with different types of vehicles.

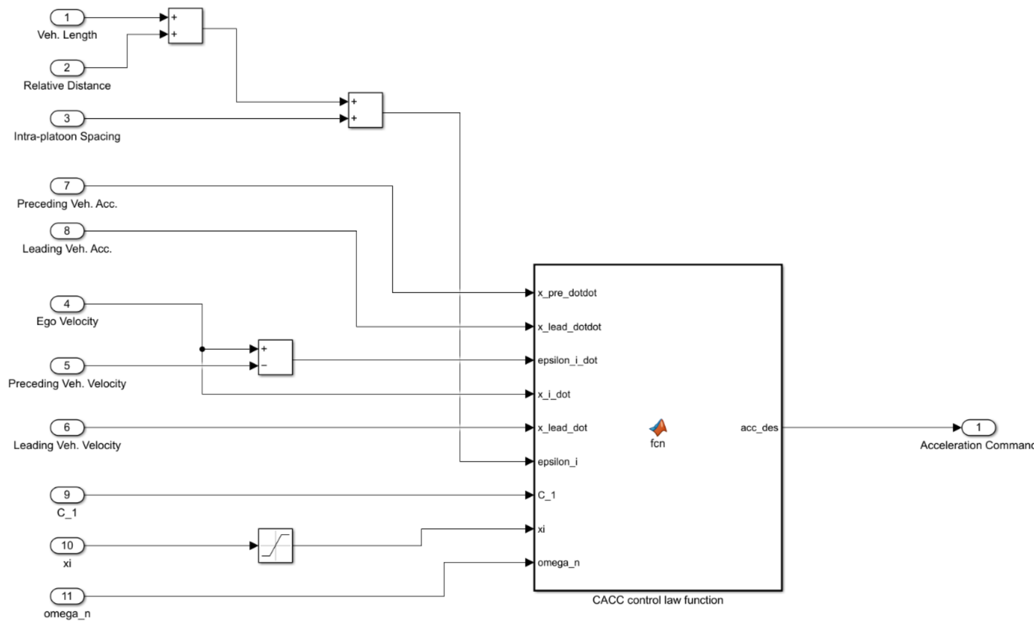

Figure 20. Cooperative Adaptive Cruise Control (CACC) Longitudinal Control Module

HVs, AVs, and CAVs without a connected preceding vehicle used the combination of CC and ACC longitudinal control modules. CAVs with a connected preceding vehicle and follower vehicles in platoons used the combination of CC and CACC longitudinal control modules.

From the behavior selection perspective, the control task is to choose a proper longitudinal control module and perform control behaviors. Finite state machines (FSMs) are designed and deployed to perform behavior selections. For the vehicles using CC/ACC longitudinal control laws, the FSM consists of three states – emergency braking mode, CC driving mode, and ACC driving mode. The transitions among states are triggered when the inter-vehicle distance between the ego vehicle and its preceding vehicle increase/decrease to certain thresholds. The following graph (Figure 21) shows the state diagram of CC/ACC longitudinal control FSM. As shown, buffers are added to improve the robustness of the FSM. Transitions between CC and ACC modes depend

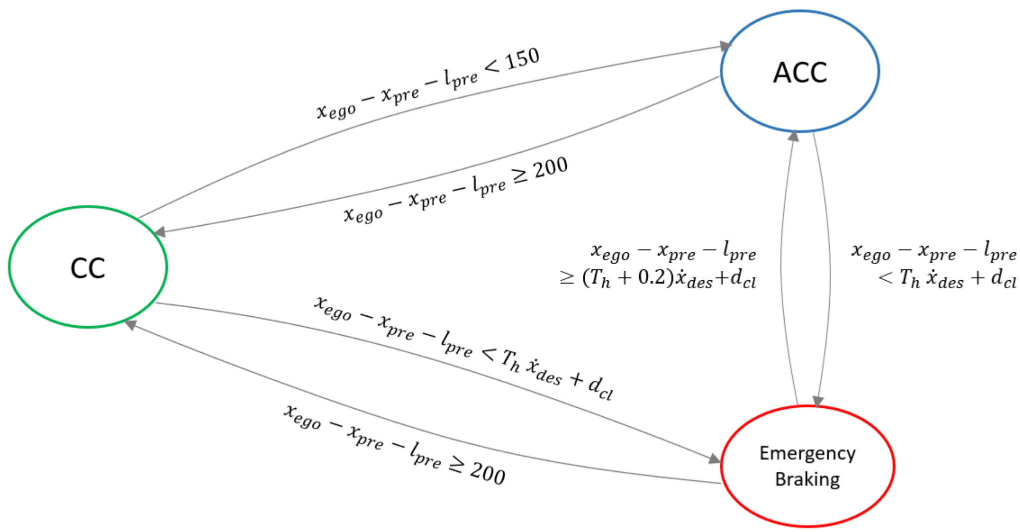

Figure 21. State Diagram of FSM for CC/ACC Longitudinal Control Module

on if the onboard perception systems have detected a preceding vehicle. As mentioned before, when a preceding vehicle has been detected, and the gap is less than 150 meters, the vehicle switch from CC to ACC mode. In reverse, when no preceding vehicle has been detected within 200 meters ahead, the vehicle switches back to CC mode. The number is based on the maximum detection ranges of both LiDAR- and Radar-based ranging techniques. Emergency brake mode is activated once the distance to the preceding vehicle is less than the desired distance based on desired time headway plus a safe clearance. Time headway varies from 1 to 2 seconds for most commercial ACC modules [143]. Safe clearance is the distance a vehicle should maintain towards its preceding vehicle after a completely stopped vehicle. In ACC mode, to improve safety, the vehicle's deceleration is the minimum of the desired acceleration calculated by CC control law and ACC control law.

A similar FSM has been deployed for CC/CACC vehicles' longitudinal control modules. Figure 22 demonstrates the mechanism of the FSM for the ith vehicle in a

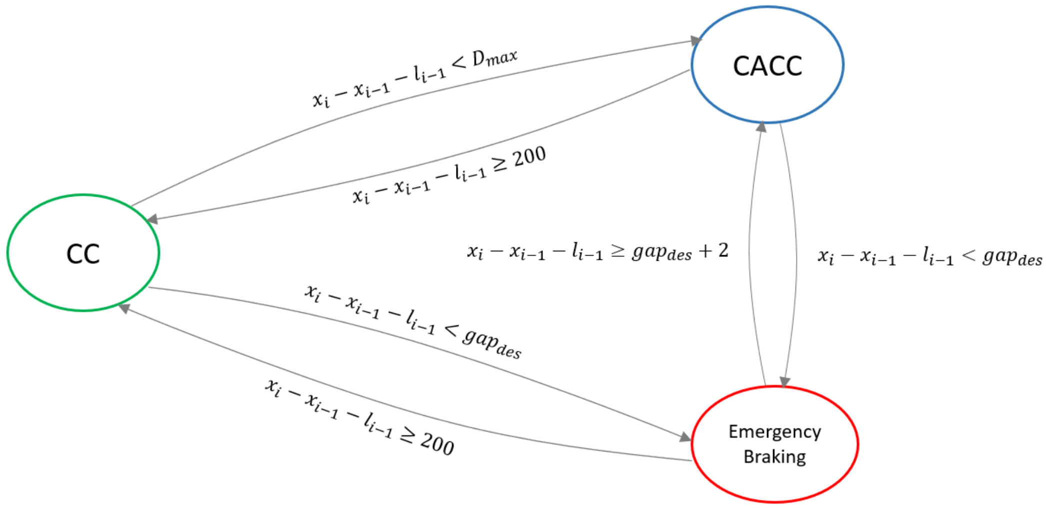

Figure 22. State Diagram of FSM for CC/CACC Longitudinal Control Module

platoon. As shown, due to the deployed spacing strategy, CACC's transition from the CC/CACC modes to the emergency braking mode depends on the desired constant spacing between vehicles. The transition between CC and CACC modes is triggered by comparing the space between two vehicles to the maximum platoon catchup distance and maximum perception distance without connection to the preceding vehicle. In CACC mode, the vehicle's deceleration is the minimum of the desired acceleration calculated by CC control law and CACC control law.

Another element to make the control model realistic is adding a transfer function to smooth the changes of acceleration command to avoid violating physical rules. However, finding out a well-performed transfer function is beyond this study's scope. Thus, the following first-order low-pass filter transfer function is used by Simulink for the adaptive cruise control example module and is also presented in [120]. The formula approximates the dynamics of the throttle body and vehicle inertia.

$$G = \frac{1}{0.5s + 1} \tag{32}$$

Then, by integrating the continuous acceleration inputs, the longitudinal control module keeps feeding the speed of the vehicle to a three-degree-of-freedom vehicle body dynamic module provided by Simulink. This module is used to assemble the longitudinal and lateral dynamics of the vehicle. More detail is provided in the following section on the lateral motion control. The demonstration of the whole vehicle motion model is presented in the following section as well.

Parameters k_p, η, λ and C_1 for longitudinal control laws are tuned before conducting the next level calibration.

Lateral control model

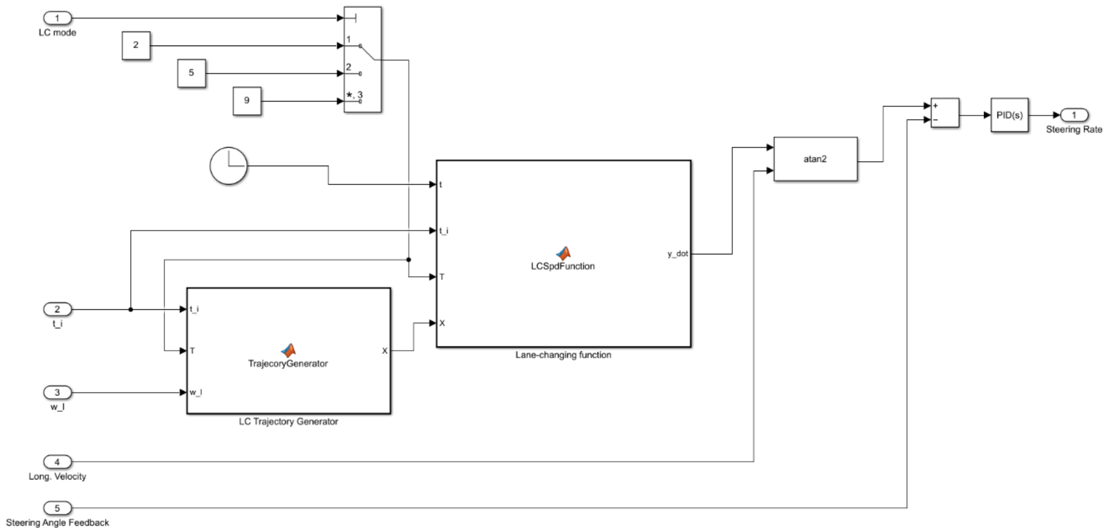

Figure 23. Lane-changing Simulink Subsystem

In this study, due to the exploration of the impact of CAV platoons on access-controlled highways, the lateral control model is simplified to reduce the computational burdens. The objective of modeling the lateral control in this study is to define when a lane-changing behavior is scheduled and how it will influence the traffic on the target lane.

The selected behavior for exploring lateral motion is the lane-changing maneuver. A lane changing maneuver consists of two major steps – trajectory planning and execution. To ease the computational burdens, trajectory selection is deployed as a standard trajectory generation algorithm. Three lane-changing trajectories are proposed regarding three levels of aggressiveness. Equations 19 to 21 are used to generate the trajectories. In MATLAB, a function called "TrajectoryGenerator" is created to solve the parameters for generating trajectories. The three selected durations for lane-changing maneuvers are 2 seconds, 5 seconds, and 9 seconds. The lane width is set to

3.66 metes/12 feet (the commonly used width for highway lanes in the USA). These numbers are selected based on the statistics of human-driven vehicles' behaviors [144]. The 2-seconds maneuver refers to the most aggressive strategy. The 5-seconds maneuver represents the average situation. The 9-seconds maneuver is selected when traffic is less intensive and provides comfort to the driver/passengers.

Calibration is conducted to find out when a vehicle takes lane-changing behavior, if and how the following string in the target lane consisting of different types of vehicles would be affected. A subsystem in Simulink is built and implemented to represent the process. Figure 23 is the screenshots of the lane-changing subsystem.

As shown in the picture, the subsystem takes five inputs – the lane changing mode selection (LC mode), the time to start lane changing maneuver (t_i), lane width (w_l), longitudinal velocity (Long. Velocity), and steering output feedback. A multiport switch takes the lane-changing mode selection input to determine how long the maneuver should take. The three options refer to the three levels of aggressiveness. Option 1 is the most aggressive maneuver, which lasts 2 seconds. Option 2 is the moderate maneuver that lasts 5 seconds. And the third option is the most relaxed maneuver, which takes 9 seconds. The duration of the maneuver feeds to two function modules. One is the trajectory generation module, and the other is the lane-change function module. The prior solves the Equations 19 to 21 formulas by inserting the initial state and the final state of the vehicle. The state information includes lateral position, lateral velocity, and lateral acceleration. The output of this module is the parameters (a_1 to a_5) of Equations 19 to 21, then feed to the lane-changing function model to calculate the desired lateral velocity dynamically. The lane-changing function module also takes the simulation time and time to start maneuvering as inputs. The embedded "if" logic of the module determines if the vehicle is changing its lane and calculates the desired lateral velocity during the maneuver. Then after taking the four-quadrant inverse tangent of the ratio of longitudinal and lateral velocity and subtracting steering angle feedback, the desired steering angle for the current time step is determined. Then a PD controller is added as a filter to achieve a more smooth and more realistic control performance.

As mentioned above, the lateral control output is sent to the 3-DOF vehicle body block that was provided by Simulink to generate comprehensive motion data of a vehicle. The vehicle's specifications are set to:

i. Two axles and two wheels for each axle
ii. Vehicle mass of 1,800 kilograms
iii. Longitudinal distance from the center of mass to front and rear axles of 1.4 and 1.6 meters
iv. Height of center of mass to the axel plane of 0.35 meters

This setting reflects the specifications of a typical sedan.

Scenarios

Multiple scenarios are established for running the microscopic simulations to calibrate the model. Scenarios are set based on different platoon sizes, spacing strategies, control parameters, and types of the preceding vehicles.

Three levels of platoon size are selected – 2-vehicle platoon, 5-vehicle platoon, and 8-vehicle platoon. For ACC control module, the time headways have 3 levels – 1 seconds, 1.5 seconds, and 2 seconds. For CACC control module, spacing strategies have four levels – 2 meters, 5 meters, 12 meters, and 20 meters.

The one scenario is built to reflect a slow-moving vehicle being detected ahead of a string of vehicles in a single-lane roadway section. The speed difference between the string and the slow-mover varies from -5 to -20 mph in every five mph interval. The Initial position of the slow-moving vehicles is 300 meters ahead of the leading vehicle of the string. The desired speed of the string is set to be 75 mph (33.5 m/s).

The other scenario is built to reflect how a lane-changing vehicle impacts a string of vehicles behind it with or without connectivity. A vehicle running slower than the string of vehicles takes the lane-changing behavior and enters the lane of the string. The speed difference of the lane-changing vehicle is 5 to 20 mph slower than the string. The lane-changing vehicle is set prior to the string with longitudinal distance. The numbers are set based on different time headways – 1 second, 1.5 seconds, 2 seconds, and 3 seconds.

In Simulink, vehicle state information (e.g., velocity, acceleration, and position) are shared and transmitted among vehicles directly. Communication packet loss and missing perception are not introduced into these microscopic models directly. However, arbitrary signal delays are inserted. Based on the state of the art of perception technologies and vehicular communication technologies, the delays for communications are set to between 1 ms (millisecond) to 10 ms. Signal delays for the perception of speed and velocity are set to the range from 5 ms to 20 ms [145] [146] [147] [148] [149].

$$f_{hn}(x;\ \mu,\sigma) = \begin{cases} \sqrt{\dfrac{2}{\pi\sigma^2}}\ e^{-\frac{(x-\mu)^2}{2\sigma^2}} & y \geq \mu \\ 0 & y < \ \mu \end{cases} \tag{33}$$

Where μ stands for the lower bond of the distribution. σ is a parameter for determining the variance. For communications delays, μ is set to 1 and σ is set to 5. For perception delays, μ is set to 5, and σ is set to 5. These are hypothetical settings.

Table 5. Summarizes the setting items of the calibrated model

Setting Items	Values	Other Information
Platoon Size (number of Veh.)	2, 5, 8	-
ACC Time Headway (sec)	1, 1.5, 2	-
CACC Constant Spacing Distance (m)	2, 5, 12, 20	-
Velocity of slow-moving vehicle (mph)	55, 60, 65, 70	-
Communication Delay (ms)	1 to 10	PDF: $f_{hn}(x;\ 1,5)$
Perception Delay (ms)	5 to 20	PDF: $f_{hn}(x;\ 5,5)$

Evaluation and Results

To evaluate the performance of the models, two measurements of effectiveness are used. One is the vehicle's maximum gap error (time or distance) towards its preceding vehicle. The other is duration of speed synchronization. The first MOE is defined as:

$$\varepsilon_{i\_int} = \int_{0}^{T_{sim}} \varepsilon_i(t)\, dt \tag{34}$$

Where ε_{i_int} means the gap error integral over the simulation time of the vehicle i. $\varepsilon_i(t)$ indicates the gap error of the vehicle i at time t. T_{sim} is defined as the duration of the entire simulation.

The second MOE measures how long it takes a vehicle to complete the speed synchronization to its preceding vehicle. The start time of speed synchronization is considered when the speed drops 0.1 m/s from the initial speed. Then the counting ends when the vehicle's speed is 0.1 m/s higher than the preceding vehicle's speed. Such buffer values are installed to improve the accuracy of the assessment.

The calibration simulation based on the slow-moving vehicle scenario showed that when using different spacing distances of a platoon, the impact on the durations of speed synchronization is subtle among 2-meters, 5- meters, 12- meters distance headway spacing strategies. However, when deploying 20-meters spacing strategy, there are fluctuations in speed during the speed transition periods. Furthermore, the duration of speed synchronization dropped by 35 % when compared to when other strategies were deployed. Increasing the platoon leader vehicle's desired ACC time headway, the integral spacing error of the follower vehicle decreased. The drop for each adjacent level was between 4% to 5%.

The results also found that either communication or perception delays brought limited impacts to the MOEs to all spacing strategies. The changes in gap error integral dropped by 10% to 15%. The impacts on the duration of speed synchronization were subtle.

One of the important findings of the calibration process is the state transfer condition between ACC car following and emergency braking. The relative distance to the preceding vehicle has to be less than $T_h \, \dot{x}_{ego}$, otherwise, the ACC will be too sensitive to synchronize its speed to its preceding vehicle. It causes frequent deployment of the braking system.

The simulation based on the lane-changing scenario showed that the longitudinal distance from the string to the lane-changing vehicle impacts both MOEs significantly when it goes below the state transition condition from CC state to emergency braking state. The influences on the integral of gap error became steady when the lane-changing behavior took place $T_h \, \dot{x}_{ego}$ before the leader of the platoon. The duration of speed synchronization also reached the minimum at that point.

Thus, the selected parameters based on calibration are listed in the following table. k_p, CC control law function's parameter is set to 0.2. λ for ACC control module is set to 0.1. There are three parameters for the CACC control law function. ξ, as the damping rate, which is set to 1 to avoid inappropriate damping behavior. C_1 is set to 0.7 to make the leader vehicle play a more important role when determining the vehicle's longitudinal control. ω_n as the bandwidth of the control system is set to 0.2 Hz to perform the common condition.

The condition to change the vehicle's drive mode from ACC to emergency braking mode is adjusted to a constant value – 25 meters, to guarantee the performance of the control system.

A 5-meters fixed spacing strategy is selected for operating the platoon. The leader vehicle or ACC vehicles deployed a 1-second time headway spacing strategy. These strategies were found to be safe enough under the scenarios explained above. Thus, such strategies are expected to improve the traffic when implemented in macroscopic simulations.

Case Study I:

Northbound I-95/Delaware Turnpike, From Delaware House Travel Plaza to Christiana Interchange

This first case study's infrastructure is developed to represent a section of the US Interstate highway I-95. The section is located in the New Castle County, Delaware. The I-95 corridor is a major north-south interstate highway along the East Coast of the United States that runs from Florida to Maine. The total length of I-95 is over 1,900 miles. In Delaware, the section of I-95 is also known as the Delaware Turnpike. It's the main highway for the north region of Delaware. Figure 24 shows the satellite map of the highway section.

Figure 24. Studied I-95 Section in New Castle County, Delaware

Scenario

The basic setup is based on the existing infrastructure of the I-95. The studied section is the northbound roadway section from the Delaware House Travel Plaza to the

Christiana Interchange. It's a four-lane road section with three exits and one entrance. Tapered and parallel lane(s) are added for splitting traffic at exits and merging traffic at the entrance. Based on existing facilities, illumination is assumed to be sufficient during the nighttime. Asphalt pavement provides good friction factors and drainage capability.

Based on such information, a SUMO network is developed. Figure 25 shows the road network which is built to represent the studied roadway section (the lower bond) in SUMO. The opposite bound is also built in SUMO, but only hypothetical traffic is added, and analyses are not conducted based on the traffic on that bound. In Figure 25, even the lane transition looks mis-aligned, but according to observations during the simulations, vehicles travel through such areas as through traveling on straight lanes.

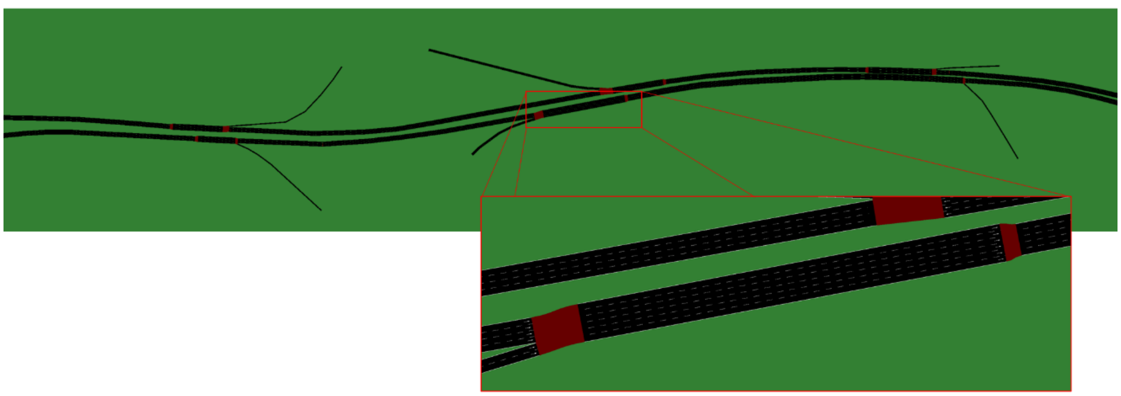

Figure 25. SUMO Road Network Built for Representing the Studied Roadway Section.

The land uses of the nearby areas are diverse. As shown in Figure 26, the land uses of nearby areas include commercial (red), residential (yellow), and recreational. Thus, the traffic on this section of the freeway is considered to have a high penetration of large vehicles (trucks and buses). The land-use/land-cover information is gathered from the FirstMap GIS Database. Figure 27 shows two screenshots of the interface of

Figure 27. Land-use Map of the Nearby Areas.

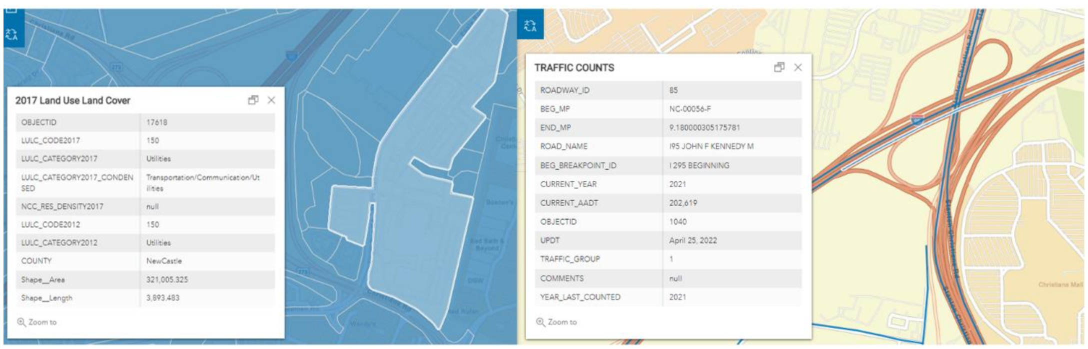

Figure 26. Land-Use/Land-Cover (Left) and Traffic (Right) Information Presented via GIS Platform/Database

the FirstMap/ArcGIS and examples of the common way to present traffic and other spatial information via the GIS platform.

According to the data released on Geographic Information System (GIS) platform, the AADT for this section (both directions) for 2021 was 202,619 vehicles. According to DelDOT’s report [150], the annual growth factor of all interstate highways, freeways, and expressways in Delaware was 1.305 for 2021. However, the growth factor for 2020 was 0.795. One of the reasons for such a dramatic fluctuation in growth factors is the COVID-19 pandemic. Thus, the growth factor is estimated by calculating the median of the growth factors for 2016 to 2019. The value is 1.008. The k-factor, which is used to calculate the Design Hourly Volume, is 8.3. The D-factor, the factor to determine the Directional Design Hourly factor, is 57%. The percentage of trucks (both single-unit and combination units) is 9.3%. Thus, the directional hourly design volume for inserting vehicles into the network is calculated and rounded to 9,663 vehicles/hour (Veh./hr). Table 6 summarizes the current traffic information. Due to the method of inserting the vehicles into the macroscopic model, the design volumes are converted to the probability of introducing a vehicle into the network at the beginning point of the studied road section. Table 7 shows the directional design hourly volumes for the two types of vehicles and their corresponding probabilities of a vehicle entering the network from the 5-year, 10-year, and 15-year perspectives.

Table 6. Input Information for Future Traffic Estimation

Year	**AADT**	**D-Factor**	**K-Factor**	**Truck Percentage**	**Growth Factor**
2021	202,619	8.3	5.7	9.3%	1.008

Table 7. Traffic Prediction and Macroscopic Simulation Inputs

Year	**2026**	**2031**	**2036**
DHV	17501	18212	18952
DDHV	9976	10381	10803
DDHV per Lane	2494	2595	2701
Probability of a vehicle entering per second	0.693	0.721	0.750
Probability of a non-truck vehicle entering per second	0.628	0.654	0.680
Probability of a truck entering per second	0.064	0.067	0.070

Along with the volume, in order to explore the impacts of the CAV platoons impacts toward the traffic, another element chosen to be varied for generating the simulation scenarios is the percentages of different types of vehicles. For different types of vehicles, it means vehicles deployed distinguish levels of driving autonomy. As was previously described, human-driven vehicles, and autonomous vehicles without communication with others are considered. The longitudinal control laws of vehicles deploy the calibrated model described in the first section of this chapter. Three laws are implemented – CC, ACC, and CACC. The lane-changing behavior was managed by the SUMO. The Lane-changing module embedded in SUMO provides a comprehensive and effective lane-changing behavior selection and performing system.

To configure a simulation model, there are a few steps needed to be done on different simulators. In SUMO, different vehicle configurations need to be completed. The different vehicle types are added in a SUMO route configuration file. In that file, information about acceleration, deceleration, driver's imperfections, vehicle length, acceptable minimum gap, the probability of entering the road network per second, the intention and magnitude of speeding, and destination are defined. Moreover, the vehicle car-following model and longitudinal controllers are selected by setting corresponding parameters in the configuration file. The way to inject traffic into the system is also defined in the SUMO configuration file. For this case study, to eliminate the variance of the scenarios, platoons were injected into the network at consistent time points. Other traffic components, human-driven vehicles, and autonomous vehicles were injected probabilistically.

Additionally, platooning behaviors and communication protocols are defined in the OMNeT++ configuration file for each simulation. In this case, autonomous vehicles and human-driven vehicles are set as non-cooperative components of the traffic. No communication node is created in the communication simulator to represent those types of vehicles. As was mentioned before, all CAVs attain a representing node the in OMNeT++ simulation model.

Table 8 summarizes the crucial parameters of both mobility and communication models.

Table 8. Mobility model and communication model parameters

Parameters	Values	Information
k_p	0.2	CC control law parameter
λ	0.1	ACC control law parameter
ξ	1	ACC control law parameter
C_1	0.7	CACC control law parameter
ω_n	0.2	CACC control law parameter
Actuator delay	0.5 s	Denoted as sigma
Error Rate	0.5%	Rate of error meets 99.5% communication success rate
Bitrate	6 Mbps	Numbers of but transmitted per second
Beaconing Interval	0.1 s	Interval for communication units to update the existence
Max Platoon Size	8	Maximum Number of Vehicles in a platoon
Update Interval	0.01	The time interval for update simulation status between simulators
Speed Limit	105km/h (65 mph)	-
Speed factor	norm(1.15, 0.12)	Probability of speeding, normal distributed with a mean of 120 km/h (75 mph) and a standard deviation of 15 km/h (9 mph)
Acceleration Rate	2.3 m/s^2	-
Deceleration Rate	4.5 m/s^2	-

Parameters	Values	Information
Length (passenger car)	5 m	Average vehicle's length of passenger cars [151]
Length (Truck)	13.5 m	Average length of interstate semitrailer [151]

Another factor for building simulation scenarios is the percentage of CAV platoon traffic. In this case study, four levels of percentages of CAV platoon traffic are a) 10%, c) 30%, and 50%. But only passenger platoons are simulated. Similar to truck traffic injection, different size CAV platoons were injected into the network by defining the probability of injecting one CAV per second.

Results

To validate the simulation model, field works have been conducted to collect time headway of the traffic flow. The description of fieldwork and raw data are presented in Appendix A. When comparing to the probability of a vehicle entering the network, such settings have been validated. Moreover, validation simulations (traffic without CAV traffic) have been conducted by injecting traffic with the settings presented above. The comparison of volume and travel time for both real traffic and simulated traffic showed that the simulated traffic performed better than the real traffic. That might be due to: a) the uniformity of the vehicle's acceleration, deceleration, and dimension, b) the lack of vehicle type diversity, and c) the lack of imperfectness of vehicle maneuver. However, two sample t-tests with unequal variances results showed that the p-value is about 0.07, greater than 0.05. Thus, it is not statistically significant to reject the null hypothesis and assert that the means of the two groups of data are similar. But the relatively low p-value means there is a high chance the two groups of data are similar. Thus, further steps are taken to complete the case study.

The result of simulations showed that the benefits on average travel speed grew as well as the total traffic volume grew with the same percentages of platoon traffic. However, with 10% of CAV platoon traffic, the average travel speed increased by about 0.2% for 2026, 0.1% for 2031, and 0.2% for 2036 traffic. With both 30% of CAV platoon traffic, the average travel speed increased by 0.9% for 2026 traffic, 1.4% for 2031 traffic, and 1.6% for 2036 traffic. Increasing the percentage of CAV platoon traffic grows to 50% out of the total traffic, which could increase the average travel speed by 3%, 3.2%, and 4%, respectively for 2026, 2031, and 2036 volumes.

Then more scenarios were built to conduct simulations with higher traffic. When adding 40% more traffic to the 2036 traffic, the travel speeds increased similarly compared to the results with the predicted situations based on traffic growth rate.

However, the time of vehicle driving below the desired speed dropped more observably. In general, when 30% to 50% of traffic was CAV platoons, the duration of vehicle driving below the desired speed dropped by 20% on average.

Conclusion

Based on current traffic patterns and infrastructure design, when introducing CAV platoons, the overall traveling speed under relatively high-volume situations may gain improvements. However, the increase might not be significant when the traffic remains steady.

Case Study II:

CAV Platoon Dedicated Lane(s): Northbound I-95/Delaware Turnpike, From Delaware House Travel Plaza to Christiana Interchange

This second case study is developed based on the same roadway section in Case Study I. The purpose of this case study is to investigate the impact of deploying a dedicated lane(s) for the CAV platoon traffic. Scenarios are built based on the first case study. The design volume representing 2036 is selected to insert vehicles.

The modifications on the infrastructure consist of either adding or converting a lane(s) to serve as dedicated CAV platoon lane(s). The following table (Table 9) shows how lane settings differ. In this case study, dedicated lane(s) are constructed as separated express lane(s). As defined in FHWA documents, such a lane(s) is designed to separate high-speed, non-stop traffic. Thus, the speed limits for the dedicated lanes (75 mph) are usually higher than the regular lanes (65 mph) in the same section of a highway.

Table 9. Dedicated Lanes Settings and Percentage of Platooning Traffic

Settings	Percentage of CAV platoon vehicles	Converted Lanes	Added Lanes
A	10%	1	-
B	15%	1	-
C	20%	1	-
D	25%	1	-
E	30%	1	-
F	10%	-	1
G	20%	-	1
H	30%	-	1
I	40%	-	1
J	50%	-	1
K	15%	1	1
L	25%	1	1
M	35%	1	1
N	45%	1	1
O	55%	1	1

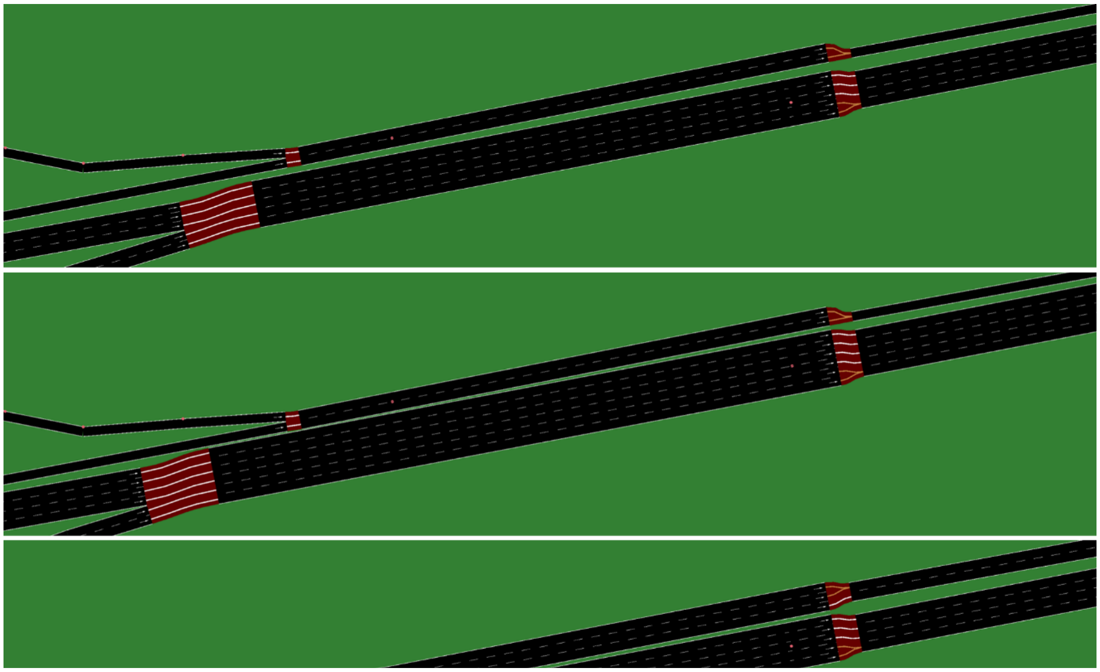

Figure 28. Dedicated Lane Alignment with (Top) One Lane Converted
(Mid) One Lane Added,
(Bottom) One Lane Converted and One Lane Added

The percentage of platooning CAVs varies depending on the number of dedicated lanes. The corresponding simulated percentages of platooning CAVs versus the numbers of dedicated lanes are listed above in Table 9. Figure 28 demonstrates the lane alignment of three different settings.

Results

The simulation results showed that when speed limits increased for the CAV platoon dedicated lane(s), CAV platoon flow could achieve better travel time than the regular lanes. Moreover, converting a lane to dedicated with low percentages (settings A and B) of CAV platoon traffic increased the regular lanes' burden (lowered the capacity). But as the percentage increased, the average travel speed of all types of traffic increased. When adding a dedicated lane instead of converting one, settings F, G, and H showed the benefits for both types of traffic. However, when the percentage went higher, such a pattern solely improved the regular traffic, but situations of CAV platoon traffic remained similar for both adding and converting. As a result, improvement for both types of traffic has been achieved. As the percentage of CAV traffic increased, the improvement increased gradually.

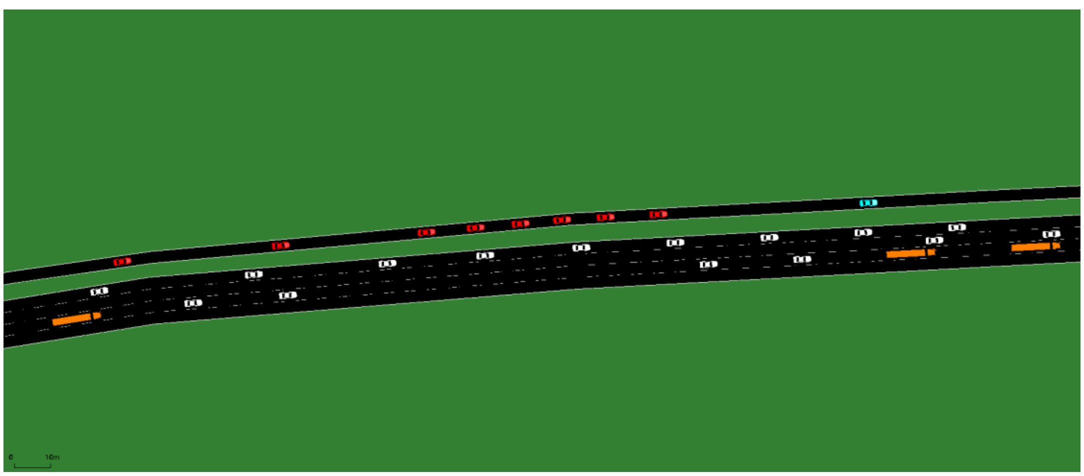

Figure 29. Screenshots of CAV Platoon Traffic running on the Dedicated Lane

Figure 29 shows the traffic of the scenarios with setting E. In the figure, the traffic on the separated lane is CAV platoon traffic. Red vehicles indicate the vehicle was considered a member of a platoon. The Vehicle in cyan is not in a platoon. In regular traffic, white vehicles represent passenger cars as well as orange vehicles represent trucks.

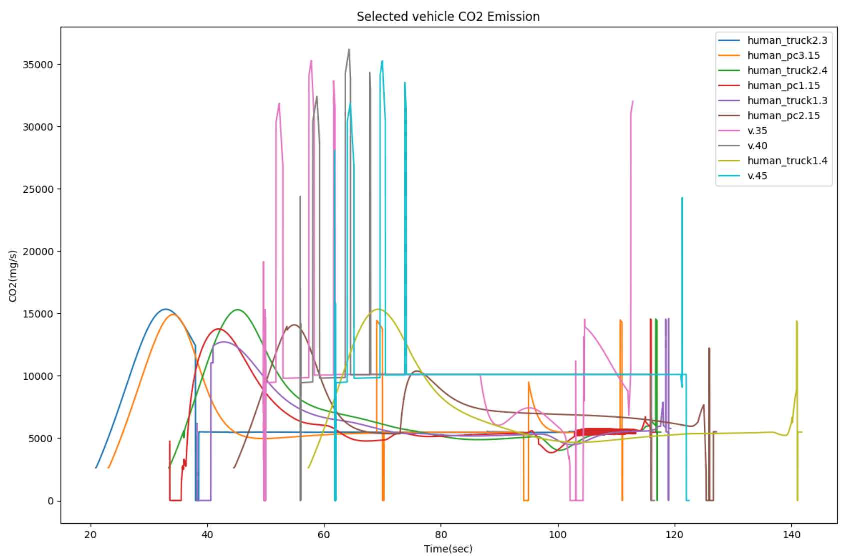

Figure 30. Vehicles' CO2 Emission Curve over Time Under Setting E

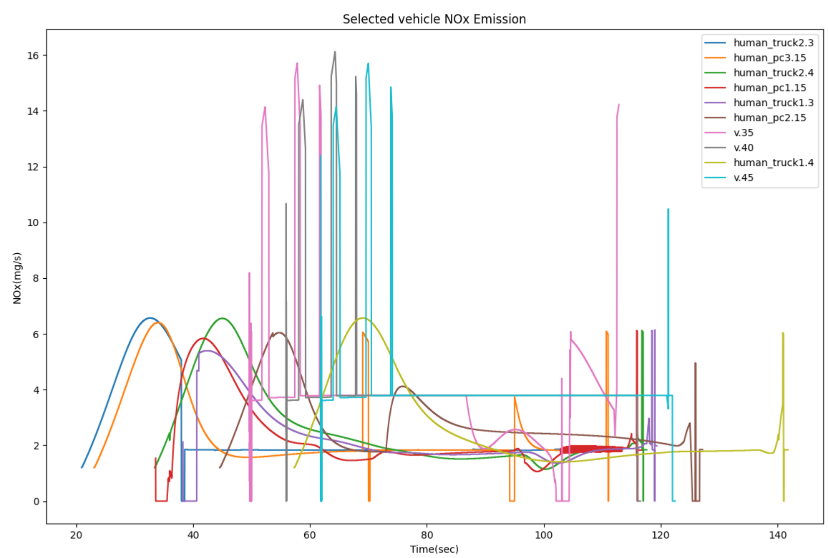

Figure 31. Vehicles' NOx Emission Curve over Time Under Setting E

From the emission perspective, Figures 30 and 31 show the emission trends of several randomly selected vehicles under setting E. The two items that were inspected are the CO2 emission curve and the NOx emission curve. As shown, With identical emission configurations, platoon CAVs' emissions may fluctuate more dramatically than regular vehicles. That could be caused by a shorter spacing strategy. During the platoon merging or splitting, the demand for synchronizing speed made vehicles accelerate/decelerate more intensively. Otherwise, due to operating at a higher speed on dedicated lanes, both the sensitivity towards either speed or gap error and the need for more engine power to maintain a higher speed could lead to higher overall emissions. However, since good proportions of traffic have been separated. The vehicle's on regular lanes could operate less intensively. The overall emissions of regular traffic are relatively reduced compared to the situations when CAV platoons were running with others.

Discussion

Deploying dedicated lane(s) for CAV platoon traffic has the potential to improve the totality of traffic. This provides a stable situation where CAV platoons could attain higher travel speeds. Such infrastructure design could promote the market penetration of platoon-able CAVs. However, since dedicated lanes are commonly tolled roadways in the US highway system, more studies could be appreciated to find out the outcome of the dedicated lanes for CAV platoons from other perspectives rather than the engineering one.

On the other hand, when the percentage of CAV platoon traffic is low, converting lane(s) to dedicated platoon lane(s) could increase the burden of the regular lanes. Thus, dedicated CAV platoon lanes are a conditional improving infrastructure design when the percentage of CAV platoon traffic is steadily increased to a higher level. When considering emission factors, if the powertrain designs are not varied significantly between CAVs and human-driven vehicles, implementing CAV's into traffic may lead to worse results.

Chapter 5

SUMMARY, CONCLUSION AND RECOMMENDATIONS

Summary

In this study, a conceptual multi-platform simulation framework is presented. The framework consists of two major components: a vehicle-level simulation model and an integrated traffic-level simulator. The vehicle-level model provided an in-depth perspective to researchers who are interested in studying CAV platoons. Vehicle dynamics, autonomy, and control theories could be the target topics when conducting this level of simulations. When studying from a higher level, integrated traffic level-simulators could be utilized. However, at this level, communication protocols and traffic pattern designs might be essential topics to be studied.

The vehicle-level model is instantiated on the MATLAB/Simulink platform. The vehicle's longitudinal and lateral dynamic and control theories were designed and implemented to calibrate the vehicle model for the network-level simulations. By feeding the calibration results (critical control parameters) to the traffic-level simulations, the users could interpret the simulations more precisely. Calibrations were conducted for tuning the selected longitudinal control laws – CC, ACC, and CACC. The three laws are used to represent the three components of traffic – human-driven vehicles, AVs, and CAVs. Then, the results of calibrations were injected into the integrated CAV simulators to explore the impacts of CAV platoons on traffic under different scenarios. PLEXE framework was selected to perform traffic simulations. PLEXE is a simulation framework developed upon the integration of SUMO and OMNeT++. The former is one of the most popular graphical microscopic traffic simulators, and the latter is also a well-known communication network simulator in the inter-vehicle communication field. One of the main reasons to use these two platforms, MATLAB/Simulink and PLEXE/SUMO/OMNeT++, is that they allow users to customize simulation models extensively and in-depth.

For vehicle-level simulation, a review of vehicle autonomy and connected driving technologies was conducted. It summarized the technological structure of CAV systems. In this study, the traditional functional structure of the autonomous driving system is selected to build the vehicle model. The three main components of a traditional autonomous driving system are perception, planning and control. Communication systems perform in an inter-component manner and provide information to leverage the performance of all three components of the ADS. Derived from the CAV system and platooning management system, the basic functional structure of a CAV platooning system is completed.

Following the review of the CAV and platooning technologies, modeling and simulation algorithms were reviewed. Vehicle models included the longitudinal dynamic model, lateral dynamic model, three degrees-of-freedom vehicle body model, and string stability model. Longitudinal control laws of lane-changing trajectory methods were also investigated. Then, communication issues such as communication topology and communication network simulator were studied and used in the developed system.

The traffic-level simulation was conducted by utilizing SUMO, the OMNeT++ simulator, and a CAV platooning simulation framework, PLEXE. In SUMO, a selected real-world infrastructure, a section of the US Interstate highway was modeled as a roadway network based on its geometric information gained through the GIS platform. Injection of the traffic without connection to other vehicles was managed by SUMO. From the communication perspective, OMNeT++ simulates the communication actions among vehicles simultaneously. During the simulation, each connected vehicle received a representation in OMNeT++ as a communication unit. The communication protocols, message types, and lower-layer applications were preset in OMNeT++ based upon the IEEE 1609.4p family. PLEXE performed as the platooning management system. This framework managed platooning maneuvers and the platooned vehicle's behaviors.

Two case studies were conducted to explore the potential influences of CAV platoons on the traffic. The studies focused first on how different percentages of CAV platoon traffic affected the traffic based on existing infrastructure in a realistic manner. The other topic addressed how different platoon-oriented infrastructure designs and traffic patterns impacted traffic. A section of a US Interstate highway was selected as the infrastructure to establish the scenarios in. Before studying CAV platoon traffic's impacts, validation was conducted to examine if the simulation models could represent the real-world traffic. The validation was completed by comparing the traffic data (e.g., density, travel time, and volume) from the fieldwork and simulations. The settings of the simulation were based on calibration outputs and included vehicle specifications, vehicle control laws, and corresponding parameters.

The results of the case studies showed that with CAV platoon traffic, the overall performance of the traffic improved. When no changes were deployed to the traffic pattern or infrastructure design, improvements were found in measures such as the travel times under different levels of traffic density. However, traffic with low percentages of CAV platoon or with low overall traffic volume made the improvement of introducing

CAV platoon traffic subtle. After deploying a new traffic pattern and infrastructure design (dedicated separated lane(s) for CAV platoon traffic), the simulation results showed that such patterns and designs have the potential to help with reducing the overall travel time. However, when the percentages of CAV platoon traffic were low, the regular traffic flow could deteriorate. Emission issues were measured in the second case study. The results showed that separating CAV platoon traffic from regular traffic could lessen the emissions on the regular traffic side when CAV traffic reaches a relatively high percentage. But overall, due to implementing higher speed limits on the CAV platoon's dedicated lanes, the emissions of CAVs increased significantly. Otherwise, the sensitivity to speed and spacing error required CAVs to adjust speed more intensively, which eventually led to higher overall emissions.

Conclusion

A conceptual simulation framework for exploring CAV platoons was the main objective of this study. The framework takes both vehicle-level models and traffic-level simulations into account. From a single-vehicle perspective, the technologies that leverage vehicle platooning were investigated. A component-based functional structure of the CAV system is presented in this study. The four components are perception, planning, control, and communication. Such a structure has been studied and widely implemented in the development of autonomous driving systems and vehicle communication protocol, and application designs. To test the theory, a CAV model was developed. The model integrated vehicle dynamic models, car longitudinal control laws, controller actuation delay, perception delay, lane-changing trajectory planning, and lateral control. Then, the model was utilized to calibrate the CAV's longitudinal control algorithms for the traffic-level simulation model.

Based on the reviewed literatures, conducting the traffic level simulation of CAV platoons has been found to be a multi-platform integration task. Such integration of traffic simulators and communication network simulators helps the users to focus on single or multiple objective topics instead of managing every piece of the simulation task. The most challenging part was to integrate those simulators properly. In this study, a SUMO-OMNeT++-based extension/framework was studied. Based upon this, and by modifying the framework and applications, an instance of the CAV platoon simulator has been developed.

By utilizing the developed simulator, two case studies were conducted to investigate two topics related to CAV platooning traffic. The first topic was how CAV platoon traffic could affect existing traffic patterns and infrastructure when considering traffic growth and truck traffic components. The second topic was whether implementing CAV platoon dedicated lanes could improve the traffic flow. The results of the first case study showed that when traffic increased as expected, the CAV platoon could slightly improve the average travel speed (also reflects the reduction of average travel time) when its percentage reached a certain level. Then, the next case study proved that dedicated lanes could improve the performance of traffic with some percentages of CAV traffic. Either adding lanes or converting lanes could benefit the overall traffic. However, deploying dedicated lanes with low percentages of CAV platoon traffic was not suggested since it could increase the burden of the regular lanes in the form of capacity reduction. There is no evidence to assert that separating CAV platoon traffic from the regular traffic with dedicated lanes could mitigate the CO_2 and

NOx emissions when powertrain techniques do not vary between CAVs and human-driven vehicles.

Limitations, Recommendations and Future Work

A simulation framework instance has been developed and implemented for studying two cases based on a realistic situation. However, as a limitation of all modeling and simulation algorithms, in order to achieve a trade-off between performance and cost, proper abstractions are necessary.

As a part of the framework instance, inter-vehicle communication technologies were studied and simplified when developing the vehicle-level model. Inserting delays could not represent all adverse conditions of inter-vehicle communication. Inter-vehicle communication is only one aspect of communication of CAV technologies. Thus, for further study, including more communication factors with details into the framework would be important.

As mentioned in the second chapter of this study, component-based functional architecture is not the only way to design autonomous driving systems. End-to-end technologies, such as learning-based technologies, also have been studied extensively for designing and building autonomous driving systems. On the other hand, instantiating the framework with an integrated usage of certain tools could limit the applications of the framework. For example, researchers have deployed other tools for modeling and simulating CAVs with end-to-end technologies.

As the conclusion draws above, dedicated lanes for CAV platoons could help to mitigate the burden on dedicated lanes as well as improve the traveling speed on the dedicated lanes. However, such methods could not be effective when the proportion of CAV platoon traffic is low.

Moreover, according to the definition of dedicated lanes in the US highway system, the traffic using dedicated lanes may be tolled for usage. It could discourage using the functionality of platooning CAVs and the lanes when the rates of toll are considered high by the users. Thus, the necessity of deploying dedicated lanes is a complex problem that needs to be studied from more perspectives than the engineering one.

The integrations of simulators have been widely accepted when studying CAV platoon problems. Thus, more efforts could be made to improve the integration of different simulators. It could not only provide more options when studying CAV platoons by simulation but also make it possible to avoid the imprecision caused by over-trusting a few integrated simulations.

REFERENCES

[1] National Highway Traffic Safety Administration, "Automated Vehicles for Safety," National Highway Traffic Safety Administration, U.S. DOT, [Online]. Available: https://www.nhtsa.gov/technology-innovation/automated-vehicles. [Accessed 11 2020].

[2] B. Friedrich, "The Effect of Autonomous Vehicles on Traffic," in *Autonomous Driving: Technical, Legal and Social Aspects*, Berlin, Germany, Springer, 2016, pp. 317-334.

[3] A. Eugensson, M. Brannstrom, D. Frasher, M. Rothoff, S. Solyom and A. Robersson, "Environmental, Safety, Legal and Societal Implications of Autonomous Driving Systems," in *23rd International Technical Conference on the Enhanced Safety of Vehicles*, Seoul , Korea, South, 2013.

[4] C. Liu, K. T. Chau, D. Wu and S. Gao, "Opportunities and Challenges of Vehicle-to-Home, Vehicle-to-Vehicle, and Vehicle-to-Grid Technologies," *Proceedings of the IEEE,* vol. 101, no. 11, pp. 2409-0427, 2013.

[5] S. Guler, M. Menedez and L. Meier, "Using connected vehicle technology to improve the efficiency of Intersections," *Transportation Research Part C,* no. 46, pp. 121-131, 2014.

[6] S. Narla, "Narla, Siva R K. “The Evolution of Connected Vehicle Technology: From Smart Drivers to Smart Cars to....Self-Driving Cars," *Ite Journal-institute of Transportation Engineers*, no. 83, 2013.

[7] A. Talebpour and H. S. Mahmassani, "Influence of Connected and Autonomous Vehicles on Traffic Flow Stability and Throughput," *Transportation Research Part C,* vol. 71, pp. 143-146, 2016.

[8] X. Krasniqi and E. Hajrizi, "Use of IoT Technology to Drive the Automotive Industry from Connected to Full Autonomous Vehicles," in *International Federation of Automatic Control*, 2016.

[9] Y. Xie, H. Zhang, N. H. Gartner and T. Arsava, "Collaborative Merging Strategy for Freeway Ramp Operations in a Connected and Autonomous Vehicles Environment," *Journal of Intelligent Transportation Systems,* vol. 21, no. 2, pp. 136-147, 2017.

[10] C. Li, Q. Luo, G. Mao, M. Sheng and J. Li, "Vehicle-mounted base station for connected and autonomous vehicles: Opportunities and challenges," *IEEE Wireless Communications,* vol. 26, no. 4, pp. 30-36, 2019.

[11] J. Rupp and A. King, "Autonomous Driving - A Practical Roadmap," SAE International, 2010.

[12] M. Campbell, M. Egerstedt, J. How and R. Murray, "Autonomous driving in urban environments: approaches, lessons and challenges," *Philosophical Transactions of The Royal Society*, no. 368, pp. 4649-4672, 2010.

[13] U. Montanaro, S. Dixit, S. Fallah, M. Dianati, A. Stevens, D. Oxtoby and A. Mouzakitis, "Toward Connected Autonomous Driving: Review of Use-Cases," *Vehicle System Dynamics International Journal of Vehicle Mechanics and Mobility*, vol. 57, no. 6, pp. 779-814, 2019.

[14] California Partners for Advanced Trasit and Highways, "California PATH Annual Report," University of California, Berkeley, Berkeley, CA, 1996.

[15] California Partners of Advanced Transit and Highways, "California PATH Annual Report 1997," University of California, Berkeley, Berkeley, 1997.

[16] S. D. Pendleton, H. Andersen, X. Du, X. Shen, M. Meghjani, Y. H. Eng, D. Rus and M. H. Ang, "Perception, Planning, Control, and Coordination for Autonomous Vehicles," *Machines*, vol. 5, no. 1, 2017.

[17] National Highway Traffic Safety Administration, "Vehicle-to-Vehicle Communications: Readiness of V2V Technology for Application," National Highway Traffic Safety Administration, U.S. DOT, Washington, DC, 2014.

[18] J. Luo and J.-P. Hubaux, "A Survey of Inter-Vehicle Communication," School of Computer and Communication Sciences, EPFL, Lausanne, Switzerland, 2004.

[19] B. Xu, X. J. Ban, Y. Bian, W. Li, J. Wang, S. E. Li and K. Li, "Cooperative Method of Traffic Signal Optimization and Speed Control of Connected Vehicles at Isolated Intersections," *IEEE Transactions on Intelligent Transportation Systems*, vol. 20, no. 4, pp. 1390-1403, 2019.

[20] Y. Li, "An Overview of the DSRC/WAVE Technology," in *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*, 2010.

[21] ETSI, "Intelligent Transport Systems (ITS); Radiocommunications equipment operating in the 5 855 MHz to 5 925 MHz frequency band; Harmonized EN covering the essential requirements of article 3.2 of the R&TTE Directive," 2013.

[22] P. Lindberg, T. Abbas, U. Zang, A. Laya, M. Fallgren, A. Fernandez, A. Servel, O. Sancier, R. Comte, E. Le Fur, M. Bouillon, M. Gharba, Z. Li and G. Vivier, "Deliverable D2.2 - Intermediate Report on V2X Business Models and Spectrum," European Commission, 2019.

[23] K. Abboud, H. A. Omar and W. Zhuang, "Interworking of DSRC and Cellular Network Technologies for V2X Communications: A Survey," *IEEE Transactions on Vehicular Technology*, vol. 65, no. 12, pp. 9457-9470, 2016.

[24] J. Wang, Y. Shao, Y. Ge and R. Yu, "A Survey of Vehicle to Everything (V2X) Testing," *Sensor*, vol. 19, no. 2, pp. 334-353, 2019.

[25] K. Willett and J. Tomic, "Vehicle-to-Grid Power Fundamentals: Calculating Capacity and Net Revenue," *Journal of Power Sources*, vol. 144, no. 1, pp. 268-279, 2005.

[26] W. Kempton, V. Udo, K. Huber, K. Komara, S. Letendre, S. Baker, D. Brunner and N. Pearre, "A Test of Vehicle-to-Grid (V2G) for Energy Storage and Frequency Regulation in the PJM System," The University of Delaware, Newark, DE, 2008.

[27] C. Liu, K. T. Chau, D. Wu and S. Gao, "Opportunities and Challenges of Vehicle-to-Home, Vehicle-to-Vehicle, and Vehicle-to-Grid Technologies," *roceedings of the IEEE*, vol. 101, no. 10, pp. 2409-2427, 2013.

[28] J. Pillmann, B. Sliwa, J. Schmutzler, C. Ide and C. Wietfeld, "Car-To-Cloud Communication Traffic Analysis Based on the Common Vehicle Information Model," in *IEEE Vehicular Technology Conference*, Sydney, Australia, 2017.

[29] M. Whaiduzzaman, M. Sookhak, A. Gani and R. Buyya, "A Survey on Vehicular Cloud Computing," *Journal of Network and Computer Applications*, vol. 40, pp. 325-344, 2013.

[30] AB Volvo, "Volvo Trucks andFedEx demonstrate Truck Platooning," Volvo Group, 27 6 2018. [Online]. Available: https://www.volvotrucks.us/news-and-stories/press-releases/2018/june/volvo-trucks-and-fedex-successfully-demonstrate-truck-platooning-on-nc-540/.

[31] Team CV, " Daimler Platooning Tech," commercialvehicle.in, 15 2 2018. [Online]. Available: https://commercialvehicle.in/daimler-platooning-tech/.

[32] U.S. Department of Transportation, "USDOT Automated Vehicles Activities," 24 3 2022. [Online]. Available: https://www.transportation.gov/AV. [Accessed 15 11 2022].

[33] European Commission, "Connected and automated mobility," 7 6 2022. [Online]. Available: https://digital-strategy.ec.europa.eu/en/policies/connected-and-automated-mobility. [Accessed 15 11 2022].

[34] Ministry of Economy, Trade and Industry of Japan, "Automated Driving and Mobility Service," 21 09 2022. [Online]. Available: https://www.meti.go.jp/english/policy/mono_info_service/automobile_industry/adms/index.html. [Accessed 15 11 2022].

[35] F. Fan, "Nation Releases First Draft of Rules on Self-driving Vehicles in Public," Chinadaily.com.cn, 10 08 2022. [Online]. Available: https://global.chinadaily.com.cn/a/202208/10/WS62f2e7bda310fd2b29e71420.html. [Accessed 15 11 2022].

[36] U.S. Department of Transportation Federal Highway Administration, "Highway Functional Classification Concepts, Criteria and Procedures," 2013. [Online]. Available:

https://dot.sd.gov/media/documents/HwyFunctionalClassification.pdf. [Accessed 6 10 2022].

[37] P. Ioannou and Z. Xu, "Throttle And Brake Control Systems For Automatic Vehicle Following," UC Berkeley: California Partners for Advanced Transportation Technology, Berkeley, USA, 1994.

[38] M. Park, S. Lee and W. Han, "Development of Steering Control System for Autonomous Vehicle Using Geometry-Based Path Tracking Algorithm," *ETRI Journal*, vol. 37, no. 3, pp. 617-625, 2015.

[39] Society of Automotive Engineers, "Taxonomy and Definitions for Terms Related to Driving Automation Systems for On-Road Motor Vehicles J3016_202104," Society of Automotive Engineers, 30 4 2021. [Online]. Available: https://www.sae.org/standards/content/j3016_202104/.

[40] National Highway Traffic Safety Administration, "Diver Assistance Technologies," National Highway Traffic Safety Administration, [Online]. Available: https://www.nhtsa.gov/equipment/driver-assistance-technologies.

[41] E. Coelengh, A. Eidehall and M. Bengtsson, "Collision Warning with Full Auto Brake and Pedestrian Detection - a practical example of Automatic Emergency Braking," in *13th International IEEE Conference on Intelligent Transportation Systems*, Funchal, Protugal, 2010.

[42] A. Ziebinski, R. Cupek and D. Grzechca, "Review of advanced driver assistance systems (ADAS)," in *AIP Conference Proceedings 1906*, Thessaloniki, Greece, 2017.

[43] C. Badue, R. Guidolini, R. V. Carneiro, P. Azevedo, V. B. Cardoso, A. Forechi, L. Jesus, R. Berriel, T. Paixao, F. Mutz, L. Veronese, T. Oliveira-Santos and A. F. De Souze, "Self-Driving Cars: A Survey," *Expert Systems with Applications,* vol. 165, no. 113816, 2021.

[44] D. J. Yeong, G. Velasco-Hernandez, J. Barry and J. Walsh, "Sensor and Sensor Fusion Technology in Autonomous Vehicles: A Review," *Sensors,* vol. 6, no. 2140, 2021.

[45] F. Rosique, P. J. Navaro, C. Fernandez and A. Padilla, "A Systematic Review of Perception System and Simulators for Autonomous Vehicles Research," *Sensors,* vol. 19, no. 648, 2019.

[46] E. Yurtsever, J. Lanbert, A. Carballo and K. Takeda, "A Survey of Autonomous Driving: Common Practices and Emerging Technologies," *IEEE Access,* vol. 8, pp. 58443-58469, 2020.

[47] E. Marti, M. A. de Miguel, F. Garcia and J. Perez, "A Review of Sensor Technologies for Perception in Automated Driving," *IEEE Intelligent Transportation Systems Magazine*, vol. 11, no. 4, pp. 94-108, 2019.

[48] J. Z. Varghese and R. G. Boone, "Overview of Autonomous Vehicle Sensors and Systems," in *Proceedings of the 2015 International Conference on Operations Excellence and Service Engineering*, Orlando, Florida, 2015.

[49] L. G. Calvao, M. Abbod, T. Kalganova, V. Palade and M. N. Huda, "Pedestrian and Vehicle Detection in Autonomous Vehicle Perception Systems—A Review," *Sensor*, vol. 21, no. 7267, 2021.

[50] S. Patole, M. Torlak, D. Wang and M. Ali, "Automotive radars: A review of signal processing techniques," *IEEE Signal Processing Magazine*, vol. 34, no. 2, pp. 22-35, 2017.

[51] Z. Wang, Y. Wu and Q. Niu, "Multi-Sensor Fusion in Automater Driving: A Survey," *IEEE Access*, vol. 8, pp. 2847-2868, 2020.

[52] J. Kocic, N. Jovicic and V. Drndarevic, "Sensors and Sensor Fusion in Autonomous Vehicles," in *26th Telecommunications Forum (TELFOR)*, Belgrade, Serbia, 2018.

[53] S. Kuutti, S. Fallah, K. Katsaros, M. Dianati, F. Mccullough and A. Mouzakitis, "A Survey of the State-of-the-Art Localization Techniques and Their Potentials for Autonomous Vehicle Applications," *IEEE Internet of Things Journal*, vol. 5, no. 2, pp. 829-846, 2018.

[54] G. A. Z. Y. L. G. S. Bresson, "Simultaneous Localization and Mapping: A Survey of Current Trends in Autonomous Driving," *IEEE Transactions on Intelligent Vehicles*, vol. 2, no. 3, pp. 194-220, 2017.

[55] W. Schwarting, J. Alonso-Maro and D. Rus, "Planning and Decision-Making for Autonomous Vehicles," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 1, pp. 187-210, 2018.

[56] H. Bast, D. Delling, A. V. Goldberg, M. Müller-Hannemann, T. Pajor, P. Sanders, D. Wagner and R. F. Werneck, "Route Planning in Transportation Networks," in *Algorithm Engineering*, 2016.

[57] L. Claussmann, M. Revilloud, D. Gruyer and S. Glaser, "A Review of Motion Planning for Highway Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems*, pp. 1826-1848, 2019.

[58] R. Bellman, "A Markovian Decision Process," *Journal of Mathematics and Mechanics*, vol. 6, no. 5, pp. 679-684, 1957.

[59] D. Gonz´alez, J.´. P´erez, V. Milan´es and F. Nashashibi, " Review of Motion Planning Techniques for Automated Vehicles," *IEEE Transactions on Intelligent Transportation Systems* , vol. 17, pp. 1-11, 2015.

[60] N. H. Amer, H. Zamzuri, K. Hudha and Z. A. Kadir, "Modelling and Control Strategies in Path Tracking Control for Autonomous Ground Vehicles: A Review of State of the Art and Challenges," *Journal of Intelligent & Robotic Systems*, vol. 86, pp. 225-254, 2017.

[61] Y. Xiao, F. COdevilla, A. Gurram, O. Uralioglu and A. M. López, "Multimodal End-to-End Autonomous Driving," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 1, pp. 537-547, 2022.

[62] J. Kim and C. Park, "End-To-End Ego Lane Estimation Based on Sequential Transfer Learning for Self-Driving Cars," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR) Workshops,* Honolulu, HI, USA, 2017.

[63] M. Bojarski, D. D. Testa, D. Dworakowski, B. Firner, B. Flepp, P. Goyal, L. D. Jackel, M. Monfort, M. Monfort, U. Muller, J. Zhang, X. Zhang, J. Zhao and K. Zieba, "End to End Learning for Self-Driving Cars," 25 4 2016. [Online]. Available: https://arxiv.org/abs/1604.07316. [Accessed 6 11 2022].

[64] A. Tampuu, T. Matiisen, M. Semikin, D. Fishman and N. Muhammad, "A Survey of End-to-End Driving: Architectures and Training Methods," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 33, no. 4, pp. 1364-1384, 2020.

[65] S. Al-Sultan, M. Al-Doori, A. Al-Bayatti and H. Zedan, "A comprehensive survey on vehicular Ad Hoc network," *Journal of Network and Computer Applications,* 2013.

[66] D. S. Gaikwad and M. Zaveri, "VANET Routing Protocols and Mobility Models: A Survey," in *WeST 2011, NeCoM 2011, WiMoN 2011*, 2011.

[67] C. Campolo, Vehicular ad hoc networks : standards, solutions, and research, Springer, 2015.

[68] J. Siegel, D. Erb and S. Sarma, "A Survey of the Connected Vehicle Landscape — Architectures, Enabling Technologies, Applications, and Development Areas," *IEEE Transactions on Intelligent Transportation Systems,* vol. 19, no. 8, pp. 2391-2406, 2018.

[69] Z. McHardy, A. Khan, K. Obana and S. Iwashina, "V2X Access Technologies: Regulation, Research, and Remaining Challenges," *IEEE Communications Surveys & Tutorials,* vol. 20, no. 3, pp. 1858-1877, 2018.

[70] SAE International, "V2X Communications Message Set Dictionary J2735_202007," 23 7 2020. [Online]. Available: https://www.sae.org/standards/content/j2735_202007/. [Accessed 2 10 2022].

[71] ETSI European Telecommunications Standards Institute, "ETSI EN 302 637-3 Intelligent Transport Systems (ITS) Vehicular Communications Basic Set of Applications Part 3," 9 2014. [Online]. Available: https://www.etsi.org/deliver/etsi_en/302600_302699/30263703/01.02.01_30/en_30263703v010201v.pdf. [Accessed 2 10 2022].

[72] E. E. T. S. Institut, "ETSI EN 302 637-2 Intelligent Transport Systems (ITS) Vehicular Communications Basic Set of Applications Part 2," 9 2014. [Online]. Available:

https://www.etsi.org/deliver/etsi_ts/102800_102899/10289402/01.02.01_60/ts_10289402v010201p.pdf. [Accessed 2 10 2022].

[73] M. Ren, J. Zhang, L. Khoukhi, H. Labiod and V. Vèque, "A review of clustering algorithms in VANETs," *Annals of Telecommunications,* vol. 76, no. 9, pp. 581-603, 2021.

[74] A. Katiyar, D. Singh and R. S. Yadav, "State-of-the-art approach to clustering protocols in VANET: a survey," *Wireless Networks,* vol. 26, no. 7, pp. 5307-5336, 2020.

[75] C. Cooper, D. Franklin, M. Ros, F. Safaei and M. Abolhasan, "A Comparative Survey of VANET Clustering Techniques," *IEEE Communications Surveys & Tutorials,* vol. 19, no. 1, pp. 657-681, 2016.

[76] A. K. Bhoopalam, N. Agatz and R. Zuidwijk, "Planning of truck platoons: A literature review and directions for future research," *Transportation Research Part B: Methodological,* vol. 107, pp. 212-228, 2018.

[77] V. Viswanathan, V. Shankar and S. Sripad, "Final Research Rep: Platooning for Improved Safety and Efficiency of Semi-trucks [PISES]," U.S. Department of Transportation's University Transportation Centers, Pittsburgh, USA, 2019.

[78] Z. Wadud, D. MacKenzie and P. Leiby, "Help or hindrance? The travel, energy and carbon impacts of highly automated vehicles," *Transportation Research Part A: Policy and Practice,* vol. 86, pp. 1-18, 2016.

[79] A. Vinel, L. Lan and N. Lyamin, "Vehicle-to-Vehicle Communication in C-ACC/Platooning Scenarios," *IEEE Communications Magazine,* vol. 53, no. 8, pp. 192-197, 2015.

[80] H.-S. Tan, R. Rajamani and W.-B. Zhang, "Demonstration of an automated highway platoon system," *Proceedings of the 1998 American Control Conference. ACC (IEEE Cat. No.98CH36207),* vol. 3, pp. 1823-1827, 1998.

[81] D. Jia, K. Lu, J. Wang, X. Zhang and X. Shen, "A Survey on Platoon-Based Vehicular Cyber-Physical Systems," *IEEE Communications Surveys & Tutorials,* vol. 18, no. 1, pp. 263-284, 2016.

[82] S. Shladover, C. Nowakowski, X.-Y. Lu and R. Ferlis, "Cooperative Adaptive Cruise Control: Definitions and Operating Concepts," in *94th TRB Annual Conference*, Washington D.C., US, 2015.

[83] K. C. Dey, L. Yan, X. Wang, Y. Wang, H. Shen, M. Chowdhury, L. Yu, C. Qiu and V. Soundararaj, "A Review of Communication, Driver Characteristics, and Controls Aspects of Cooperative Adaptive Cruise Control (CACC)," *IEEE Transactions on Intelligent Transportation Systems,* vol. 17, no. 2, pp. 491-509, 2016.

[84] S. Malik, M. A. Khan and H. El-Sayed, "Collaborative Autonomous Driving—A Survey of Solution Approaches and Future Challenges," *Sensors (Basel)*, vol. 21, no. 11, 2021.

[85] B. Schünemann, "V2X Simulation Runtime Infrastructure VSimRTI: An Assessment Tool to Design Smart Traffic Management Systems," *Computer Networks*, vol. 55, no. 14, pp. 3189-3198, 2011.

[86] R. Protzmann, K. Schrab, M. Schweppenhäuser and I. Radusch, "Implementation of a Perception Module for Smart Mobility Applications in Eclipse MOSAIC," in *SUMO User Conference Preceedings*, 2022.

[87] M. Segata, F. Dressler, R. Lo Cigno and M. Gerla, "A Simulation Tool for Automated Platooning in Mixed Highway Scenarios," in *Proceedings of the 18th annual international conference on Mobile computing and networking*, New York, NY, 2012.

[88] M. Segata, S. Joerer, B. Bloessl, C. Sommer, F. Dressler and R. L. Cigno, "Plexe: A platooning extension for Veins," *2014 IEEE Vehicular Networking Conference (VNC)*, pp. 53-60, 2014.

[89] L. Zhao and J. Sun, "Simulation Framework for Vehicle Platooning and Car-following Behaviors under Connected-Vehicle Environment," in *Procedia - Social and Behavioral Sciences*, Shenzhen, China, 2013.

[90] D. Jia, K. Lu and J. Wang, "A Disturbance-Adaptive Design for VANET-Enabled Vehicle Platoon," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 2, pp. 527-539, 2014.

[91] J. Larson, K.-Y. Liang and K. H. Johansson, "A Distributed Framework for Coordinated Heavy-Duty Vehicle Platooning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 16, no. 1, pp. 419-429, 2015.

[92] R. Riebl, H.-J. Günther, C. Facchi and L. Wolf, "Artery: Extending Veins for VANET applications," in *2015 International Conference on Models and Technologies for Intelligent Transportation Systems (MT-ITS)*, Budapest, Hungary, 2015.

[93] A. Hegde and A. Festag, "Artery-C: An OMNeT++ Based Discrete Event Simulation Framework for Cellular V2X," in *Proceedings of the 23rd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Alicante, Spain, 2020.

[94] S. Santini, A. Salvi, A. S. Valente, A. Pescape, S. Michele and R. L. Cigno, "A Consensus-based Approach for Platooning with Inter-Vehicular Communications," in *2015 IEEE Conference on Computer Communications (INFOCOM)*, Hong Kong, China, 2015.

[95] S. Santini, A. Salvi, A. S. Valente, A. Pescape, M. Segata and R. L. Cigno, "Platooning Maneuvers in Vehicular Networks: A Distributed and Consensus-

Based Approach," *IEEE Transactions on Intelligent Vehicles*, vol. 4, no. 1, pp. 59-72, 2019.

[96] S. E. Li, Y. Zheng, K. Li and J. Wang, "An Overview of Vehicular Platoon Control under the Four-Component Framework," in *2015 IEEE Intelligent Vehicles Symposium*, Seoul, Korea, 2015.

[97] Q. Deng, "A General Simulation Framework for Modeling and Analysis of Heavy-Duty Vehicle Platooning," *IEEE Transactions on Intelligent Transportation Systems,* vol. 17, no. 10, pp. 3252-3262, 2016.

[98] B. Ribeiro, F. Gonçalves, A. Santos, M. J. Nicolau, B. Dias, J. Macedo and A. Costa, "Simulation and Testing of a Platooning Management Protocol Implementation," *Lecture Notes in Computer Science,* vol. 10372, 2017.

[99] S. Bang and S. Ahn, "Platooning Strategy for Connected and Autonomous Vehicles: Transition from Light Traffic," *Transportation Research Board,* vol. 2623, no. 1, pp. 73-81, 2017.

[100] V. Jain, S. Lapoehn, T. Frankiewicz, T. Hesse, M. Gharba, S. Gangakhedkar, K. Ganesan, H. Cao, A. R. Ali, Y. Zou and L. Gu, "Prediction based framework for vehicle platooning using vehicular communications," in *2017 IEEE Vehicular Networking Conference (VNC)*, Torino, Italy, 2017.

[101] J. Liu, D. Ma, A. Weimerskirch and H. Zhu, "A Functional Co-Design towards Safe and Secure Vehicle Platooning," in *CPSS '17: Proceedings of the 3rd ACM Workshop on Cyber-Physical System Security*, Abu Dhabi, United Arab Emirates, 2017.

[102] H. Ramezani, S. E. Shladover, X. Lu and O. D. Altan, "Micro-Simulation of Truck Platooning with Cooperative Adaptive Cruise Control: Model Development and a Case Study," *Transportation Research Record,* vol. 2672, no. 19, pp. 55-65, 2018.

[103] A. Ibrahim, M. Chetan, D. Goswami, T. Basten and H. Li, "Co-simulation Framework for Control, Communication and Traffic for Vehicle Platoons," in *2018 21st Euromicro Conference on Digital System Design (DSD),* Prague, Czech, 2018.

[104] B. Vieira, R. Severino, A. Koubaa and E. Tovar, "Towards a Realistic Simulation Framework for Vehicular Platooning Applications," in *2019 IEEE 22nd International Symposium on Real-Time Distributed Computing (ISORC)*, Valencia, Spain, 2019.

[105] B. Vieira, R. Severino, E. Filho, A. Koubaa and E. Tovar, "COPADRIVe - A Realistic Simulation Framework for Cooperative Autonomous Driving Applications," in *2019 IEEE International Conference on Connected Vehicles and Expo (ICCVE),* Graz, Austria, 2019.

[106] B. Gerrits, M. Mes and P. Schuur, "Simulation Of Real-Time And Opportunistic Truck Platooning At The Port Of Rotterdam," in *2019 Winter Simulation Conference (WSC)*, Naitonal Harbor, MD, 2019.

[107] G. Sethuraman, X. Liu, F. Bachmann, M. Xie, A. Ongel and F. Busch, "Effects of Bus Platooning in an Urban Environment," in *2019 IEEE Intelligent Transportation Systems Conference (ITSC)*, Auckland, New Zealand, 2019.

[108] S. V. D. Hoef, J. Mårtensson, D. V. Dimarogonas and K. H. Johansson, "A Predictive Framework for Dynamic Heavy-Duty Vehicle Platoon Coordination," *ACM Transactions on Cyber-Physical Systems,* vol. 4, no. 1, pp. 1-25, 2019.

[109] S. Hyun, J. Song, S. Shin and D.-H. Bae, "Statistical Verification Framework for Platooning System of Systems with Uncertainty," in *2019 26th Asia-Pacific Software Engineering Conference (APSEC)*, Putrajaya, Malaysia, 2019.

[110] C. Sommer, D. Eckhoff, A. Brummer, D. S. Buse, F. Hagenauer, S. Joerer and M. Segata, "Veins: The Open Source Vehicular Network Simulation Framework," in *Recent Advances in Network Simulation*, Springer, Cham., 2019, pp. 215-252.

[111] C. Quadri, V. Mancuso, M. A. Marsan and G. P. Rossi, "Platooning on the edge," in *Proceedings of the 23rd International ACM Conference on Modeling, Analysis and Simulation of Wireless and Mobile Systems*, Alicante, Spain, 2020.

[112] M. R. Hidayatullah and J.-C. Juang, "Centralized and Distributed Control Framework Under Homogeneous and Heterogeneous Platoon," *IEEE Access,* vol. 9, pp. 49629-49648, 2021.

[113] F. Miekautsch, F. Seeland, J. Horn and A. Fay, "Situation-aware Switching of the Communication Topology in Heterogeneous Platooning Applications," *IFAC-PapersOnLine,* vol. 54, no. 2, pp. 306-313, 2021.

[114] R. Xu, Y. Guo, X. Han, X. Xia, H. XIang and J. Ma, "OpenCDA: An Open Cooperative Driving Automation Framework Integrated with Co-Simulation," in *2021 IEEE International Intelligent Transportation Systems Conference (ITSC)*, Indianapolis, United States, 2021.

[115] H. Teper, A. Bayuwindra, R. Riebl, R. Severino, J.-J. Chan and K.-H. Chen, "AuNa: Modularly Integrated Simulation Framework for Cooperative Autonomous Navigation," 7 2022. [Online]. Available: https://www.researchgate.net/publication/361949965_AuNa_Modularly_Integrated_Simulation_Framework_for_Cooperative_Autonomous_Navigation. [Accessed 12 10 2022].

[116] C. Liu, Y. Luo, P. Li, C. Xing and W. Kong, "An Extendable Maneuver Management Framework with Fault-Tolerant Mechanism for Vehicle Platoon Control System in Highway Scenario," 4 7 2022. [Online]. Available: https://arxiv.org/abs/2207.01167. [Accessed 18 10 2022].

[117] C. Claramunt, E. Peytchev and A. Bargiela, "A real-time GIS for the analysis of a traffic system," in *1999 IEEE Africon. 5th Africon Conference in Africa (Cat. No.99CH36342)*, 1999.

[118] Y. Fu, Z. Li, K. Song, Z. Qiu and X. Ma, "Integrated Traffic Management Platform Design Based on GIS-T," in *2006 6th International Conference on ITS Telecommunications*, 2006.

[119] U.S. Department of Transportation, "GIS At DOT," U.S. Department of Transportation, [Online]. Available: https://www.transportation.gov/gis. [Accessed 15 11 2022].

[120] R. Rajamani, Vehicle Dynamics and Control, New York, NY, USA: Springer, 2012.

[121] M. C. Kang, S.-H. Lee and C. C. Chung, "Comparative Evaluation of Dynamic and Kinematic Vehicle Models," in *53rd IEEE Conference on Decision and Control*, Los Angeles, CA, 2014.

[122] J. Kong, M. Pfeiffer, G. Schildbach and F. Borrelli, "Kinematic and Dynamic Vehicle Models for Autonomous Driving Control Design," in *2015 IEEE Intelligent Vehicles Symposium (IV)*, Seoul, Korea, 2015.

[123] X. Fan, B. Fan and H. Wang, "Fuzzy control for vehicle stability," in *2nd International Conference on Artificial Intelligence, Management Science and Electronic Commerce (AIMSEC)*, Dengleng, China, 2011.

[124] T. Ba, X. Guan and J. Zhang, "Vehicle lateral dynamics modelling by subspace identification methods and tyre cornering stiffness estimation," *International Journal of Vehicle Systems Modelling and Testing*, vol. 10, no. 4, pp. 340-355, 2015.

[125] E. Kim, J. Kim and M. Sunwoo, "Model predictive control strategy for smooth path tracking of autonomous vehicles with steering actuator dynamics," *International Journal of Automotive Technology*, vol. 15, pp. 1155-1164, 2014.

[126] B. Wu, T. Lee, H. Chang, J. Jiang, C. Lien, T. Lian and J. Perng, "GPS Navigation Based Autonomous Driving System Design for Intelligent Vehicles," in *2007 IEEE International Conference on Systems, Man and Cybernetics*,, Montreal, Que., 2007.

[127] T. Ort, L. Paull and D. Rus, "Autonomous Vehicle Navigation in Rural Environments Without Detailed Prior Maps," in *2018 IEEE International Conference on Robotics and Automation (ICRA)*, Brisbane, QLD, 2018.

[128] R. Matthaei, G. Bagschik and M. Maurer, "Map-relative Localization in Lane-Level Maps for ADAS and Autonomous Driving," in *IEEE Intelligent Vehicles Symposium*, Dearborn, Michigan, USA, 2014.

[129] D. Swaroop and J. K. Hedrick, "String Stability of Interconnected Systems," *IEEE Transportation on Automatic Control,* vol. 41, no. 3, pp. 349-357, 1996.

[130] D. Swaroop and J. K. Hedrick, "Constant Spacing Sttrategies for Platooning in Automated Highway Systems," *Journal of Dynamic Systems, Measurement and Control,* vol. 121, no. 3, pp. 462-470, 1999.

[131] B. Besselink and K. H. Johansson, "String Stability and a Delay-Based Spacing Policy for Vehicle Platoons Subject to Disturbances," *IEEE Trasportations on Automatic Control,* vol. 62, no. 9, pp. 4376-4391, 2017.

[132] P. Ioannou and C. Chien, "Autonomous intelligent cruise control," *IEEE Transactions on Vehicular Technology,* vol. 42, no. 4, pp. 657-672, 1993.

[133] P. Kavathekar and Y. Chen, "Vehicle Platooning: A Brief Survey and Categorization," in *International Design Engineering Technical Conferences & Computers and Information in Engineering Conference*, Washington D.C., USA, 2011.

[134] M. Ratiu and M. A. Prichici, "Industrial robot trajectory optimization- a review," *MATEC Web of Conferences,* vol. 126, no. 5, p. 02005, 2017.

[135] T. Flash and N. Hogan, "The Coordination of Arm Movements: An Experimentally Confirmed Mathematical Model," *The Journal of Neuroscienc*, vol. 5, no. 7, pp. 1688-1703, 1985.

[136] K. Kyriakopoulos and G. N. Saridis, "Minimum jerk path generation," in *Proceedings. 1988 IEEE International Conference on Robotics and Automation*, Philadelphia, USA, 1988.

[137] A. Varga and R. Hornig, "An Overview of the OMNeT++ Simulation Environment," in *1st International ICST Conference on Simulation Tools and Techniques for Communications, Networks and Systems*, Marseille, France, 2010.

[138] omnetpp.org, "OMNeT++ - Simulation Manual," [Online]. Available: https://doc.omnetpp.org/omnetpp/manual/. [Accessed 7 11 2022].

[139] J. N. a. S. T. Powers, N. Urquhart, T. Farrenkopf and M. Guckert, "An overview of agent-based traffic simulators," *Transportation Research Interdisciplinary Perspectives,* vol. 12, 2021.

[140] G. Kotusevski and K. Hawick, "A Review of Traffic Simulation Software," *Research Letters in the Information and Mathematical Sciences,* vol. 13, 2009.

[141] P. M. Ejercito, K. G. E. Nebrija, R. P. Feria and L. L. Lara-Figueroa, "Traffic simulation software review," in *2017 8th International Conference on*

Information, Intelligence, Systems & Applications (IISA), Larnaca, Cyprus, 2017.

[142] Eclipse SUMO, "TraCI - SUMO Documentation," 14 2 2022. [Online]. Available: https://sumo.dlr.de/docs/TraCI.html. [Accessed 7 11 2022].

[143] C. a. X. Z. Wu, Y. Liu, C. Fu, K. Li and M. Hu, "Spacing Policies for Adaptive Cruise Control: A Survey," *IEEE Access,* vol. 8, pp. 50149-50162, 2020.

[144] T. Toleedo and D. Zohar, "Modeling Duration of Lane Changes," *Transportation Research Record,* vol. 1, pp. 71-78, 1999.

[145] R. Qian, X. Lai and X. Li, "3D Object Detection for Autonomous Driving: A Survey," *Computer Vision and Pattern Recognition*, 2022.

[146] E. Arnold, O. Y. Al-Jarrah, M. Dianati, S. Fallah, D. Oxtoby and A. Mouzakitis, "A Survey on 3D Object Detection Methods for Autonomous Driving Applications," *IEEE Transactions on Intelligent Transportation Systems,* vol. 20, no. 10, pp. 3784-3795, 2019.

[147] A. Jafari, S. Al-Khayatt and A. Dogman, "Performance Evaluation of IEEE 802.11p for Vehicular Communication Networks," in *8th IEEE, IET Int. Symposium on Communication Systems, Networks and Digital Signal Processing*, Poznan, Poland, 2012.

[148] M. I. Ashraf, C.-F. Liu, M. Bennis and W. Saad, "Towards Low-Latency and Ultra-Reliable Vehicle-to-Vehicle Communication," in *Conference: European Conference on Networks and Communications*, Oulu, Finland, 2017.

[149] S. Zeadally, M. A. Javed and E. B. Hamida, "Vehicular Communications for ITS: Standardization and Challenges," *IEEE Communications Standards Magazine*, vol. 4, no. 1, pp. 11-17, 2020.

[150] Delaware Department of Transportation, "Vehicle Volume Summary (Traffic Counts)," 2021. [Online]. Available: https://deldot.gov/Publications/manuals/traffic_counts/index.shtml. [Accessed 10 11 2022].

[151] The American Association of State Highway and Transportation Officials, A Policy on Geometric Design of Highways and Streets (The Green Book), 7th Edition, 2018.

Appendix A

DATA COLLECTION FIELD WORK

For model validation, the real traffic data was collected for conducting statistical t-test analyses. The topic of collected data is the time headway of pairs of vehicles in traffic consecutively. The pairs of vehicles are selected randomly. A vehicle and its following vehicle are considered as a pair of vehicles. The observations were conducted by counting how long a vehicle to pass the same point that its preceding vehicle just passed.

Three observations were taken on August 10, 2022, 2022, September 23, 2022, and September 30, 2022. Each observation lasted 15 min and obtain 25 samples. The following table summarized when the observations were conducted, and the numbers of sample were recorded.

Table A.1. Data collection observation information

GROUP	DATE	TIME	DURATIO N	NUMBER OF SAMPLES
A	8/10/2022	4:50 – 5:05 pm	15 min	25
B	9/23/2022	5:00 – 5:15 pm	15 min	25
C	9/30/2022	6:07 – 6:22 pm	15 min	25

Table A.2. Statistical Summary of Collected Data

Group	A	B	C
Mean	3.84	2.84	2.34
Median	3.68	2.89	2.88
Maximum	6.10	7.10	5.66
Minimum	1.01	0.82	0.80
Variance	0.89	1.21	0.74
Standard Deviation	0.78	1.08	0.76