Quasi-Cooperative Adaptive Cruise Control: Design and Validation

---

A Dissertation

Presented to

the faculty of the School of Engineering and Applied Science

University of Virginia

---

in partial fulfillment
of the requirements for the degree

Doctor of Philosophy

by

Zheng Chen

August 2020

APPROVAL SHEET

This Dissertation
is submitted in partial fulfillment of the requirements
for the degree of
Doctor of Philosophy

Author Signature: ______________________

This Dissertation has been read and approved by the examining committee:

Advisor: B. Brian Park

Committee Member: Brian L. Smith

Committee Member: Cody H. Fleming

Committee Member: Gang Tao

Committee Member: Michael D. Fontaine

Committee Member: ______________________

Accepted for the School of Engineering and Applied Science:

Craig H. Benson, School of Engineering and Applied Science

August 2020

Acknowledgements

First of all, I would like to express the sincerest gratitude to my advisor Prof. B. Brain Park for his continuous support through my Ph.D. study. He always patiently guided me with profound knowledge and great foresight. I appreciate his motivating questions and invaluable suggestions which drove my steps forward in completing this dissertation. It is a fortune in my life to have studied and worked under the supervision of Prof. Park.

I would like to thank the rest of committee members, Prof. Brian L. Smith, Prof. Cody H. Fleming, Prof. Gang Tao and Prof. Michael D. Fontaine for their insightful criticisms and encouragements which broadened the scope of my mind and kept this research in a right track.

I would like to give a big thank you to our collaborators, Daegyu Lee and Seungwook Lee in Korea Advanced Institute of Science and Technology (KAIST), for their excellent and hard work in the field validation with real vehicles. I will never forget their precious support.

I'm grateful to my colleagues and former colleagues in the Traffic Operation Lab, Seunghan Ryu, Dr. Lian Cui, Dr. Bingrong Sun, and Dr. Seongah Hong, as well as all my friends for their helps and cares in both my living and research activities. I will be missing the fun time we spent together.

Lastly, I would like to say thinks to my parents and family members for the irreplaceable supports and encouragements throughout the Ph.D. study and my whole life. I want to especially thank my wife, Qin Qin, for her selfless dedication and sweet company, for which I never felt alone in the past three years.

Abstract

The emergence of Connected Automated Vehicle (CAV) has enabled a variety of Cooperative Automated Driving applications. Cooperative Adaptive Cruise Control (CACC), as the prevailing longitudinal control method for CAV allowing automated vehicle platooning, is claimed to bring enormous improvements in transportation efficiency and safety. However, such benefits of CACC cannot be easily unleashed in mixed traffic where CAVs are interacting with human-driven non-CAVs.

The existing CACC cannot work effectively in mixed traffic environment due to two limitations. Firstly, when CACC vehicles follow a human-driven and/or unconnected vehicle, they fall back to Adaptive Cruise Control (ACC), which requires much longer headway and deteriorates the traffic stability. Secondly, CACC is unable to benefit connected human-driven vehicles (CHVs) which will also largely appear in the near future. The goal of this research is to address these critical limitations of CACC, by developing quasi-CACC applications that can fully utilize the benefits of vehicular connectivity and take effects in the near future. These applications are referred as “quasi--CACC” because they aim at achieving the CACC-like behaviors while the equipped vehicles do not fully meet the operating requirements, especially in mixed traffic environment.

To address the issue of unconnected vehicle in the traffic, a CACC algorithm with unconnected vehicle in the loop (CACCu) is proposed. Unlike the traditional CACC that requires a connected preceding vehicle, CACCu aims to closely follow an unconnected preceding vehicle utilizing the information from the further (connected) preceding vehicle. CACCu is designed to maintain string stability given various behaviors of

unconnected vehicles, without requiring identification process or extra information on the unconnected vehicles. A linear time-invariant CACCu on top of feedback-feedforward control structure of typical CACC is first designed. It is analytically proven that by attaching a filter of "virtual preceding vehicle" to the original CACC feedforward filter, the CACCu vehicle can stay string-stable at a gap significantly shorter than that required by ACC. The proposed CACCu along with ACC and Connected Cruise Control (CCC) were evaluated in high-fidelity simulations using real vehicle trajectory data from Next Generation Simulation (NGSIM) program and a physics-based vehicle dynamics model from PreScan. Results showed that CACCu avoided most of speed overshootings happening to ACC and CCC, indicating improved string stability. CACCu also led to overall 6~9% acceleration reduction, 35~49% spacing error reduction and 3~7% fuel saving from ACC. Compared with CCC, CACCu achieved 5~8% acceleration reduction, 26~38% spacing error reduction and 2~3% fuel saving. These numbers indicated benefits of CACCu in safety, ride comfort and energy efficiency. Then, an Adaptive Model Predictive Control (A-MPC) approach is proposed to optimize the performance of CACCu. This method make use of both *a priori* knowledge on the human driver's behaviors, and the real-time observation of the actual traffic situation. The simulation results indicated that this A-MPC CACCu can facilitate a more robust implementation.

Moreover, the favorable behaviors of CACCu was validated in the field with real vehicles. CACCu reduced 10.82% acceleration RMS, 60.79% spacing error RMS and 6.24% fuel consumption from ACC's. Compared with human driving, CACCu reduced 17.64% acceleration and 13.43% fuel consumption. The speed profiles showed that CACCu greatly attenuated the traffic disturbances while ACC and human driving tended to amplify them. It was confirmed that CACCu can greatly attenuate the traffic disturbance and improve safety, comfort, and fuel efficiency.

On the other hand, a human-in-the-loop CACC algorithm (hCACC) is developed for human-driven connected vehicle. In hCACC, the human driver remains engaged in the longitudinal control of the vehicle, while hCACC controller applies additional acceleration/deceleration on top of human actions according to the received status of preceding vehicle. By allowing coexistence of the automatic control and driver's actions in a beneficial way, hCACC helps the human driver stabilize the vehicle more efficiently and safely. The proposed hCACC inherits the feedback-feedforward control structure and velocity-dependent spacing policy from typical CACC. String stability analysis shows that hCACC can offer broader string-stable ranges of human parameters than human driving alone or the existing human-in-the loop Connected Cruise Control (CCC), indicating a better capability to mitigate traffic disturbance with the uncertain human behaviors. The desirable properties of hCACC were validated in driving simulator experiments, which showed that hCACC could reduce 36.8% acceleration, 31.2% time-gap fluctuation, 81.2% exposure time to unsafe driving situations, and 15.8% fuel consumption from those of human driving alone.

Contents

Chapter 1. Introduction

1.1 Background

Improving safety and efficiency are the two major goals for the future transportation systems. In 2018, over 1.35 million people worldwide died from traffic crashes, which have become now the leading cause of deaths of people aged 5-29 years [1], and congestion makes the average American commuter yearly waste nearly 7 full working days, which translated to over $1,000 in personal costs [2].

As most of roadway crashes are associated with drivers' improper behaviors [3], Automated Vehicle (AV), featured by sensor technologies, has been considered the most effective way to prevent crashes. Society for Automotive Engineers (SAE) defined five different levels of vehicular automation [4], from single-function driver assistance (level 1) to fully automated driving (level 5). The mainstream manufacturers tended to develop their automated vehicle progressively. In the past decades, a variety of Advanced Driver Assistance Systems (ADAS) have been commercialized and successfully implemented in the fleet [5], [6]. These ADASs consist of Adaptive Cruise Control (ACC), Lane Keeping Assist (LKA), automatic lane change, automatic parking, etc. By enabling single or integrating multiple ADAS applications, Leve-1 to Level-3 vehicular automations have been be achieved. On the other hand, technology companies (e.g., Waymo) defied the progressive roadmap and focus on fully automated (Level-5) vehicle. These self-driving vehicles have been extensively tested in certain areas of America and show lower crash rates per million miles than human drivers do (2.19 vs 6.06) [7]. However, the massive implementation of them in near future is unlikely due to the high cost of the required computing capability and sensors, especially the long-range Lidar [8].

Another cost-effective approach to enhancing safety and efficiency of transportation is vehicular connectivity. Equipped with wireless communication device, an AV become a Connected and Automated Vehicle (CAV). The major advantage of CAV is that the Vehicle-to-Vehicle (V2V) and Vehicle-to-Infrastructure (V2I) communications enable the equipped vehicles to drive not only automatically but also cooperatively, bringing additional benefits than traditional Automated Vehicles (AV) can do. To promote vehicles' connectivity, the governments in U.S., European Union, Asian countries and Australia have respectively depicted the short/long-term plans for vehicular connectivity [9]. The international organizations have been seeking standardization of vehicular communication. In 2010, Institute of Electrical and Electronics Engineers (IEEE) formally approved the IEEE 802.11p standard, which resembles European ITS-G5 and the U.S. DSRC [10], to enable wireless access in vehicular environments (WAVE). Recently, the 3rd Generation Partnership Project (3GPP) has released standards of Cellular-V2X (C-V2X) [11] and the Fifth-generation wireless technology (5G) [12], which have been two strong candidates for vehicular communication worldwide.

CAV is drawing increasing research attention and thus a plenty of CAV applications have been developed. Aided by V2I communication, speed harmonization algorithm for CAV [13] was proposed to optimize speed of CAVs before they enter a speed reduction zone. This helps prevent traffic break-downs and keeping bottlenecks operating at constant traffic feeds. As a result, remarkable reductions in fuel consumption and travel time can be reduced. CAV can also play an important role in intersections. Cooperative Vehicle Intersection Control (CVIC) [14], which aims to cancel the need for signal in intersections, can coordinate maneuvers of CAVs from all the approaches so that they can safely cross the intersection without colliding with other vehicles. Compared with conventional actuated intersection control, CVIC can reduce 99% stop delay, 33% of total travel time and 44% fuel consumption. The same ideal can be applied to ramp merging problem. Coordinated Ramp Control (CRC) [15] was proposed to avoid conflicting trajectories of vehicles on ramp and mainstream while minimizing these vehicles' speed fluctuations. It is noted that CRC can be operated either in centralized form with V2I communication or decentralized form with V2V communication [15].

The most frequent use of V2V communication is in Cooperative Adaptive Cruise Control (CACC) or platooning [16]. The instant situation awareness supported by V2V communication allows the CAVs to preview the traffic situation, and thus precisely maintain a safe inter-vehicle time gap or distance gap. In addition to the longitudinal motion control, cooperative lane change/merging have been developed [17]. By jointly controlling both longitudinal and lateral motions of vehicles in adjacent lanes, the lane change or merge-in can be accomplished safely and smoothly. These vehicle-level CAV applications have been extensively tested and validated in the field [18], [19].

Nevertheless, the promised benefits of CAV may not be easily unleashed in the near future when CAVs are mixed with large numbers of non-CAVs in the traffic. Although governments and manufacturers worldwide are making effort to promote vehicular connectivity [9], CAVs are not likely to gain a dominant Market Penetration Rate (MPR) in the next two decades [20]. It was also predicted that the future growth of AV would be slower than that of Connected Vehicle (CV) due to the relatively low cost and possible mandate of connectivity [20]. NHTSA has been pursuing a mandate of connectivity on light vehicles since 2016 [21], and major manufacturers (such as Toyota and Ford) are planning broad deployments of V2V communication anyway starting around 2022. An optimistic prediction [20] is that the MPR of Connected Vehicles (CV) in the U.S. would grow up to 98% in 2030. However, this does not necessarily mean that CAVs would dominate the market. It is likely that only 25%~60% vehicles would be automated vehicles with ACC in 2030 [20]. This leads to a more complicated situation where a large portion of CVs would be still driven by human. Unlocking the potential of vehicular connectivity on both CAVs and Connected Human-driven Vehicles (CHV) in the mixed traffic is a fascinating but challenging task.

1.2 Motivations

Among all the CAV applications, Cooperative Adaptive Cruise Control (CACC), developed from Adaptive Cruise Control (ACC), has become one of the most promising and ready-to-go longitudinal control strategies. CACC utilizes the acceleration/ intended acceleration sent from preceding vehicle(s) to quickly respond to the speed perturbation from downstream. This feature allows the CACC vehicle to stably travel in short time headways (e.g., 0.6s [22]) that could not be achieved easily by ACC or human drivers. CACC only requires level-1 vehicular automation but previous studies have revealed its great potentials, including doubling the roadway throughput [23], [24] and considerable reduction in fuel consumption and greenhouse gas emissions of vehicles [25]–[27]. Unfortunately, like many CAV applications, CACC's usability is inevitably restrained by the low adoption rate of CAVs in the early stage of implementation [23]–[27]. While there have been a variety of CACC systems [28]–[32] developed and tested, they share the same limitations in the mixed traffic:

1) First, CACC vehicle requires the nearest preceding vehicle to be a CAV or at least CHV that can transmit its status, otherwise CACC falls back to ACC with a much longer headway;
2) Second, the CACC vehicle must be both connected and automated. Although CHVs can facilitate CACC operations by spreading information to CAV followers, CHVs gain no benefit from it. As pointed out by [33], CHVs would have neither a shorter gap to deter cut-ins nor a fuel saving as the CAV followers do.

Although a few previous works have attempted to address CACC from its first dilemma, but they tend to introduce new problems that limit their implementations. For example, graceful degradation of CACC (dCACC) [34] proposed to estimate preceding vehicle's acceleration with radar sensor when the communication is available, but it turned out that the ride comfort would be sacrificed due to the remarkable estimation noise. Connected Cruise Control (CCC) explored the benefits of communication with out-line-of-sight preceding vehicles when the closest one is unconnected, but it required additional information on behavior pattern of the unconnected vehicle, which is not easy to obtain in real traffic.

Therefore, CACC, as well as the existing complementary methods of CACC, are subject to their limited usability or feasibility in the mixed traffic environment. To sufficiently take full advantage of vehicular connectivity for the vehicle's longitudinal control, more advanced CAV/CHV applications dedicated for mixed traffic are in need.

1.3 Research goal and contributions

The goal of this research is to adequately address the existing CACC's limitations and fully harvest the benefits of vehicular connectivity, by developing two quasi-CACC applications, namely, CACC with unconnected vehicle (CACCu) and human-in-the-loop

CACC (hCACC). Fig. 1.1 illustrates how CACCu and hCACC would improve the traffic situation and throughput by extending the use of connectivity. CACCu is designed for CAV to closely follow an unconnected preceding vehicle, while hCACC is designed to assist human driver in CHV for safer and smoother response, using the received information of preceding. These applications are referred as "quasi-CACC" because they aim at partly achieving the favorable properties of CACC when the equipped vehicle does not meet the operational requirements of CACC i.e., having connected nearest preceding vehicle and being automatically driven.

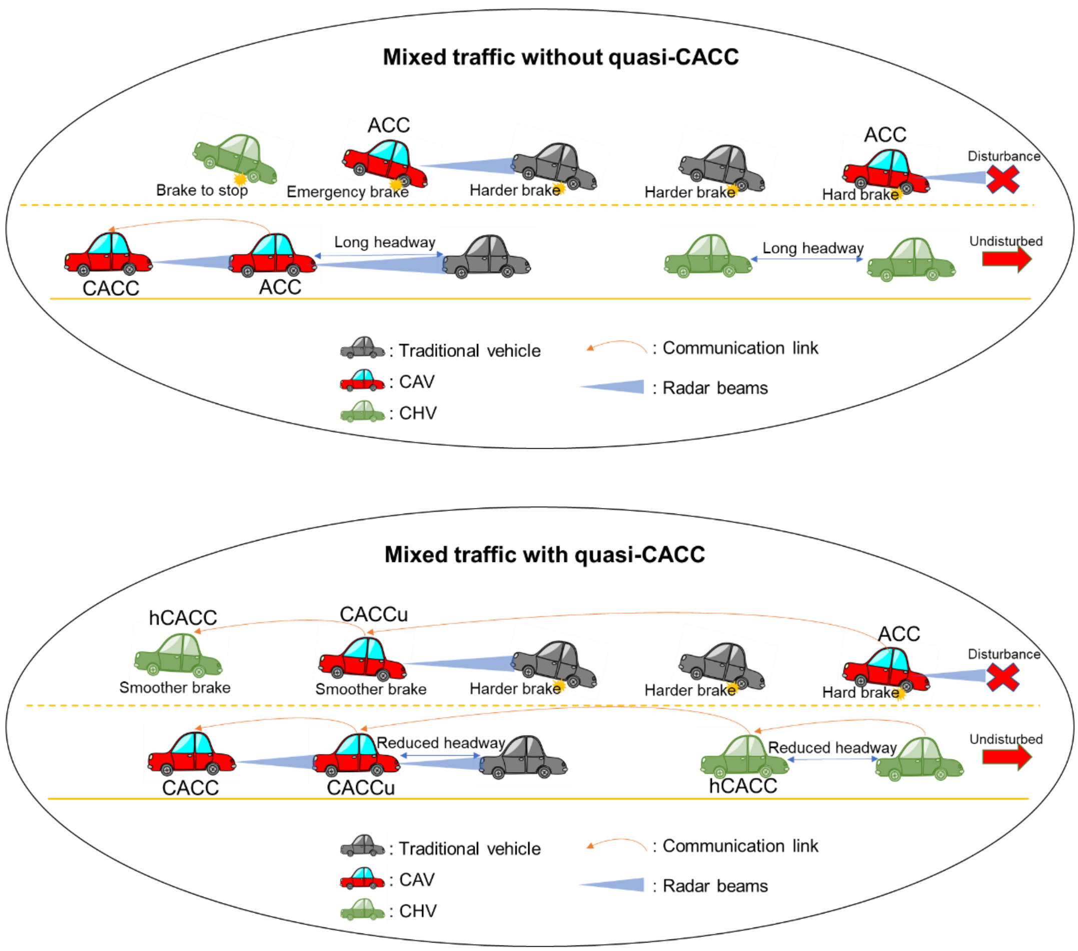


Fig. 1.1 Mixed traffic with and without quasi-CACC

There are three main contributions of this study:

- The control design and analytical validation of CACCu, which aims to enable CAV to stably follow an unconnected vehicle in short spacing instead of falling back to ACC;
- The control design and analytical validation of hCACC, which can improve the performance of CHV by copiloting with the human driver in beneficial ways;
- Validations of the favorable performance (i.e., in string stability, safety, fuel efficiency, etc.) of quasi-CACC applications through high-fidelity simulations and field tests with real vehicles /human drivers.

Chapter 2. Literature Review

This chapter reviews the existing research efforts in the cooperative automated car-following control of vehicles. Especially, the recent progresses of complementing CACC under the imperfect market penetration are covered in detail.

2.1 ACC/CACC and string stability

As one of the basic automated driving applications, Adaptive Cruise Control (ACC) has been commercialized and equipped in a portion of new cars [5]. Using onboard sensors (e.g., radar or lidar), ACC can automatically regulate the vehicle's longitudinal motion to maintain a safe spacing from the preceding vehicle, thus the labor intensity of drivers can be greatly reduced.

The most common ACC controller [35] adopts constant feedback gains in terms of inter-vehicle spacing error (the difference between actual spacing and desired spacing), and relative speed to the preceding vehicle. Some studies also utilize Proportional-Derivative (PD) controller [30], [36], which regulates the spacing error and spacing error rate, for a better internal stability and damping effect on traffic disturbance. There were two popular spacing policy in car-following control: Constant-Distance-Gap (CDG) policy and Constant-Time-Gap (CTG) policy but most of the state-of-art ACC/CACC systems have turned to CTG. This is because not only it better complies with human's expectation (i.e., the higher vehicle speed, the larger spacing), but also it makes "string stability" possible [35]. String stability refers to the property to attenuate the traffic disturbance from downstream [30], [37]. For a linear time-invariant system, string stability can be conveniently checked based on the system's transfer function in frequency domain [30], as the energies of a disturbance in time domain and in frequency are the same [38]. It was proven that ACC with CTG policy can guarantee string stability as either the feedback gains or desired time gap are sufficiently large [39].

In real life, however, the feedback gains are constrained by the sensor noise and ride comfort requirement, thus ACC may need a time gap significantly longer than usual (e.g., >2.5s [30]) for the guaranteed string stability. Considering that production ACC systems are typically operated with desired time gap < 2s, string stability can hardly be fulfilled [40]. As ACC vehicles tend to be string-unstable, the speed/spacing fluctuations from the preceding vehicle would be amplified, causing shockwaves along the upstream. This could in turn lead to unnecessary fuel consumption, traffic congestion or even crash. An example of sting instability was demonstrated in [40], where seven commercial ACC vehicles were tested to travel as a platoon in headway of 1.5s. It was shown that an initial speed disturbance of 6 mph to the platoon was amplified to a 25 mph disturbance, and the last vehicle in the platoon was even observed to disengage ACC. Open-road test in [41] also indicated that there shall not be more than three or four ACC vehicles travelling successively due to string instability.

On the bright side, ACC can be upgraded to CACC with better string stability by enabling V2V communications. Using these communications, a CACC vehicle can almost immediately obtain the preceding vehicle’s acceleration or intended acceleration, and then utilized it as a feedforward signal to help quickly respond to the speed perturbation from downstream. [22], [28]–[32]. As a result, CACC vehicles are able to safely travel in a headway as short as 0.6s with guaranteed string stability, which is difficult to achieve for human driver or ACC. Such short headway can directly increase the roadway throughput, and the smooth behaviors of CACC vehicle can further improve the safety and quality of traffic flow [42]. Due to the favorable performances brought by string stability, it has become a primary design goal of CAV's longitudinal control [32].

In the past decade, CACC systems with different architectures and control methods have been proposed [43]. Based on communication topology, CACC can be briefly divided into three categories: predecessor-following (PF) CACC [30] which only communicates with the nearest preceding vehicle, predecessor-leader-following (PLF) CACC [22] which communicates with both the nearest preceding vehicle and platoon leader, and all-predecessor-following (APF) CACC [44] which requires communications with all the preceding vehicles.

In terms of control method, the rule-based linear control has been frequently adopted in CACC demonstrations. A widely-accepted design of PF CACC was presented in [22], featured by feedback-feedforward control structure. PD controller was adopted as the feedback controller to regulate the states regarding to the immediately preceding vehicle. In addition, the acceleration of preceding vehicle (obtained via communication) served as feedforward signal. The purpose of feedforward control was to eliminate the spacing error, i.e., to apply zero-spacing-error policy. To this end, the acceleration of preceding vehicle was not directly used as the feedforward command, because applying the same acceleration with preceding vehicle only leads to zero relative speed. Instead, a feedforward filter was design based on the spacing policy and the identified vehicle dynamics model. It was shown that the filtered feedforward signal enables the CACC to drive at any desired time gap with guaranteed string stability, when communication delay and actuator delay of the vehicle are insignificant. Even in the realistic conditions, the

CACC can shorten the desired time gap to one third that required by ACC. Such CACC design is especially suitable for ad-hoc platooning with heterogeneous vehicles, as it does not depend on any information of other vehicles' dynamics or control methods. Another influential form of CACC is the one developed by California PATH program [45]. This CACC adopted PLF communication topology and assume homogeneous vehicle dynamics in the platoon. PD feedback controller was used to regulate the spacing and speed relative to both preceding vehicle and the platoon leader (which shared its GPS location). The control command of preceding vehicle was taken as feedforward command. As there was no feedforward filter, zero-spacing-error could not be achieved, but the string stability could still be easily maintained given the swift response to downstream disturbance.

In recent years, there are increasing number of CACC designs based on optimization-based control methods. The performance of such methods usually relies on an accurate system model. Linear Quadratic Regulator (LQR) is a widely-used optimal controller in longitudinal vehicle control [43], [46]. LQR determines the optimal control law by solving an unconstrained optimization problem offline. The infinite-horizon objective function can include multiple system states and/or control inputs in quadratic form. For example, the APF CACC [46] considered the spacing error regarding to the nearest preceding vehicle, relative speed regarding to all the preceding vehicle, and the control command of ego vehicle. Then, by solving the Riccati equation, the optimal control law can be explicitly derived as constant feedback/feedforward gains. This means LQR require almost no computational effort in the real-time implementation. Another prevailing optimization-based control method is Model Predictive Control (MPC)[47].

Similar to LQR, the objective function in MPC usually considers the system states and control command [38], [48], but a difference is that the time horizon of cost function is finite. Moreover, MPC is indeed solving the actual optimization problems in real time. Based on a system model and currently measured system states, the MPC makes predictions of system states in a finite rolling horizon given different planned control efforts, and find the optimal control effort which minimizes the quadratic objective function. However, only the first move of the control efforts will be executed at the present time step. Compared with LQR, constraints on the system states or control efforts can be applied to the optimization problem. Besides, MPC can better handle the modelling error of the system due to the feedback nature of the rolling horizon process [48]. The major downside of MPC is that the online optimization requires much higher computing capability for the real-time implementation.

A variety of CACC systems, including PF/APF CACC and rule-based/optimization-based CACC, were tested together in Grand Cooperative Driving Challenge (GCDC) 2011 [32] and 2016 [49], which positively indicated that heterogeneous CACC vehicles can be compatible and implemented cooperatively in the traffic. To prepare CACC for the final large-scale deployment, the latest research efforts have paid more attention to the issues of communication unreliability [50], [51], cyber-attacks [52], [53], and forming/splitting of CACC platoon [54]–[56].

Except the two technical limitation of CACC as mentioned in Chapter 1, the studies above also share a drawback in evaluating their methods. These CACC designs were evaluated either by numerical simulation or field test, but they uniformly used fabricated trajectories of preceding/leading vehicle to construct the car-following scenarios. For example, trapezoid speed profile was used in [45], triangle speed profile was used in [22], and [48] assumed impulse disturbance in speed of leading vehicle. With these fabricated test scenarios, the benefits of the different methods in real life are hard to be quantified and compared. More realistic test scenarios, especially those based on real-world data, should be utilized for more meaningful evaluations of performances.

2.2 CAV: platooning with unconnected vehicles

There have been a few insightful previous research that tried to enhance the performance of CAV when encountering an unconnected preceding vehicle. This section goes into detail of these existing methods and points out their limitations.

To partially maintain the favorable properties of CACC when the communication from preceding vehicle becomes unavailable or unstable, graceful degradation of CACC (dCACC) was proposed in [34]. Since the feedforward signal from the preceding vehicle cannot be obtained, dCACC turned to estimated preceding vehicle's acceleration using onboard radar. Singer acceleration model [57] was adopted to roughly predict how the acceleration of preceding vehicle evolves from the current value and random disturbances from downstream. Based on Singer model, a Kalman filter was used as a state observer. This observer fused the roughly predicted acceleration of preceding vehicle, and the actual measurements on inter-vehicle spacing and relative speed (which were linked to preceding vehicle's acceleration but flawed with noises). Lastly, the acceleration estimated by this state observer was given to the CACC controller as the feedforward signal. It was proven that by assuming a proper disturbance variance in the Singer model, dCACC can fulfill string stability at a short time gap which is less than a half of that needed for ACC. Nevertheless, the quality of radar measurement was shown as a main drawback of dCACC. According to the nature of the state observer, the estimated acceleration is either smooth but lagged (slow estimation mode with more trust in prediction) or jerky but fast (with more trust in radar measurements). In pursuit of such improvement in string stability, the estimation must be sufficiently fast. Therefore, the control command of dCACC vehicle tended to be jerkier than usual. The smoothness of the vehicle trajectory, i.e., the ride comfort had to be sacrificed for string stability.

On the other hand, Connected cruise control (CCC) [58] was proposed to explore the benefits of communication with out-line-of-sight preceding vehicles when the closest one or more preceding vehicles are unconnected. By incorporating the accelerations or relative speed of remote vehicles in the state feedback with intended delay, CCC was able to stabilize initially unstable vehicle strings when the car-following behaviors of unconnected vehicles are within certain ranges. In [59], the dynamics of CCC systems under effects of complex connectivity structure and communication delay were further investigated, and motif-based approach was proposed to facilitate CCC design in large networks. However, the tuning of such CCC designs require the dynamics of unconnected vehicle to be known. Thus, the car-following behavior identification of unconnected vehicle [60] needs to be conducted before the CCC can work properly. This process could take tens of seconds [60], and even after the identification being done, car-following behavior of the unconnected vehicle is unlikely to remain time-invariant if it is driven by human.

An optimal-control-based CCC [61] was proposed to represent the time-variant behaviors of the preceding vehicle with mean values and distributions of human parameters, but it costed even longer time to identify such distributions. Actually, this can be problematic for control design because the human parameters were not perfectly stochastic, i.e., they can continuously deviate from their past mean values for tens of seconds [61], which is long enough to cause unexpected consequences (such as loss of string stability). On the other hand, while the design of existing CACC systems is based on zero-spacing-error rule and the strong string stability of individual vehicle [32], the emphasis of CCC often lays on head-to-tail string stability of platoon, which could greatly suppresses the traffic turbulence but not necessarily help maintain the desired spacing for the individual vehicle [62].

A centralized CACC which considers unconnected human-driven vehicle in the platoon was proposed in [63]. The platooned vehicles are required to report their status to the lead vehicle, and the status of unconnected human-driven vehicles are assumed to be estimated by adjacent CAVs. It is also assumed that the car-following model parameters of the human-driven vehicles have been reasonably identified. The lead vehicle then decides the best acceleration for each vehicle based on min-max model predictive control (MM-MPC) and identified parameters of human-driven vehicles. This method is still restricted by the car-following behavior identification beforehand. Although the uncertainty in vehicle dynamics and communication delay were handled by the MM-MPC in its simulation validation, only a small identification error in human parameters was assumed, and the acceptable boundary of the error was not examined.

In summary, the problem of CAV platooning in mixed traffic has attracted increasing research efforts but has not been adequately addressed. The extra process of human parameters identification is one of the main restrictions on the existing approaches when faced with large uncertainty of human behaviors.

2.3 CHV: more than driver advisory systems

A commonly expectation in the previous research efforts [23], [33], [61], [62], [64], [65] of CAV and mixed traffic is that there would be a large number of human-driven vehicles with connectivity (including after-market installation), referred as connected human-driven vehicle (CHV). On one hand, many connected vehicles would not be automated due to the relatively high cost of onboard sensors [20]. On the other hand, CAVs might be degraded to manual mode sometimes when the CACC system does not satisfy the driver, who thus prefers to be in control of the vehicle. Such “intended disengagement” has been frequently occurring to the ACC in low/median-speed traffic [66].

It was proposed in [23] that CAVs should take advantages of these human-driven CVs, which are supposed to broadcast “here I am” messages, to facilitate the use of CACC. However, there would still be no direct benefit for CHV themselves. Being always leading vehicles of other CACC vehicles, human-driven CVs have neither a crisper response to preceding vehicle’s maneuvers nor a fuel saving as the CAV followers do [33]. To fully take advantage of vehicular connectivity, human-driven CVs should be able to receive assistance via the connectivity, instead of being just “information providers.”

Although there exist a few of previous work in the applications of CHV, they turned out to be mostly focused on driver advisory systems for very limited situations. A representative application is CV-based eco-driving system [67] that aims for improving fuel economy when the CV approaches a signalized intersection. It utilizes vehicle-to-infrastructure (V2I) communication to obtain signal phase and timing information, and accordingly recommends the optimal speed to drivers. Other CV-based applications include lane changing advisory system [68] which attempts to reduce merging conflicts around on-ramp by encouraging early mainline freeway lane changes, and cooperative collision warning system [69] which alerts the driver when a highly potential collision is projected using V2V communication. Obviously, the benefits of these advisory systems are subject to human driver's compliance level. The received information in these systems is only used to generate suggestions to the driver, instead of actuating the vehicle directly.

A possible step forward from advisory systems is to allow the coexistence of automatic control and driver's actions in a beneficial way. Typical CACC systems [22], [30] make the control decision based on three components: spacing feedback, speed feedback, and acceleration feedforward. For a CHV, the accurate distance to the preceding CV is unavailable due to absence of range sensor, but the reliable speed and acceleration information of the preceding vehicle can still be obtained via V2V communication, and could be used to co-actuate the vehicle, together with a human driver.

This collaboration between machine and human has become technically and economically feasible with the adoption of electronic actuators in modern vehicles. Electronic throttle [70] has been applied to almost every car. The recent generation of Electronic Stability Program/Control (ESP/ESC) [71], which enables programmed brake control, has also been massively adopted in new cars. Furthermore, there are increasing number of hybrid/electric vehicles equipped with drive-by-wire [72], [73] technology. These electronic actuators can sense human's will through pedals, monitor vehicle's status, and apply corresponding actions on throttle/brake. They not only help drivers achieve their intention faster and more precisely, but also provide the vehicle's software with the convenience to modify or override human's initial action when necessary. This means the additional investment in automatic actuators is no longer needed for many vehicles.

A preliminary design of this type of system is acceleration-based Connected Cruise Control (CCC) [58] which allows human driver in the loop of vehicle control. It proposed to give the human-driven vehicle an extra acceleration which was half that of the preceding vehicle. Theoretical analysis indicated that the extra acceleration can help stabilize initially string-unstable vehicle platoons. However, this CCC has never been evaluated with real human drivers, while it was designed and analyzed with a strong assumption that human's behavior pattern can be perfectly obtained and unaffected by the extra acceleration. This makes its effectiveness questionable in the field. In addition, the speed information of preceding vehicle was omitted and not fully made use of in the CCC. This is unreasonable as the speed difference between ego vehicle and preceding vehicle makes a big difference in the acceleration of the following vehicle in order to maintain a stable following gap.

2.4 Summary of literature review

In summary, there have been extensive research efforts made in the area of cooperative longitudinal control of vehicles. CACC with all kinds of communication topology and control methods were developed. String stability has been widely considered as an important design goal. Except the two technical limitation of CACC pointed out in Chapter 1, another drawback shared by the existing CACC studies is that the performance evaluations were not conducted in realistic car-following scenarios.

The problem of CAV platooning with unconnected vehicles has attracted increasing research efforts (e.g.., dCACC and CCC) but has not been adequately addressed. The dCACC is limited by the sacrifice of ride comfort, while CCC requires extra process of human parameters identification are the main restrictions on the existing approaches when faced with large uncertainty of human driving behaviors.

The connectivity in CHV has not been fully made use of either. Most of existing works focused on connected-vehicle-based driver advisory systems for very limited situations. A possible step forward from advisory systems is to allow the coexistence of automatic control and driver's actions in a beneficial way. The only one previous study that considered actuating the vehicle together with human driver is acceleration-based CCC. However, it was apparently flawed by its assumption of consistent human behaviors and the lack of validation with real driver.

Chapter 3. CACC with Unconnected Vehicle (CACCu)

This chapter proposes a new CACC algorithm, dubbed as CACCu, which allows CAVs to closely follow an unconnected preceding vehicle, instead of falling back to ACC. Different from the existing CCC, CACCu aims to robustly handle various unconnected vehicle's car-following behaviors, without requiring identification process or extra information on the unconnected vehicles. While CACCu can be extended and evaluated in more general scenarios, this chapter starts with the detailed control design and analysis of CACCu in three-vehicle sandwich scenario (i.e., an unconnected vehicle is in between of two connected vehicles), which is simple but with the high probability to occur among the mixed platooning scenarios. The string stability, safety, comfort, and fuel efficiency performances of CACCu are evaluated in high-fidelity simulations using real vehicle trajectory data from Next Generation Simulation (NGSIM) program and a physics-based vehicle dynamics model from PreScan. The typical ACC [30] and acceleration-based CCC [58] serves as performance baselines in the evaluation.

3.1 Framework

Typical CACC systems obtain the acceleration or desired acceleration of nearest preceding vehicle as a feedforward signal [74]. This feedforward signal can efficiently help eliminate spacing error (i.e., the difference between actual spacing and desired spacing), and thus enables safe driving at short gaps [9]. However, as shown in Fig. 3.1, when the ego vehicle encounters an unconnected preceding vehicle or vehicles, such feedforward signal is not available. Instead of degrading to ACC, the proposed CACCu turns to utilize the closest connected vehicle ahead (i.e., the $(n+1)^{th}$ preceding vehicle in Fig. 3.1) as the source of feedforward signal. An additional filter of "virtual preceding vehicle(s)" is inserted before the original feedforward filter of CACC, to compensate for the effects of n unconnected preceding vehicle(s) in between. Assuming random clustering of vehicles [33], the probability (P_n) of having different n (i.e., the number of unconnected preceding vehicles) for a CAV is directly linked to Market Penetration Rate (MPR) of vehicular connectivity. Fig. 3. 2 shows how P_n varies with n and MPR, where P_n is calculated as $MPR \cdot (1 - MPR)^{n-1}$. It can be seen that enabling CACCu for n =1 (i.e., three-vehicle sandwich scenario) could make the most considerable complement to CACC ($n = 0$). Hence, a special emphasis is laid on such three-vehicle sandwich scenario.

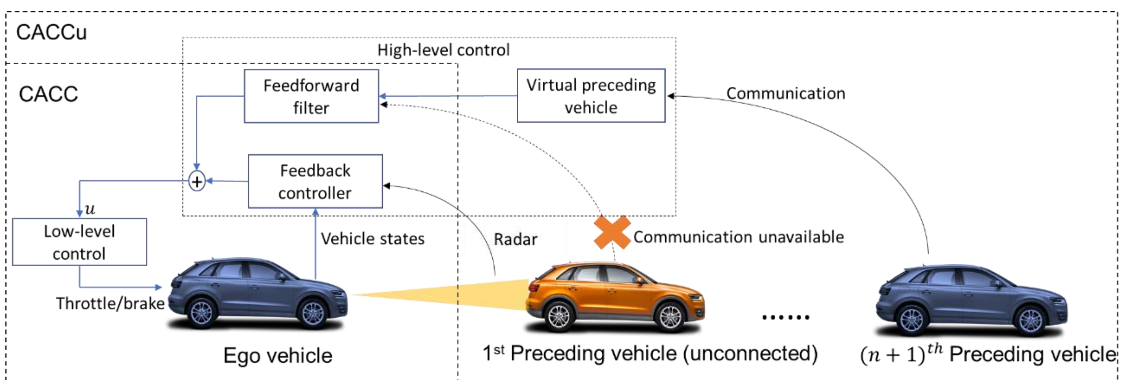


Fig. 3.1 Control scheme of CACCu

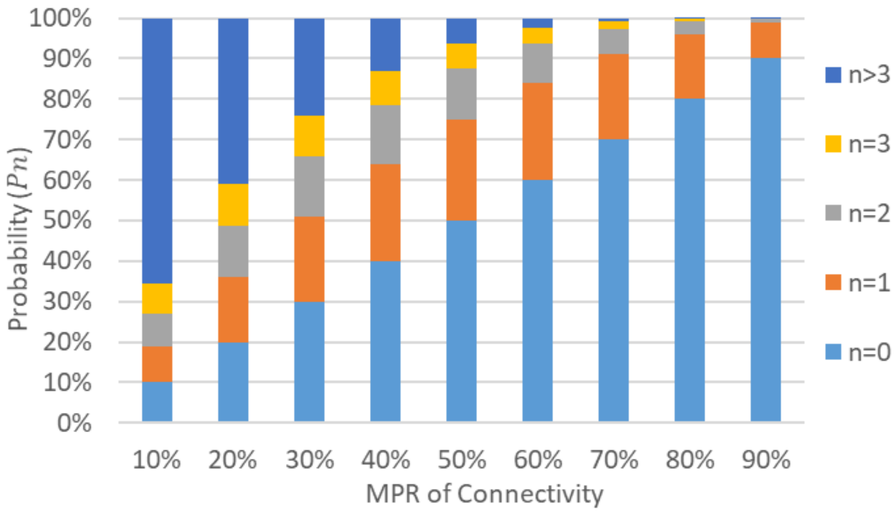


Fig. 3.2 Market Penetration Rata (MPR) of connectivity and the probability that the closest connected preceding vehicle is n vehicle(s) away

In addition, a bi-level control structure is needed due to the nonlinearity of vehicle dynamics. The high-level control decides the desired acceleration (u), while the low-level control determines how to actuate the throttle and brake to achieve this desired acceleration. For high-level control, a linear time-invariant control law is pursued in this study for easy parameterization and implementation. As shown in Fig. 3.1, the proposed CACCu can be directly extended from an existing CACC system with minimum re-design (i.e., only inserting a “virtual preceding vehicle”). Such design of CACCu would also facilitate the straightforward performance comparisons with ACC using exactly the same feedback configuration.

3.2 Control design

In this section, the three key components of CACCu are described respectively, including the consideration in human car-following behaviors and the designs of high/low-level controls.

Stochastic car-following behaviors of unconnected vehicle

The linearized optimal velocity model (OVM) [75] is adopted to describe car-following behaviors of the unconnected human-driven vehicle around a traffic equilibrium (i.e., steady state with constant velocity):

$$\begin{aligned} &h_1(t) = x_2(t) - x_1(t) - l_2 \\ &\ddot{x}_1(t) = \alpha_1 \left(\frac{1}{t_{1,h}} h(t-\varphi_1) - \dot{x}_1(t-\varphi_1) \right) + \beta_1 \dot{h}(t-\varphi_1) + e_m(t) \end{aligned} \tag{3.1}$$

Where t is time, $\dot{*}$ denotes the variable's derivative in respect to time, $x_1(t)$ and $x_2(t)$ are locations of the human-driven vehicle and its preceding vehicle, h_1 is the inter-vehicle spacing, with l_2 being the length of preceding vehicle, α_1 and β_1 are human control gains, φ_1 is the human reaction time, $\frac{1}{t_{1,h}}$ is spacing policy slope with $t_{1,h}$ being the desired time gap of the human driver, and $e_m(t)$ is a noise term representing the unmodeled behaviors of the human driver. Model (3.1) indicates that the human driver desires a velocity-dependent spacing, and regulates the spacing error and speed difference from the preceding vehicle simultaneously. In fact, other frequently used car-following models (e.g., intelligent driver model) can also be linearized into the same form of (3.1) [75].

It should be noted that the human parameters in (3.1) vary from person to person, and even for one single driver, they may change stochastically over time. To incorporate the variation of human parameters yet avoid the time-consuming identification [60], [61], a realistic and convenient assumption is adopted in this study.

Assumption 1: the driver's car-following behavior should be represented by different φ_1, α_1, β_1 and $t_{1,h}$ in every short period of regulation (i.e., the time from the traffic equilibrium being disturbed until a new equilibrium is reached).

In other words, the human driver responds to each speed perturbation in different ways, but the driver's behavior during one regulation period is relatively stable. This assumption requires that any control design involving human driver should be able to handle a range of human parameters instead of a specific combination. Meanwhile, the stochastic behavior of human driver is approximated by a sequence of linear time-invariant systems (thus transfer functions exist), which will bring great convenience in the control design and analysis.

Taking the Laplace transform of (3.1) with zero initial conditions, the transfer function of the human-driven vehicle in each regulation period can be obtained:

$$T_1(s) = \frac{L\big(x_1(t)\big)}{L\big(x_2(t)\big)} = \frac{K_1(s)}{s^2 e^{\varphi_1 s} + K_1(s) + \alpha_1 s} \tag{3.2}$$

Where $L(\cdot)$ denotes Laplace transform and

$$K_1(s) = \frac{\alpha_1}{t_{1,h}} + \beta_1 s$$

To incorporate all kinds of human drivers, the possible ranges of human parameters reported in existing studies are summarized below:

- The preferred time gap $t_{1,h}$ of highway drivers is found to be 1~2s [76].
- The human delay φ_1 was reported to be 0.5~1.5s in [77], while [78] found the brake delay in normal case to be 0.92~1.93s, and acceleration delay to be 0.4~1.5s.

- For the human control gains α_1 and β_1, previous literature [58], [79] used the average value of 0.6 and 0.9, which are derived from macroscopic data. However, field test [61] determined the average values of α_1 and β_1 to be 0.2 and 0.4. Considering the large difference between these two sets of value, the average values of 0.4 and 0.65 can be assumed for α_1 and β_1, respectively, as compromise.

It is worth noting that different human parameters are not likely to appear with the equal probability. In control design, the recurrent combinations of human parameters should be given more considerations. Thus, a probability model is needed to capture the uneven distribution of human parameters. Although with limited number of participating drivers, [61] has identified bell-shaped distributions of human parameters and treated them as independent. In this paper, the human parameters are assumed to follow independent normal distributions, whose means and variances are determined based on the aforementioned ranges of human parameters.

Assumption 2: for the population of all drivers, the φ_1, α_1, β_1 and $t_{1,h}$ follow independent normal distribution as below:

- Desired time gap $t_{1,h} \sim N(1.5, 0.25^2)$, which means it has 95% probability to be 1~2;
- Human delay $\varphi_1 \sim N(1, 0.25^2)$, which means it has 95% probability to be 0.5~1.5;
- Human gain $\alpha_1 \sim N(0.4, (\frac{0.4}{2.6})^2)$, which means it has 98% probability to be 0~0.8 and only 1% probability to be negative;

- Human gain $\beta_1 \sim N(0.65, (\frac{0.65}{2.6})^2)$, which means it has 98% probability to be 0~1.3 and only 1% probability to be negative.

Nevertheless, the design of CACCu does not rely on a specific type of probability model, as shown in the rest of paper. *Assumption* 2 is free to be modified or replaced when there are new findings from more sophisticated investigations on human parameters.

High-level control

CACCu follows the basic structure of predecessor-following CACC which is featured by the feedforward-feedback control and velocity-dependent spacing policy [30]. The main difference is that the feedforward signal is from the further preceding vehicle instead of the 1st one. Thus, the CACC feedforward filter needs to be modified. When the 2nd preceding vehicle is a connected vehicle, the car-following behavior of CACCu vehicle is as below:

$$
\begin{aligned}
&h_0(t) = x_1(t) - x_0(t) - l_1 \\
&h_{0,d}(t) = t_{0,h}\dot{x_0}(t) + h_{0,st} \\
&e_0(t) = h_0(t) - h_{0,d}(t) \qquad (3.3) \\
&u_0(t) = k_{0,p}e_0(t) + k_{0,d}\dot{e_0}(t) + f_0(\ddot{x}_2(t-\theta_0)) \\
&\ddot{x}_0(t) = g_0(u_0(t))
\end{aligned}
$$

Where $x_0(t)$ is the location of the ego vehicle, h_0 is the spacing from the preceding vehicle, with l_1 being the length of the 1st preceding vehicle, $h_{0,d}(t)$ is the desired spacing, $h_{0,st}$ is the standstill spacing, $t_{0,d}$ is the desired time gap, $e_0(t)$ is the spacing error, $k_{0,p}$ and $k_{0,d}$ are the gains of the proportional-derivative (PD) feedback controller, $f_0(\cdot)$ is the new feedforward filter, $\ddot{x}_2$ is the acceleration of the second preceding vehicle, and θ_0 is the communication delay, $g_0(\cdot)$ is the vehicle dynamics of the CACCu vehicle.

A first-order lag system with constant delay is adopted to describe the longitudinal vehicle dynamics $g_0(\cdot)$ in time domain:

$$g_0\big(u_0(t+\emptyset_0)\big) + \tau_0 \dot{g}_0\big(u_0(t+\emptyset_0)\big) = u_0(t) \tag{3.4}$$

The corresponding transfer function in Laplace domain is:

$$G_0(s) = \frac{L\big(x_0(t)\big)}{L\big(u_0(t)\big)} = \frac{1}{s^2(1+\tau_0 s)} e^{-\emptyset_0 s} \tag{3.5}$$

Where τ_0 is the system lag and $\emptyset_0$is the actuator delay.

The $f_0(\cdot)$ should be designed so that the spacing error can be eliminated. According to (3.3) with zero initial conditions, the Laplace transform of spacing error can be obtained:

$$L\big(e_0(t)\big) = \frac{1}{1+G_0(s)K_0(S)H_0(s)} L\big(x_1(t)\big) - \frac{D_0(s)G_0(s)F_0(S)H_0(s)s^2}{1+G_0(s)K_0(S)H_0(s)} L\big(x_2(t)\big) \tag{3.6}$$

Where

$$F_0(s) = L(f_0(t))$$

$$K_0(s) = k_{0,p} + k_{0,d}s$$

$$H_0(s) = 1 + t_{0,h}s$$

$$D_0(s) = e^{-\theta_0 s}$$

Let $L\big(e_0(t)\big) = 0$, then:

$$L\big(x_1(t)\big) - D_0(s)G_0(s)F_0(S)H_0(s)s^2L\big(x_2(t)\big) = 0$$

And thus

$$F_0(s) = \frac{1}{D_0(s)G_0(s)\,H_0(s)s^2}\frac{L\big(x_1(t)\big)}{L\big(x_2(t)\big)} \;= \frac{1}{D_0(s)G_0(s)\,H_0(s)s^2}T_1(s)$$

However, the exact value of communication delay and human parameters are unpredictable in real world, so a feasible feedforward filter is:

$$F_0(S) = \frac{1}{G_0(s)\,H_0(s)s^2}T_1'(s) \tag{3.7}$$

Where $\frac{1}{G_0(s)\,H_0(s)s^2}$ is the original feedforward filter used in CACC [30], and $T_1'(s)$ is the additional filter of a "virtual preceding vehicle" that has the same form of $T_1(s)$:

$$T_1'(s) = T_1'\big(\alpha_1', \beta_1', \varphi_1', t_{1,h}', s\big) \tag{3.8}$$

Where $\alpha_1', \beta_1', \varphi_1', t_{1,h}'$ are the parameters of the "virtual preceding vehicle."

Since there is little chance to make $T_1'(s)$ exactly equal to $T(s)$(thus to perfectly predict the acceleration of the first preceding vehicle), parameters $(\alpha_1', \beta_1', \varphi_1', t_{1,h}')$ are left to be tuned so that CACCu vehicle can stay string-stable for a vast range of unconnected vehicle behaviors described by $(\alpha_1, \beta_1, \varphi_1, t_{1,h})$. Obviously, when the feedforward signal comes from more distant vehicle (i.e., when there are multiple unconnected vehicles in between), $T_1(s)$ and $T_1'(s)$ should be replaced by the combined transfer function of multiple human-driven vehicles, and tuning of this transfer function will require more effort, as shown later in Section 4.

Finally, the transfer function of the CACCu vehicle can be derived combining (3.3) and (3.7):

$$T_0(s) = \frac{L(x_0(t))}{L(x_1(t))} = \frac{H_0(s)G_0(s)K_0(s) + D_0(s)T_1'(s)/T_1(s)}{H_0(s)(1 + H_0(s)G_0(s)K_0(S))} \tag{3.9}$$

As comparison, the existing CACC systems let $f_0(\ddot{x}_2(t-\theta_0)) = 0$ when following an unconnected vehicle. This setting degrades the CACC to ACC and leads to a transfer function of:

$$T_0(s) = \frac{L(x_0(t))}{L(x_1(t))} = \frac{G_0(s)K_0(s)}{1 + G_0(s)K_0(S)H_0(s)} \tag{3.10}$$

Low-level control

According to (3.6), the high-level controller outputs the desired acceleration to the vehicle dynamics. However, the longitudinal motion of vehicle is directly controlled by the throttle and brake. Thus, a low-level controller is needed to convert the desired acceleration to proper throttle and brake action so that the command from high-level controller can be accurately achieved. A typical low-level controller [80] utilizes the inverse engine torque map and a set of feedforward signals (i.e., vehicle speed, engine speed, and transmission ratio) to pre-compensate the nonlinear behaviors of the engine, transmission system, air drag and rolling resistance, leading to a first-order linear relationship between desired acceleration and actual acceleration as described by (3.4), and a third-order linear relationship between desired acceleration and vehicle position as described by (3.5).

3.3 String Stability Analysis

String stability is one of the most important design goal of longitudinal vehicle control. In this study, CACCu is required to guarantee string stability not only for a single combination of $\alpha, \beta, t_h, \varphi$ but for a broad range of $\alpha, \beta, t_h, \varphi$. A widely-accepted version of string stability is defined in [30], that is, given any disturbance in the longitudinal movement of preceding vehicle, the following vehicle should not amplify this disturbance. While string stability can also be defined in terms of spacing error or control input, they are less practical when human driver is involved. According to [30], the string stability of ego vehicle is fulfilled when the magnitude of its frequency response is always no greater than 1:

$$SS = \|T_0(j\omega)\|_\infty = \left\| \frac{H_0(j\omega)G_0(j\omega)K_0(j\omega) + D_0(j\omega)T_1'(j\omega)/T_1(j\omega)}{H_0(j\omega)(1 + H_0(j\omega)G_0(j\omega)K_0(j\omega))} \right\|_\infty \leq 1 \qquad (3.11)$$

Where $\|\cdot\|_\infty$ denotes the maximum magnitude over all frequency ω, and j is the imaginary unit. Because $T_0(j\omega) = \frac{L(x_0(t))}{L(x_1(t))} = \frac{L(\dot{x}_0(t))}{L(\dot{x}_1(t))} = \frac{L(\ddot{x}_0(t))}{L(\ddot{x}_1(t))}$, condition (3.11) can be approximately interpreted as that given any perturbation from the downstream, the speed or acceleration peak of ego vehicle caused by the perturbation should not exceed that of the preceding vehicle.

To measure CACCu's robustness against the uncertain car-following behaviors of preceding vehicle, String Stability Ratio (SSR) is defined as the probability that ego vehicle stays string-stable given all different kinds of $(\alpha_1, \beta_1, \varphi_1, t_{1,h})$. By definition, SSR can be computed as an integral of the probability density over all the string-stable combinations of $(\alpha_1, \beta_1, \varphi_1, t_{1,h})$:

$$SSR = \int\int\int\int p(\alpha_1, \beta_1, \varphi_1, t_{1,h})\xi(SS)d\alpha_1 d\beta_1 d\varphi_1 dt_{1,h} \tag{3.12}$$

Where

$$\xi(SS) = \begin{cases} 1 & if\ SS \leq 1 \\ 0 & if\ SS > 1 \end{cases}$$

$f(\alpha_1, \beta_1, \varphi_1, t_{1,h})$ is the joint probability density function (PDF) of human parameters, and SS is the string stability determinant defined by (3.11). According to *Assumption* 2, $f(\alpha_1, \beta_1, \varphi_1, t_{1,h})$ can be calculated as the product of PDFs of all the human parameters:

$$p(\alpha_1, \beta_1, \varphi_1, t_{1,h}) = \frac{1}{0.25\cdot\sqrt{2\pi}}\exp\left(-\frac{(t_{1,h}-1.5)^2}{2\cdot 0.25^2}\right)\cdot\frac{1}{0.25\cdot\sqrt{2\pi}}\exp\left(-\frac{(\varphi_1-1)^2}{2\cdot 0.25^2}\right)\cdot$$

$$\frac{1}{(\frac{0.4}{2.6})\cdot\sqrt{2\pi}}\exp\left(-\frac{(\alpha_1-0.4)^2}{2\cdot(\frac{0.4}{2.6})^2}\right)\cdot\frac{1}{(\frac{0.65}{2.6})\cdot\sqrt{2\pi}}\exp\left(-\frac{(\beta_1-0.65)^2}{2\cdot(\frac{0.65}{2.6})^2}\right) \tag{3.13}$$

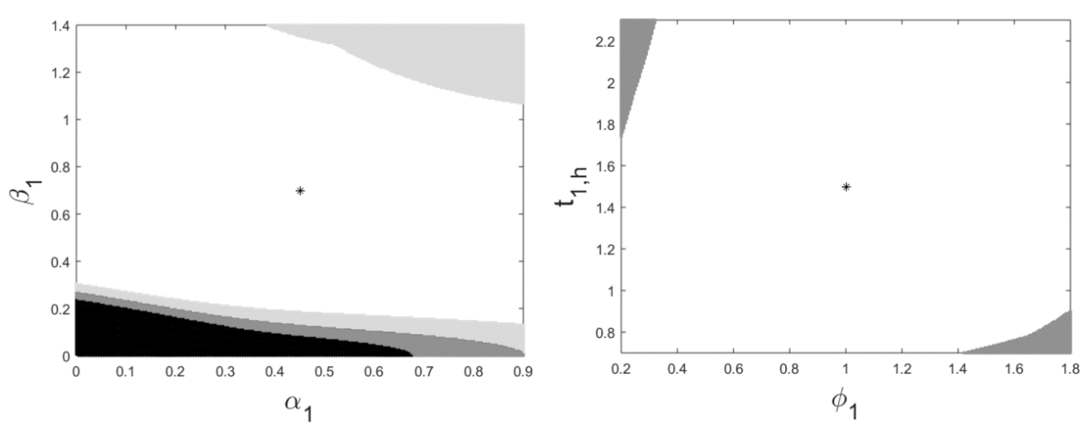
To obtain an ideal SSR, the CACCu vehicle should not only use optimal virtual vehicle T_1' but also choose proper feedback controller K_0 and spacing policy H_0 based on the operating condition, i.e., the vehicle dynamics G_0 and average communication delay D_0.

Higher control gains in K_0 typically improve string stability, but meanwhile they lead to more aggressive behaviors and higher sensitivity to sensor noise, thus may impair the ride comfort. Two pairs of $(k_{0,p},\ k_{0,d})$ adopted in field tests are considered here:

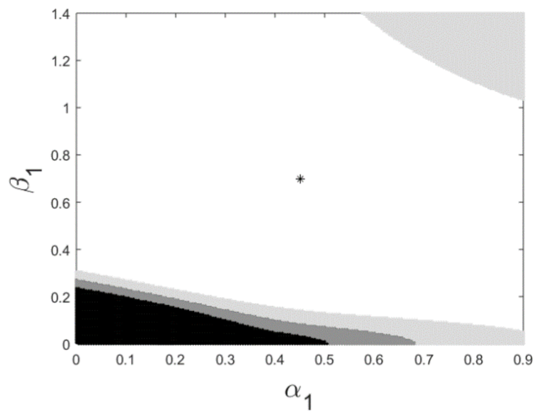
- Low gains used in field test [30]: $k_{0,p} = 0.25,\ k_{0,d} = 0.5$;
- High gains used in field test [81]: $k_{0,p} = 0.3,\ k_{0,d} = 0.7$;

First, the effects of desired time gap and control gains on the string stability are explored, assuming perfect vehicle dynamics $\tau_0 = 0$ and $\emptyset_0 = 0$ and perfect communication $\theta_0 = 0$. Due to the complexity of (3.11), the numerical results of the string-stable space of $(\alpha_1, \beta_1, \varphi_1, t_{1,h})$ are shown in Fig. 3.3. Using MATLAB optimization toolbox, $(\alpha_1', \beta_1', \varphi_1', t_{1,h}')$ have been optimized to (0.99, 0.62, 0, 0.72) for CACCu with low gains and (0.76, 0.51, 0, 0.57) for CACCu with high gains.

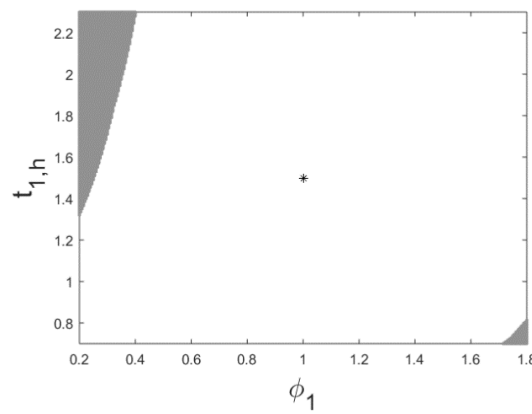
Fixing $\tau_0 = \emptyset_0 = \theta_0 = 0$, Fig. 3.3 (a)~(d) show the string-stable ranges of the human parameters ($\alpha_1, \beta_1, \varphi_1,$ and $t_{1,h}$) under low/high control gains. Blank area denotes the string-stable range when desired time gap of ego vehicle is set 0.8s; lighter/darker shaded area denotes the increased string-stable range when desired time gap of ego vehicle increases to 1.0s/1.2s; the darkest shaded area denotes the string-unstable range when desired time gap of ego vehicle is 1.2s. Fig. 3.3 (a), (c) show string-stable ranges of $\alpha_1$and β_1 (when $\varphi_1 = 1,\ t_{1,h} = 1.5$) for low and high gains, respectively. Fig. 3.3 (b), (d) show string-stable ranges of φ_1 and $t_{1,h}$ (when $\alpha_1 = 0.4,\ \beta_1 = 0.65$) for low and high gains, respectively. Fig. 3.3 (e) shows SSR for low/ high control gains when the desired time gap of ego vehicle is set 0.6s~1.4s.



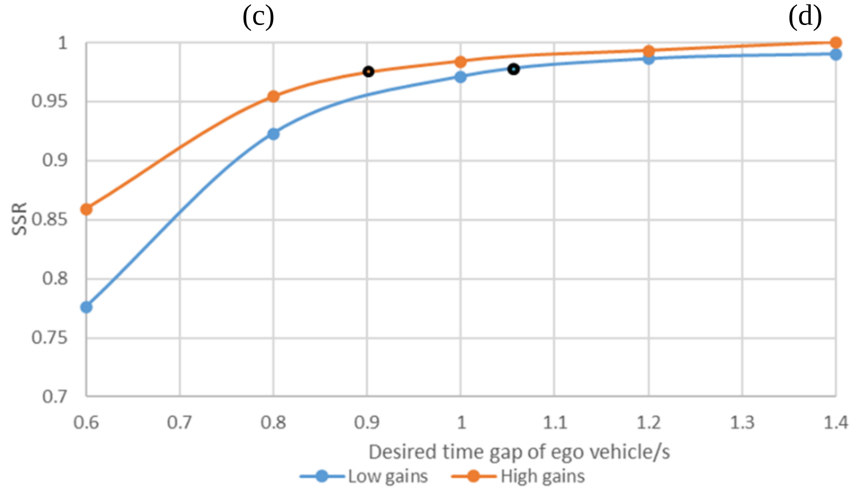
(a) (b)



(c)



(d)



(e)

Fig. 3.3 The string-stable range of human parameters and SSR under different control gains and desired time gaps

It can be seen from Fig. 3.3(a)~(d) that CACCu can provide broad string-stable ranges of human parameters. Given certain $t_{1,h}$ and φ_1, CACCu tends to lose its string stability when β_1 and α_1 are both low or both high. Given certain α_1 and β_1, CACCu tends to lose its string stability when $t_{1,h}$is much larger than φ_1(i.e., the preceding vehicle has fast response but maintains a long gap) or the inverse case. On the other hand, the string-unstable area shrinks when longer desired gap and higher control gains are used. Fig. 3.3(e) shows that the SSR climbs to 99.7% when the high gains (e.g., 0.3, 0.7) and a desired time gap of 1.2s are used, which means CACCu vehicle can keep string-stable given almost all kinds of unconnected preceding vehicle. As comparison, by using (3.10) it can be found that an ACC vehicle with the same control gains needs a time gap ≥2.6s to maintain its string stability. This gap is more than twice the gap required by CACCu. From another perspective, when driving at the same desired gap, a CACCu vehicle can better attenuate the speed oscillation from downstream than an ACC vehicle can do.

Considering that string stability is not a safety-critical requirement, it will be too trivial to prepare the CACCu for any combination of $\alpha_1, \beta_1, \varphi_1,$ and $t_{1,h}$, especially after knowing that the string-unstable areas are at the edge of the parameter space that has low probability to occur. For this reason, a "critical gap" is defined as the desired time gap which can guarantee string stability at 97.5% probability (i.e., SSR $\geq$ 97.5%). Driving at the critical gap, CACCu can offer a dominant capability to accommodate human uncertainty over previous research efforts. It cancels the necessity of human parameters identification in advance, while string stability can be fulfilled in most cases. It can be seen in Fig. 3.3(e) that the critical gaps for CACCu with low/high control gains are 0.9s/1.05s, respectively, when assuming perfect vehicle dynamics and communication. More conservative critical gap can also be defined and found in Fig. 3.3(e).

Then, the effects of communication delay and imperfect vehicle dynamics on string stability are investigated. The possible values of communication delay θ_0 and vehicle lag τ_0, and actuator delay $\emptyset_0$ according to previous field tests have been summarized in [74]:

$$0.02 \leq \theta_0 \leq 0.2,\ 0.1 \leq \tau_0 \leq 0.8,\ 0.02 \leq \emptyset_0 \leq 0.25$$

Fig. 3.4 shows the different critical gaps under communication delay of 0~0.2s when fixing $\tau_0 = \emptyset_0 = 0$. It can be found the communication delay has mild impact on the string stability. The critical gaps of CACCu with low and high both increase by 0.15s when the largest communication delay of 0.2s is present. If the V2V communication is conducted every 100ms and the zero-order hold (ZOH) is applied to the received message, an average communication delay of 50ms can be expected. In this case, the critical gap only increases by 0.05s in high-gain case.

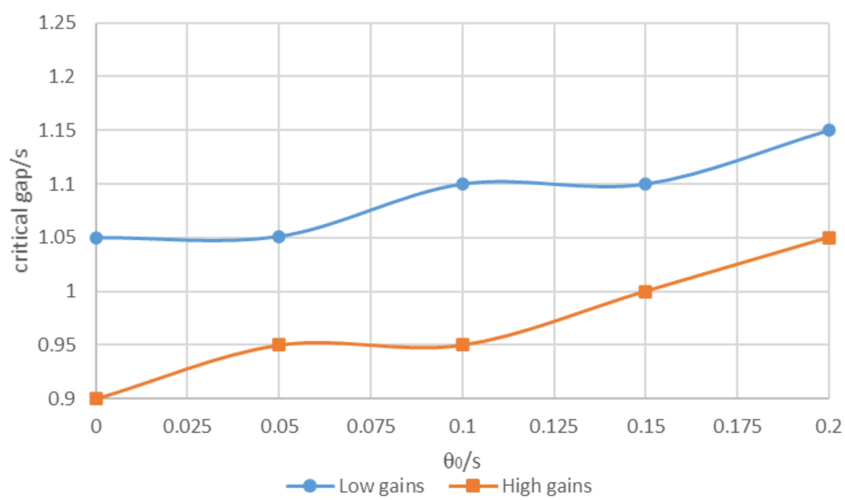
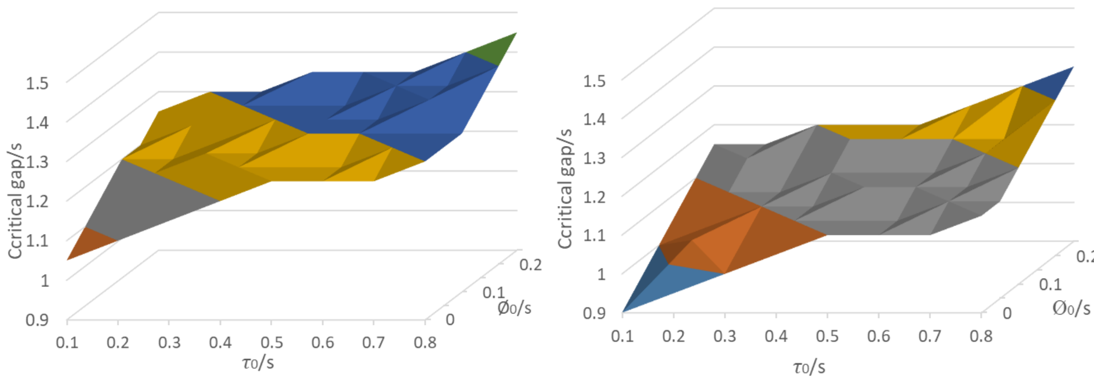


Fig. 3.4 Critical gaps under communication delay θ_0=0~0.2s ($\tau_0 = \emptyset_0 = 0$)

Fixing $\emptyset_0 = 50\text{ms}$, Fig. 3.5 shows the critical gaps under the effects of different vehicle lag τ_0 and actuator delay θ_0. The effect of vehicle dynamics imperfection is simple: higher τ_0 and θ_0 results in longer critical gap for both low-gain and high-gain CACCu, and high gains always offer shorter critical gap. It is also found that the optimal parameters of T_1' can greatly vary with the vehicle dynamics. For example, when vehicle dynamics of $\tau_0 = 0.8$ and $\emptyset_0 = 0.25$, instead of $\tau_0 = \emptyset_0 = 0$, are considered, the optimal $(\alpha_1', \beta_1', \varphi_1', t_{1,h}')$ in high-gain case shifts from (0.76, 0.51, 0, 0.57) to (0.91, 0.33, 0, 0.93). In practice, not only the proper combination of control gains and desired gap should be chosen from Fig. 3.5, but also the optimal parameters $(\alpha_1', \beta_1', \varphi_1', t_{1,h}')$ should be specified according to the vehicle dynamics.



(a) Low gains (b) High gains

Fig. 3.5 The critical gaps under different vehicle dynamics for CACCu with low/ high control gains

In summary, the CACCu controller can be tuned by maximizing the string stable ratio (SSR). The analysis shows that the proposed CACCu is able to stay string-stable at a desired time gap significantly shorter than that required by ACC, when facing almost all kinds of unconnected preceding vehicles. This desirable property of CACCu holds true under the effects of imperfect communication and vehicle dynamics.

3.4 Evaluation

Simulation Approach

Based on the proposed control structure in Section 3.2 and tuning method in Section 3.3, CACCu are designed and evaluated in three scenarios, where the preceding connected vehicle is one, two, or three vehicles away from the ego vehicle. The human-driven vehicle trajectory data from Next Generation Simulation (NGSIM) [82] are adopted to construct the car-following scenarios for the evaluation. The NGSIM was launched by FHWA's Traffic Analysis Tools Program. It used high-resolution cameras to record trajectories of the vehicles on the real roads. The US Highway 101 (US 101) dataset was one dataset that reflected highway traffic condition. It contains the trajectories of vehicles within the 640-meter long study area during 45 minutes. Trajectories of consecutive vehicles which entered the study area at 0 min, 10 min, 20 min, 30 min, and 35 min were extracted to simulate the car-following scenarios under various congestion levels. The ego vehicle is then assumed to follow these vehicles.

Besides CACCu, there are two baseline high-level systems to be evaluated, while the low-level system remains the same. First as aforementioned, the ACC controller can be obtained by removing the feedforward term in CACCu, i.e., making $f_0\left(\ddot{x}_2(t-\theta_0)\right) = 0$ in (3.3). Then, an acceleration-based CCC [58] can be developed by replacing the feedforward filter with a constant feedback gain:

$$f_0\left(\ddot{x}_2(t-\theta_0)\right) = \gamma\ddot{x}_2(t-\theta_0-\sigma_2) \tag{3.14}$$

Where γ is the feedback gain for the acceleration signal from second preceding vehicle, and σ_0 is an intended delay for the acceleration feedback. The values of $\gamma = 0.5$ and $\sigma_0 = 0.6$ are recommended in the original design [58]. However, as the original CCC assumed different feedback configuration, γ_2 and σ_0 need to be re-tuned in this study to ensure a fair comparison. Using our definition of SSR, γ and σ_0 are adjusted to 0.42 and 0.65 respectively, for the highest probability to achieve head-to-tail string stability. γ and σ_0 can be further adjusted for the scenarios where the other connected vehicle is two or three vehicles away.

Finally, to simulate the behavior of ego vehicle more realistically, the vehicle dynamics are represented by the physics-based Audi A8 model provided by PreScan [83], rather than the simplified models in (3.4) and (3.5). PreScan, developed by The Netherlands Organization for Applied Scientific Research (TNO), is a dedicated simulation platform for Advanced Driver Assistance System (ADAS). It should be noted that the simplified model is still needed for the design of high-level control.

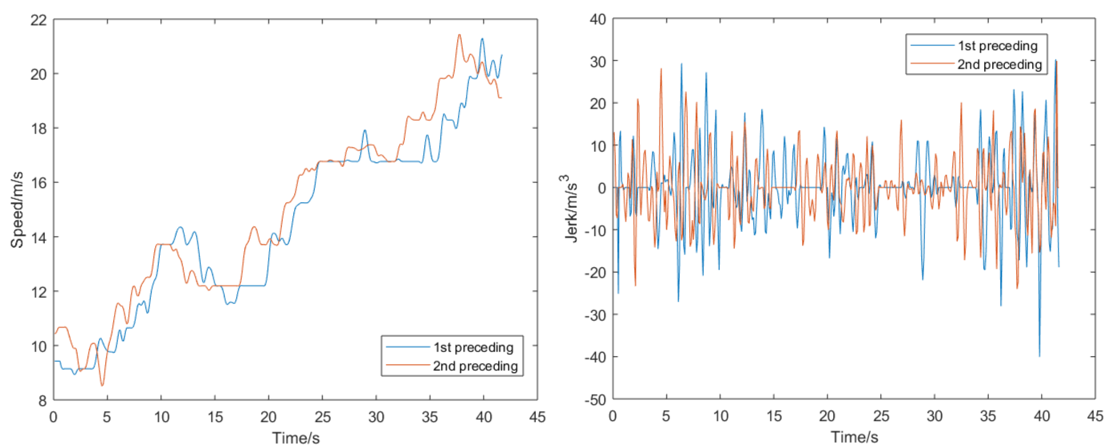
According to the trajectories of ego vehicle and its first preceding vehicle, the ego vehicle's performance can be determined. There are multiple measures of effectiveness (MOE) adopted in this study:

- String stability is measured by the count of speed overshoots (i.e., higher peak or lower valley values than the preceding vehicle's) during the ride;
- Safety/control accuracy is measured by spacing error of ego vehicle;
- Ride comfort is measured by the acceleration of ego vehicle, considering that they were commonly linked in previous research [45], [48];
- The fuel consumed by the ego vehicle is estimated using Virginia-Tech fuel consumption model [84].

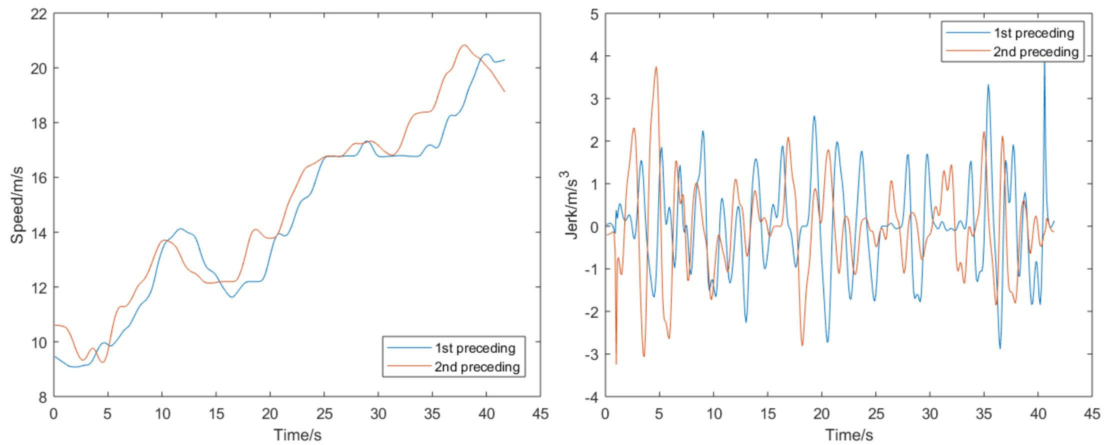
NGSIM data pre-processing

NGSIM trajectory data including position, speed, and acceleration profile of vehicles, among which the positions of vehicles were directly collected every 0.1s, while the speed and acceleration profiles of vehicles were derived from the position profiles. In the derivation of the speed and acceleration, the measurement error in position could be greatly propagated, leading to considerable noise in speed and acceleration profiles. It has been revealed that inconsistent speeds and unrealistic jerks (i.e., derivative of acceleration) can be frequently observed in the original NGSIM data, thus speed smoothing and recalculation of the acceleration is recommended before using the data [85].

In this study, the locally weighted scatterplot smoothing (LOWESS) is applied to the speed profiles of vehicles. The size of sliding window is chosen as 2s. Fig. 3.6 shows the speed and jerk profiles of a pair of preceding vehicles before/after speed smoothing as an example. It can be seen that there are many sudden jumps of the speed in the original profiles. In addition, the jerk exceeded 15m/s^3 for many times, which is mechanically unrealistic [85]. After smoothing, the speed profiles of vehicles are less noisy, and the jerks are always below 15m/s^3.



(a) Before smoothing



(b) After smoothing

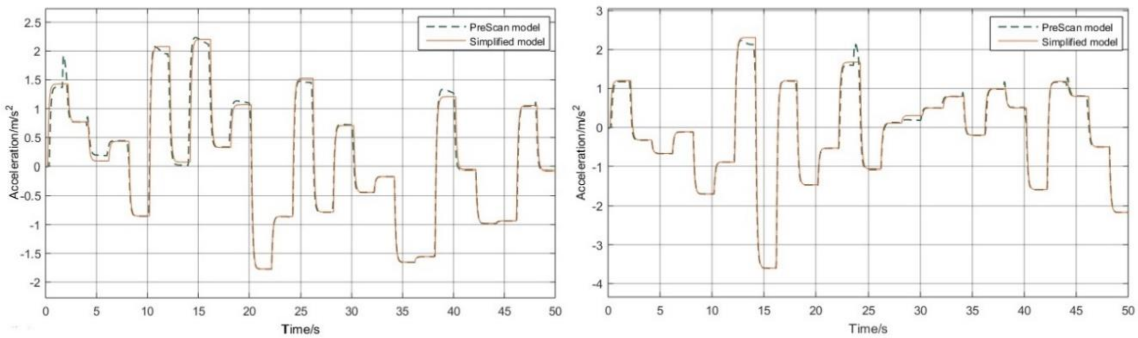
Fig. 3.6 Vehicle speed and jerk profiles before/after speed smoothing

Vehicle dynamics model

An Audi A8 sedan model from PreScan [83] plays as the ego vehicle in the evaluation. This physics-based vehicle model consists of engine, automatic gear box, 2-D chassis and other typical vehicle components. After the design of low-level control, the simplified vehicle dynamics model can be identified from the vehicle's response given a step acceleration command. MATLAB system identification toolbox is adopted to accomplish this identification. The identification result is:

$$G_0(s) = \frac{1}{s^2(1+0.12s)} e^{-0.2s} \tag{3.15}$$

Fig. 3.7 shows the acceleration/braking responses of the PreScan model and the identified simplified model given a series of step acceleration command with random magnitude, from the initial vehicle speed of and 10m/s and 20m/s respectively. It can be seen that the simplified model (3.15) approximates the PreScan model well in most of time, except when the Prescan model conducts gear shift which causes transient nonlinear response.



(a) Initial vehicle speed 10m/s (b) Initial vehicle speed 20m/s

Fig. 3.7. Comparison of the response of PreScan model and the simplified model

One-unconnected-vehicle case

1) Simulation setting

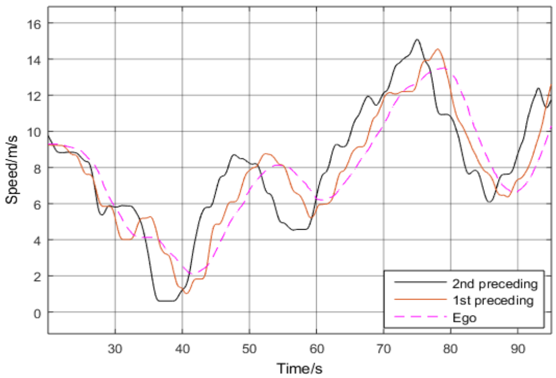
In the evaluation, the high control gains [0.3, 0.7] were adopted in all of CACCu, CCC and ACC. According to the identified vehicle dynamics (3.15) and Fig. 3.5 (b), a desired time gap of 1.1s should be sufficient for CACCu but apparently not for ACC and probably not for CCC (it is uncertain because human parameters of 1st preceding vehicle are unknown). However, to compare the performances of CACCu, CCC and ACC in the same situations, the desired time gaps for all three cases are set 1.1s. The sensor errors are modelled by normal distributions. The radar is assumed to be with 0.1m standard error on distance measurement and 0.1m/s on relative speed measurement [86]. The accelerometer on the 2nd preceding vehicle is assumed to have a standard error of 0.005m/s^2. GPS positioning is not needed. The communication delay is assumed to be 0.05s.

2) Simulation results

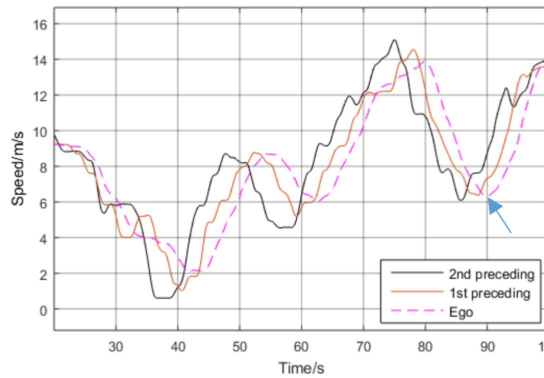
The results of the 5 simulation runs are summarized in Table 3.1. It can be seen that CACCu caused no speed overshooting in all of the cases. This means the speed perturbation from downstream was always attenuated by the ego vehicle, thus string stability was fulfilled. By contrast, ACC encountered speed overshoots for 6 times in total, which means string stability cannot be guaranteed by ACC. CCC also failed to avoid the speed overshooting in all the cases, but it had better chance to stay string-stable than stand-alone ACC. Fig. 3.8 shows the vehicle speed profiles in the case 30 min under different control types. It can be found that CACCu mitigated the speed oscillation all the time while CCC overshot once at 90s and ACC overshot twice at 55s and 90s, as labeled in Fig. 3.8(b) and (c).

TABLE 3.1
SUMMARY OF SIMULATION RESULTS IN ONE-UNCONNECTED-VEHICLE SCENARIO

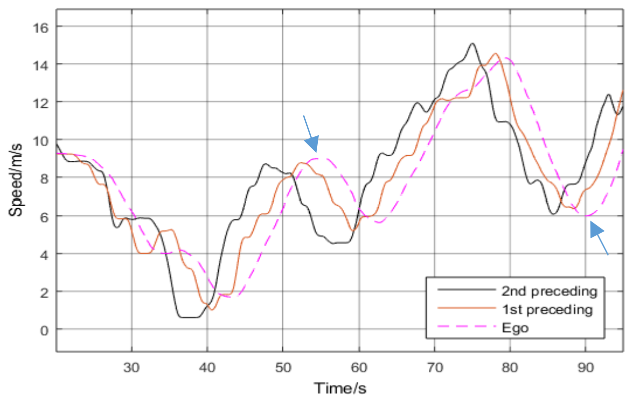
Entering time	Control type	# of speed overshootings	Acceleration peak (m/s^2)	Acceleration RMS (m/s^2)	Spacing error peak (m)	Spacing error RMS (m)	Fuel consumption (ml)
0 min	CACCu	0	0.86	0.42	1.37	0.76	47.70
	CCC	0	0.91	0.44	2.51	1.26	48.50
	ACC	1	0.94	0.47	3.33	1.71	50.00
10min	CACCu	0	0.79	0.43	1.72	0.81	31.30
	CCC	1	1.00	0.44	2.59	1.25	31.70
	ACC	1	0.99	0.45	3.33	1.58	32.70
20min	CACCu	0	0.87	0.37	1.92	0.80	30.50
	CCC	0	0.95	0.38	3.06	1.54	30.70
	ACC	0	0.95	0.38	3.80	1.60	31.30
30min	CACCu	0	1.13	0.53	3.16	1.54	39.20
	CCC	1	1.54	0.62	4.15	2.03	43.70
	ACC	2	1.47	0.67	4.96	2.47	46.50
40min	CACCu	0	1.32	0.48	2.16	0.77	23.80
	CCC	0	1.27	0.46	3.61	1.34	23.50
	ACC	2	1.40	0.50	4.38	1.67	26.10
Average reduction	From ACC	100%	13.2%	8.5%	48.7%	49.2%	7.2%
	From CCC	100%	11.5%	3.9%	36.1%	37.9%	2.5%



(a) CACCu



(b) CCC



(c) ACC

Fig. 3.8 The speed profiles of CACCu /CCC/ACC vehicle in the case of 30 min.

For the acceleration and spacing errors, both the peak value and RMS value are reported in Table 3.1. In average, CACCu reduced acceleration peak value by 13.2% and RSM value by 8.5% from those of ACC, and 11.5% and 3.9% from those of CCC, showing a moderate improvement in ride comfort. On the other hand, the spacing error peak value and RSM were greatly reduced by 48.7% and 49.2% from ACC, and 36.1% and 37.7% from CCC. This indicates that CACCu has a significantly better capability to maintain a safe inter-vehicle distance than ACC and CCC do. In addition, because of smaller acceleration and speed variation, CACCu achieved 7.2% and 2.5% fuel saving from ACC and CCC respectively.

Multiple-unconnected-vehicle case

1) Simulation setting

As aforementioned, to apply CACCu in the scenario where multiple unconnected vehicles are in between, the feedforward filter should include a combined transfer function of the multiple vehicles instead of single vehicle, that is, replacing (3.7) with:

$$F_0(S) = \frac{1}{G_0(s)\, H_0(s) s^2} T_1'(s) T_2'(s) \ldots T_n'(s) \tag{3.16}$$

Where n is the number of unconnected vehicles.

And the string stability determinant becomes:

$$SS = \|T_0(j\omega)\|_\infty = \left\| \frac{H_0(j\omega) G_0(j\omega) K_0(j\omega) + D_0(j\omega) \frac{T_1'(j\omega) \ldots T_n'(j\omega)}{T_1(j\omega) \ldots T_n(j\omega)}}{H_0(j\omega)(1 + H_0(j\omega) G_0(j\omega) K_0(j\omega))} \right\|_\infty \tag{3.17}$$

Accordingly, the calculation of SSR (12) should also be substituted by:

$$SSR = \int\int\int\int p(\alpha_1, \beta_1, \varphi_1, t_{1,h}) \ldots p(\alpha_n, \beta_n, \varphi_n, t_{n,h})\, \xi(SS) d\alpha_1 d\beta_1 d\varphi_1 dt_{1,h} \ldots d\alpha_n d\beta_n d\varphi_n dt_{n,h} \tag{3.18}$$

It is noted that the complexities of the SSR increase exponentially with the addition of the unconnected vehicles. This could bring computational issue in optimizing the parameters of $T_1'(s) T_2'(s) \ldots T_n'(s)$ for the highest SSR. If a full design of CACCu is unavailable, a simplification is to assume homogeneous traffic, i.e., all the unconnected vehicles have the same human parameters, which leads to:

$$F_0(S) = \frac{1}{G_0(s)\, H_0(s) s^2} T_1'(s)^n \tag{3.19}$$

With this simplification, an approximate SSR can be simply computed by (3.12).

In the case of two unconnected vehicles, both the full design with (3.16)-(3.18) and a simplified design of CACCu were evaluated. In full design, the optimal virtual preceding vehicles were determined to be $T_1'(s) = T_2'(s) = (1.22, 0.26, 0, 0.99)$. It corresponded to a critical gap of 1.3s and the maximum SSR of 97.8%. As comparison, the simplified design using (3.19) and (3.12) led to $T_1'(s) = (1.14, 0.4, 0, 0.95)$, corresponding to a maximum approximate SSR of 93.8% (computed by (3.12)). Meanwhile, the actual SSR was found to be 97.5% (computed by (3.18)). It is noted that although the simplified design has underestimated SSR, the obtained solution and its optimality (i.e., actual SSR) resembled the ones in full design.

2) Simulation results

As can be expected, these two designs of CACCu achieved very similar performances in the evaluation. For simplicity, the evaluation results with full design are reported in Table 3.1. The desired gap of 1.3s (critical gap) was used in all the runs. Overall, CACCu led to 83% speed overshooting avoidance, 8.2% acceleration reduction, 38% spacing error reduction and 4.7% fuel saving from ACC. It also achieved 67% speed overshooting avoidance, 5.8% acceleration reduction, 24.8% spacing error reduction and 2.3% fuel saving from CCC.

TABLE 3.2
SUMMARY OF SIMULATION RESULTS IN TWO-UNCONNECTED-VEHICLE SCENARIO

Entering time	Control type	# of speed overshootings	Acceleration peak (m/s²)	Acceleration RMS (m/s²)	Spacing error peak (m)	Spacing error RMS (m)	Fuel consumption (ml)
0 min	CACCu	0	0.72	0.32	1.89	0.96	36.10
	CCC	0	0.82	0.37	3.06	1.31	36.80
	ACC	0	0.87	0.38	3.04	1.39	37.00
10min	CACCu	0	0.57	0.34	1.74	0.76	30.30
	CCC	1	0.82	0.36	2.06	1.02	31.20
	ACC	1	0.76	0.35	2.50	1.36	31.20

20min	CACCu	0	1.04	0.39	3.38	1.10	32.30
	CCC	0	1.10	0.39	4.06	1.36	32.40
	ACC	0	1.12	0.39	4.47	1.64	32.30
30min	CACCu	1	1.03	0.42	3.28	1.57	29.70
	CCC	2	1.20	0.50	3.77	1.80	33.50
	ACC	5	1.19	0.54	4.24	2.14	35.80
40min	CACCu	0	0.93	0.35	1.14	0.56	17.40
	CCC	0	0.98	0.33	2.37	0.93	17.00
	ACC	0	0.88	0.35	2.92	1.26	18.00
Average reduction	From ACC	83.3%	11.4%	8.2%	35.2%	38.0%	4.7%
	From CCC	66.7%	13.5%	5.8%	27.1%	24.8%	2.3%

In the case of three unconnected vehicles, the full design of CACCu is infeasible because the required computation time was too long. Thus, only the simplified design was conducted. Given desired gap of 1.5s, the approximate SSR was maximized to 89.3%. Intuitively, the critical gap of CACCu in this scenario should be longer than 1.5s. However, a desired gap>1.5s means the loss of the throughput benefit over human driving which has an average desired gap of 1.5s [76]. Thus, 1.5s was assumed the maximum desired gap of ego vehicle and used in this scenario. The evaluation results for three-unconnected-vehicle scenario are summarized in Table 3.3. CACCu led to overall 60% speed overshooting avoidance, 6% acceleration reduction, 34.9% spacing error reduction and 3.3% fuel saving from ACC, and 60% speed overshootings avoidance, 4.8% acceleration reduction, 25.9% spacing error reduction and 1.5% fuel saving from CCC.

TABLE 3.3
SUMMARY OF SIMULATION RESULTS IN THREE-UNCONNECTED-VEHICLE SCENARIO

Entering time	Control type	# of speed overshootings	Acceleration peak (m/s^2)	Acceleration RMS (m/s^2)	Spacing error peak (m)	Spacing error RMS (m)	Fuel consumption (ml)
0 min	CACCu	0.00	0.66	0.34	1.71	0.87	37.90
	CCC	0.00	0.72	0.36	2.92	1.37	38.60
	ACC	0.00	0.85	0.37	3.33	1.62	38.00
10min	CACCu	0.00	1.03	0.36	1.89	0.89	26.90
	CCC	1.00	1.03	0.38	3.30	1.32	28.10
	ACC	1.00	1.05	0.37	4.00	1.60	27.90
20min	CACCu	0.00	1.01	0.43	3.37	1.52	39.00
	CCC	1.00	1.11	0.42	4.06	1.67	38.10
	ACC	1.00	1.10	0.42	4.00	1.91	38.60
30min	CACCu	0.00	0.98	0.40	3.52	1.76	26.60
	CCC	2.00	1.04	0.49	3.83	2.00	29.70
	ACC	2.00	1.05	0.51	4.39	2.17	30.80
40min	CACCu	2.00	1.08	0.34	1.81	0.79	17.70
	CCC	1.00	0.99	0.33	3.09	1.31	16.60
	ACC	1.00	0.97	0.34	3.17	1.42	17.70
Average reduction	From ACC	60.0%	5.6%	6.0%	36.0%	34.9%	3.3%
	From CCC	60.0%	2.8%	4.8%	30.1%	25.9%	1.5%

The benefits of CACCu over ACC in all the three scenarios are compared in Fig. 3.9. It shows a trend that more unconnected vehicles in between would make CACCu's benefits decline. This is expected because with more unmodelled noise in human behaviors being introduced, the information of the further preceding vehicle has weaker capability to predict the motion of 1st preceding vehicle. Nevertheless, CACCu still performed consistently better than ACC and CCC in every aspect. Generally speaking, the CACCu design described in Section II and III can be well extended to multi-unconnected-vehicle scenarios, although sometimes with approximation in determining the optimal parameters of "virtual preceding vehicles".

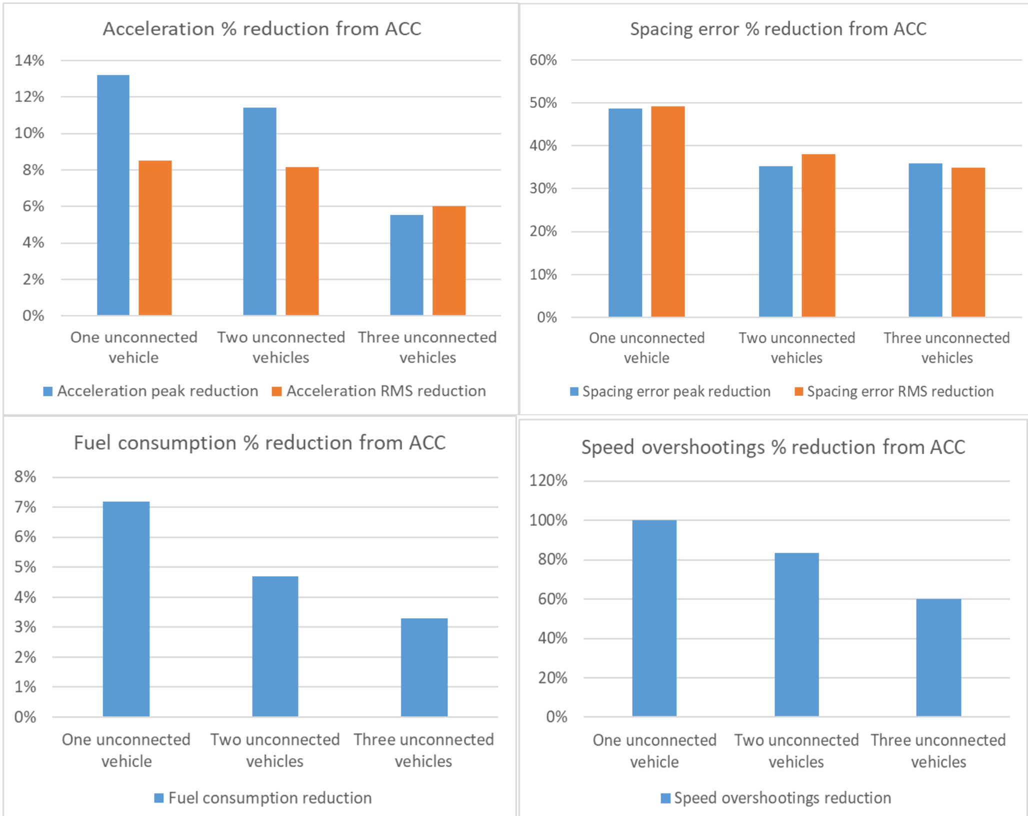


Fig. 3.9 Comparing the benefits of CACCu over ACC in all the three scenarios.

3.5 Key findings

In this chapter, CACC with unconnected (CACCu) algorithm was proposed. When encountering an unconnected preceding vehicle, CACCu can utilize the communication with the further (connected) preceding vehicle to improve the response of ego vehicle. It was analytically proven that by attaching a filter of "virtual preceding vehicle" to the original CACC feedforward filter, the CACCu vehicle can stay string-stable at a gap significantly shorter than that required by ACC. The high-fidelity evaluation results showed that CACCu avoided most of speed overshootings happening to ACC and CCC. This means the string stability was greatly improved. CACCu also led to overall 6~9% acceleration reduction, 35~49% spacing error reduction and 3~7% fuel saving from ACC; 5~8% acceleration reduction, 26~38% spacing error reduction and 2~3% fuel saving from CCC. These numbers indicated benefits of CACCu in safety, ride comfort and energy efficiency.

Chapter 4. Adaptive Model Predictive Control Approach for CACCu

In this chapter, an optimization-based control design of CACCu is proposed, to further improve the performance and usability of CACCu. Considering the essentially nonlinear time-variant and stochastic nature of CACCu system, Adaptive Model Predictive Control (A-MPC) approach was adopted. The performances of A-MPC CACCu, the previously developed CACCu (referred to as linear CACCu), ACC and human driving are compared via numerical simulations.

4.1 Problem statement

Although the preliminary control designs of CACCu indicated improvements over ACC, it will be faced with challenges commonly existing in rule-based linear control.

1) It cannot efficiently handle constraints, such as vehicle's capability of acceleration or jerk rate, speed limit, and other soft requirement for avoiding undesired situations such as harsh acceleration/brake or unsafe spacing;
2) The controller parameters are fixed at the robust values to ensure acceptable performance given a variety of human car-following behaviors, but it also means they have sacrificed their optimality. When encountering a "strange" driver whose behaviors are far from the average, such a controller will certainly lead to a continuous sub-optimal performance;
3) It still needs to estimate the number (i.e., n) of unconnected preceding vehicle vehicles for a proper choice of feedforward filter. In a loosely distributed traffic, such estimate can be difficult. For example, when the connected preceding vehicle ($(n+1)^{th}$ vehicle) is far away, it would be tricky to determine whether there are multiple unconnected vehicles in between, or it is just a plenty of empty space between the 1st preceding and n^{th} preceding vehicle. An incorrect estimate could lead to undesirable outcome.

To address these challenges and optimize the performance of CACCu, we propose Adaptive Model Predictive Control (A-MPC) approach. Model Predictive Control (MPC) is a prevailing nonlinear control framework that has been successfully applied to ACC [87] and CACC [88]. While MPC requires more tuning effort and computational cost, it showed advantages in handling constraints and achieving multiple control objectives (e.g., safety, fuel consumption and comfort). It is noted that the effectiveness of MPC relies on an accurate prediction model of the system dynamics, which is easy to obtain for deterministic systems like ACC and CACC, yet very difficult for time-variant stochastic systems like CACCu involving human drivers. To avoid unexpected behaviors caused by large modeling errors, the system dynamics model for MPC-based CACCu needs to be estimated and updated online according to actual situation, which leads to an adaptive MPC (A-MPC) [89]. A-MPC has essentially the same mechanism with normal MPC which optimizes a cost function in rolling time horizon under given constraints, except that in every time step a new prediction model is fed to the controller, instead of a fixed one. Thus, A-MPC is more suitable for nonlinear time-variant stochastic system.

The control schemes of CACCu based on A-MPC is shown in Fig. 4.1. An initial prediction model still needs to be derived from *a priori* knowledge in the population of drivers as a starting point of the control, but afterwards an online estimator will be generating new model parameters based on the actual motion of the preceding connected vehicle and the human-driven vehicle, and this estimation will be used to correct the initial/previous prediction model (through, e.g., a Kalman filter [90]) over time. Note that the A-MPC does not necessarily use the aforementioned linearized OVM (3.1) as prediction model, because OVM has been shown not a robust model to be estimated from a small piece of data [61].

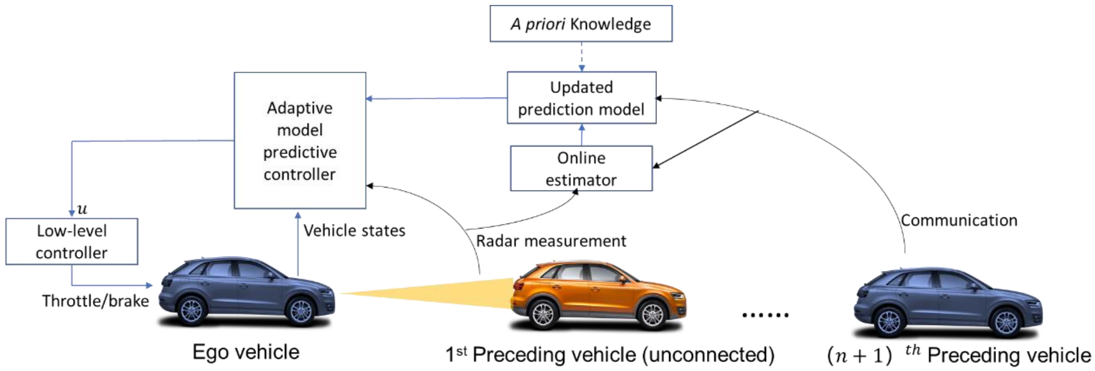


Fig. 4.1 Scheme of A-MPC-based CACCu

4.2 System modelling

A discrete-time state-space model is required for MPC design. The CACCu system can be modelled in two parts:

1) The dynamics between 1^{st} preceding vehicle and ego vehicle (i.e., how the state of ego vehicle is affected by the 1^{st} vehicle), which are relatively certain

2) The dynamics between $(n+1)^{th}$ vehicle and the 1^{st} vehicle (i.e., how the state of $(n+1)^{th}$ vehicle can help predict that of 1^{st} vehicle), which are relatively uncertain and need to be estimated online

Dynamics between 1^{st} vehicle and ego vehicle

As the control goals of CACCu are to maintain a constant time gap and a similar speed with preceding vehicle's, we define the state vector of vehicle 0 as follow:

$$X_0 = [e_p,\ e_v, a_0]^T \tag{4.1}$$

Where e_p and e_v are the spacing error and speed difference from the preceding vehicle, respectively; a_0 is the acceleration of vehicle 0. Apparently, we have:

$$e_p = h - v_0 t_h \quad (4.2)$$

$$e_v = v_1 - v_0 \quad (4.3)$$

Where h is the current inter-vehicle spacing, t_h is the desired time gap, v_1 and v_0 are the speed of 1st vehicle and ego vehicle, respectively. In addition, the acceleration of ego is ruled by the acceleration command with first-order lag, when ignoring the small actuator delay [87]:

$$\dot{a}_0 = -\frac{a_0}{\tau} + \frac{u}{\tau} \quad (4.4)$$

Where u is the acceleration command given by the controller, and τ is the response lag of the vehicle.

Combining (4.1) ~ (4.4), we can formulate a continuous-time state-space model with the inputs from ego and 1st vehicle:

$$\begin{aligned} \dot{X}_0 &= A_0 X_0 + B_0 U_0 \\ Y_0 &= C_0 X_0 + y_e \end{aligned} \quad (4.5)$$

Where:

$$X_0 = \begin{bmatrix} e_p \\ e_v \\ a_0 \end{bmatrix}, A_0 = \begin{bmatrix} 0 & 1 & -t_h \\ 0 & 0 & -1 \\ 0 & 0 & -1/\tau \end{bmatrix}, B_0 = \begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 1 \\ 1/\tau & 0 & 0 \end{bmatrix}, U_0 = \begin{bmatrix} u \\ a_1 \\ \Delta a_1 \end{bmatrix}$$

$$C_0 = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \end{bmatrix}, \text{ and } N_y = \begin{bmatrix} N_{e_p} \\ N_{e_v} \\ 0 \end{bmatrix}$$

Note that a_1 cannot be directly measured by the range sensor of vehicle 0. Instead, it should be predicted based on the acceleration of $(n+1)^{th}$ vehicle and then corrected by a state observer based on model (4.5). Therefore, Δa_1, the estimation error in a_1, is added as an unmeasured disturbance. N_y is the sensor error vector, consisting of spacing measurement error N_{e_p} and speed measurement error N_{e_v}. The ego vehicle's acceleration error is assumed insignificant.

This study assumes $\tau = 0.12$ and $t_h = 1.1s$ to keep the consistency with linear CACCu. The MPC controller is set to update its command every 0.1s. Then, the continuous-time model (4.5) is transformed to discrete-time model (4.6) using zero-order hold [38]:

$$\begin{aligned} X_0(t+0.1) &= A_0^d X_0(t) + B_0^d U_0(t) \\ Y_0(t) &= C_0 X_0(t) + y_e(t) \end{aligned} \tag{4.6}$$

Where:

$$A_0^d = \begin{bmatrix} 1 & 0.1 & -0.0784 \\ 0 & 1 & -0.0678 \\ 0 & 0 & 0.4346 \end{bmatrix} \text{ and } B_0^d = \begin{bmatrix} -0.0365 & 0.05 & 0.05 \\ -0.0321 & 0.1 & 0.1 \\ 0.5654 & 0 & 0 \end{bmatrix}$$

Estimation of dynamics between $(n+1)^{th}$ vehicle and 1st vehicle

Second-order linear models are the most commonly used in existing studies [40], [60], [91] for identifying car-following dynamics between a pair of vehicles (i.e., 1st and 2nd preceding vehicles). Although [60] further proposed that the order of the car-following model should increase by $2n$ when $n > 1$. However, this would again require the estimation of n and make the model estimation much more complicated. In fact, our preliminary work showed that a higher order of model does not necessarily improve the prediction accuracy of the model. Therefore, in this study, second-order model with input delay is adopted to represent the dynamics between $(\mathrm{n}+1)^{\mathrm{th}}$ vehicle and 1st vehicle, no matter what value of n.

As this model is to be estimated and updated at every time step, we write the second-order model in Autoregressive Exogenous (ARX) structure:

$$a_1(t)+c_1a_1(t-0.1)+\ c_2a_1(t-0.2)=c_3a_{n+1}(t-0.1k)$$

Which can then be converted to a discrete-time state-space model in general form:

$$\begin{bmatrix} a_1(t) \\ a_1(t-0.1) \end{bmatrix} = A_1 \begin{bmatrix} a_1(t-0.1) \\ a_1(t-0.2) \end{bmatrix} + B_1 a_{n+1}(t-0.1k) \tag{4.7}$$

Where a_1 and a_{n+1} are the accelerations of the 1st vehicle and $(n+1)^{th}$ vehicle, respectively. A_1, B_1 are coefficient matrices and k is the input time delay, all to be estimated. Specifically, we have:

$$A_1=\begin{bmatrix} -c_1 & -c_2 \\ 1 & 0 \end{bmatrix} \text{ and } B_1=\begin{bmatrix} c_3 \\ 0 \end{bmatrix}$$

For a linear identified model, we can replace a_1 and a_{n+1} with v_1 and v_{n+1} as the input/output data, because a_1 cannot be directly measured.

The recursive polynomial model estimator provided by MATLAB system identification toolbox is used for real-time model estimation. Kalman filter [90] is chosen as the estimation method. To estimate the input delay, we adopt a sweeping method:

1) Create multiple estimators;
2) No delay is assumed inside each estimator (to keep a consistent model structure over all the estimator);

3) Impose incremental time delays to the input signals of different estimators, as shown in Fig. 4.2;

4) At every time step, find the estimated model which indicates the least uncertainty in c_3 and also fulfill internal stability, i.e., find k that $min\ (Var(c_3)\,/c_3^2)$, s.t. the eigen values of A_1 have norms ≤ 1. c_3 is used here because this coefficient directly links system state to input signal, while c_1 and c_2 are responsible for internal state transition. In practice, we can observe that c_1 and c_2 converge to certain values no matter what input delay is assumed;

5) To prevent frequent switches between different estimators, a slack is added in determining k. We compared $Var(c_3(t))/c_3^2(t)$ of $k(t)$ with $Var(c_3(t))/c_3^2(t)$ of $k(t-0.1)$. If the difference is smaller than a slack (e.g., 10%), then keep $k(t) = k(t-0.1)$;

6) The time delay (i.e., $0.1k$) imposed to the input of selected estimator is considered the input delay for the estimated model.

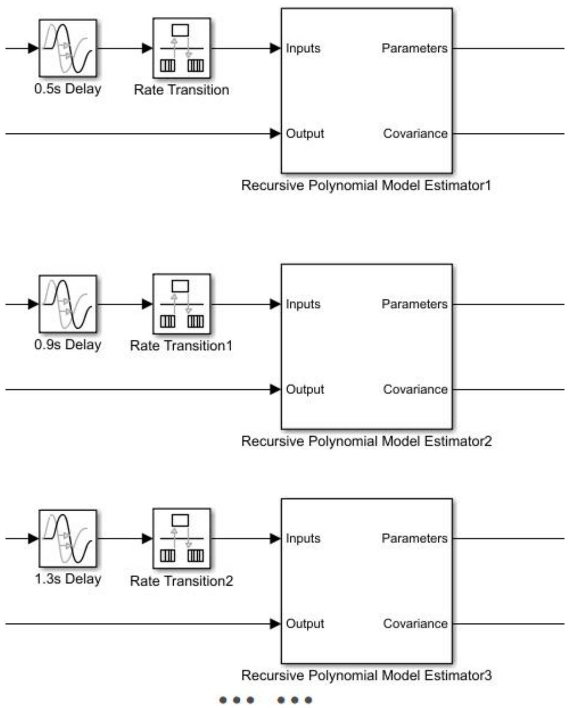


Fig. 4.2 Input delay identification with multiple estimators

In this study, the minimum input delay (i.e., minimum human reaction time) is assumed 0.5s [77]; 10 estimator were employed with incremental time delay of 0.4s, to cover a range of 0.5s ~ 4.5s input delay. For each estimator, the initial estimate of model parameters (c_1, c_2, c_3, k) are set at (-1.413, 0.437,0.03, 10), which represent typical dynamics between 1st and 2nd preceding vehicle shown in Fig. 3.7.

Complete system model

Combining (4.6) and (4.7), the model of the complete system can be obtained:

$$\begin{aligned} X(t+0.1) &= AX_0(t) + BU(t) \\ Y(t) &= CX(t) + y_e(t) \end{aligned} \quad (4.8)$$

Where $X(t) = \begin{bmatrix} e_p(t) \\ e_v(t) \\ a_0(t) \\ a_1(t) \\ a_1(t-0.1) \end{bmatrix}$, $A = \begin{bmatrix} 1 & 0.1 & -0.0785 & 0.005 & 0 \\ 0 & 1 & -0.0679 & 0.1 & 0 \\ 0 & 0 & 0.4346 & 0 & 0 \\ 0 & 0 & 0 & -c_1 & -c_2 \\ 0 & 0 & 0 & 1 & 0 \end{bmatrix}$,

$B = \begin{bmatrix} -0.0365 & 0.05 & 0.05 \\ -0.0321 & 0.1 & 0.1 \\ 0.5654 & 0 & 0 \\ 0 & 0 & c_3 \\ 0 & 0 & 0 \end{bmatrix}$, $U(t) = \begin{bmatrix} u(t) \\ a_{n+1}(t-0.1k) \\ \Delta a_1(t) \end{bmatrix}$,

$Y(t) = \begin{bmatrix} e_p(t) \\ e_v(t) \\ a_0(t) \end{bmatrix}$, and $C = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 \end{bmatrix}$.

c_1, c_2 and c_3 are updated every time step, while other parameters in matrices A and B are constant. Note that the second input signal in $U(t)$, i.e., a_{n+1}, should be delayed by $0.1k$ before given to MPC for state prediction.

4.3 Model predictive control

State estimation and prediction

Because the system state vector $X(t) = \left[e_p(t), e_v(t), a_0(t), a_1(t), a_1(t-1)\right]^T$ contains unmeasured states $(a_1(t), a_1(t-0.1))$, in every time step we need to estimate values of unmeasured states as the basis for predictions. A Kalman filter based on (4.6) is utilized as the state observer, combining both the prediction based on a_{n+1}, and the sensor measurements on e_p and e_v. This Kalman filter program is briefly described below:

$Innovation$: $E(t) = Y(t) - C_0\hat{X}(t|t-0.1)$

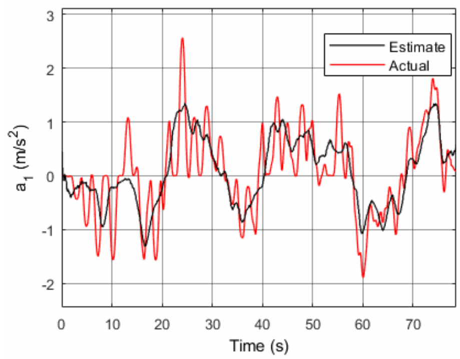
$Current\ state$: $\hat{X}(t|t) = \hat{X}(t|t-0.1) + ME(t)$

$$Prediction: \quad \hat{X}(t+0.1|t) = A\hat{X}(t|t-0.1) + B_u u(t) + B_v a_{n+1}(t-0.1k) + LE(t)$$

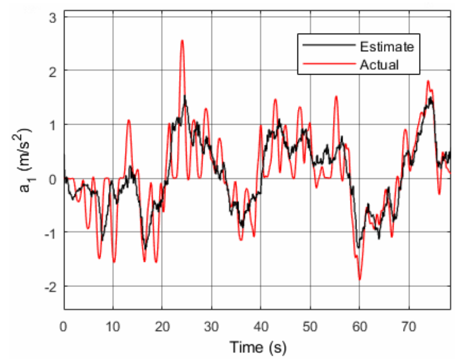
Where B_u and B_v are the first and second columns of matrix B, respectively; $E(t)$ is the innovation, i.e., the discrepancy between the prediction and measurement. L and M are Kalman gain matrices linked to the variances of unmeasured disturbance (Δa_1) and measurement noises (N_{e_p} and N_{e_v}) [92]. These noises are assumed to be unbiased white noises with constant variances.

$Var(N_{e_p})$ and $Var(N_{e_v})$ can be set compliant with the actual radar sensor. In this study, we assumed $Var\left(N_{e_p}\right) = 0.029$ and $Var\left(N_{e_v}\right) = 0.017$ from field test [34]. These noises will also be applied to the later evaluation. However, higher measurement accuracy may be achieved with state-of-art radar sensor, such as $Var\left(N_{e_p}\right) = 0.01$ and $Var\left(N_{e_v}\right) = 0.01$ of Bosch long-range radar (LLR) [86].

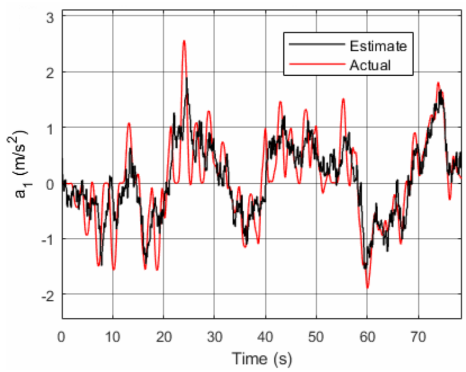
Different values of $Var(\Delta a_1)$ were tried in a state estimation test. Assuming a higher $Var(\Delta a_1)$ tends to make the estimated a_1 closer to the truth, but also more sensitive to measurement noise, leading to jerky estimation. Fig. 4.3 shows the estimation results of $a_0(t)$ assuming different $Var(\Delta a_1)$, during a simulation run in the later section Performance Evaluation. It was found that $Var(\Delta a_1) = 1$ achieved a good estimation accuracy without excessive noise.



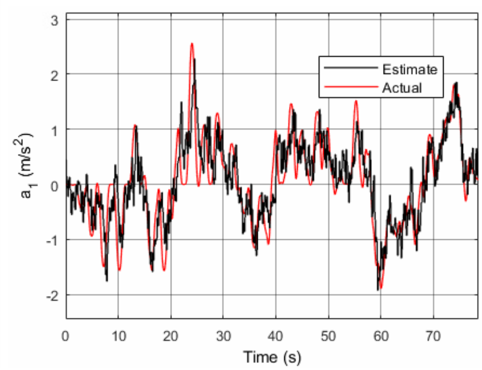
(a) $Var(\Delta a_1) = 0$



(b) $Var(\Delta a_1) = 0.5$



(c) $Var(\Delta a_1) = 1$



(d) $Var(\Delta a_1) = 2$

Fig. 4.3 The estimated a_0 when assuming different $Var(\Delta a_1)$ in a simulation run

To predict the system state in the future time using (4.8), the MPC controller accepts preview of the measured disturbance (i.e., a_{n+1}). As (4.8) indicates that the current system state is affected by the acceleration of $(n+1)^{th}$ vehicle only up to k steps ago, the MPC controller can preview it for the next k time steps. If the prediction horizon N is longer than k, then the a_{n+1} is assumed unchanged in the last $(N-k)$ time steps. Therefore, the review signal reads:

$$a_{n+1}(t-0.1j \mid t) = \begin{cases} a_{n+1}(t-0.1j), & 0 \le j \le k \\ a_{n+1}(t), & k < j \le N \end{cases} \tag{4.9}$$

Rolling-horizon optimization

MPC solves a rolling-horizon optimization problem at every time step (0.1s). The objective function is defined as to reduce spacing error, speed difference form preceding vehicle, and acceleration of ego vehicle (which links to fuel consumption):

$$Min \sum_{j=1}^{N} q_p e_p^2(t+0.1j|t) + q_v e_v^2(t+0.1j|t) + q_a a_0^2(t+0.1j|t) + q_u u^2(t+0.1j|t) + \rho\varepsilon^2 \quad (4.10)$$

Where N is the prediction horizon; q_p, q_v, q_a and q_u are weights for e_p, e_v, a_o and u; $\rho\varepsilon^2$ is the penalty term when there are constraints being violated; ε is the slack variable for the constraints, and ρ is the penalty factor.

Constraints are set as below:

- To guarantee safety, e_p should be constrained:

$$e_{pmin} - \varepsilon \leq e_p \leq e_{pmax} + \varepsilon$$

- To consider physical limits on vehicle dynamics and ride comfort, the control signal u and its increment rate should be constrained:

$$u_{min} - \varepsilon \leq u \leq u_{max} + \varepsilon$$

$$u'_{min} - \varepsilon \leq u' \leq u'_{max} + \varepsilon$$

- Other constraints (such as speed limit) can also be added when necessary.

As the optimization adopts a linear system model, linear constraints and the objective function in quadratic form, the optimization can be solved in finite steps by a standard Quadratic Programming (QP) solver (e.g., KWIK algorithm [93]).

4.4 Performance evaluation

Simulation settings

To comprehensively compare the performances of the A-MPC-based CACCu and the previously proposed linear CACCu, the entire NGSIM data from the US101 innermost lane (with the fewest lane-changes) are used to simulate various car-following scenarios. For meaningful evaluations, the period of each used scenario is required to be longer than 60s.

A fair comparison between A-MPC-based CACCu and the linear CACCu would require their parameters to be tuned for the similar trade-off between control accuracy and energy consumption. Although MPC is a nonlinear control method and the control law cannot be expressed as a state feedback, we approximate it as a Linear-Quadratic Regulator (LQR), whose feedback gains can be explicitly derived from Racatti equations [94]. As we have known the state feedback gains in linear CACCu are $K = [-0.3 \ \ -0.7 \ \ 0.77]$, the weights $[q_p \ q_v \ q_a \ q_u]$ in (4.10) were tuned to [1 3 3 9] so that the following LQR has the similar state feedback gains with linear CACCu:

$$Min \sum_{t=0}^{\infty} q_p e_p^2(t) + q_v e_v^2(t) + q_a a_0^2(t) + q_u u^2(t) \tag{4.11}$$

All the parameters used in the MPC controller are listed in Table 4.1.

TABLE 4.1
SUMMARY OF MPC CONTROLLER PARAMETERS

Parameter	Value	Parameter	Value
$\boldsymbol{q_p}$	1	e_{pmin}	-2m
$\boldsymbol{q_v}$	3	e_{pmax}	2m
$\boldsymbol{q_a}$	3	u_{min}	-2
$\boldsymbol{q_u}$	9	u_{max}	1
N	30	u'_{min}	-3

$\boldsymbol{\rho}$	10000	u'_{max}	3

Evaluation results

Firstly, the evaluation was conducted in the scenarios of $n = 1$, i.e., the three-vehicle-sandwich scenarios. 380 such scenarios were extracted from the NGSIM dataset. The average performances of A-MPC CACCu, linear CACC over the 380 runs are listed in Table 4.2. The fuel consumption of ego vehicle is estimated using VT-micro model [84]. The performance of ACC and the actual human driver collected by NGSIM are also provided for reference.

TABLE 4.2
EVALUATION RESULTS IN THE SCENARIOS OF $n = 1$

Control method	**Spacing error RMS (m)**	**Acceleration RMS (m/s^2)**	**Fuel consumption (ml)**
A-MPC CACCu	0.90	0.63	38.15
Linear CACCu	0.96	0.60	38.15
ACC	1.79	0.67	41.38
Human driver	N/A	0.89	47.50

It can be seen that A-MPC CACCu has slightly better control accuracy (spacing error) than linear CACCu, while the fuel consumptions are almost the same. Both CACCu methods outperformed ACC in all the aspects, especially in spacing error. The human driver is found to cause much higher acceleration and fuel consumption than any of the CACCu/ACC.

The performance evaluations were further conducted in 322 scenarios of $n = 2$ and 50 scenarios $n = 3$. The results are listed in Table 4.3 and 5.4, respectively. For CACCu methods, the values of n were assumed to be unknown in each run. However, the performances of linear CACCu knowing the values of n are also shown (referred as Linear CACCu-ideal). They can be seen as the ideal performances of linear CACCu.

TABLE 4.3
EVALUATION RESULTS IN THE SCENARIOS OF $n = 2$

Control method	Spacing error RMS (m)	Acceleration RMS (m/s^2)	Fuel consumption (ml)
A-MPC CACCu	1.07	0.65	36.67
Linear CACCu	1.69	0.61	36.44
Linear CACCu-ideal	1.20	0.61	36.17
ACC	1.82	0.68	39.01
Human driver	N/A	0.90	44.63

TABLE 4.4
EVALUATION RESULTS IN THE SCENARIOS OF $n = 3$

Control method	Spacing error RMS (m)	Acceleration RMS (m/s2)	Fuel consumption (ml)
A-MPC CACCu	1.18	0.70	36.82
Linear CACCu	2.26	0.69	37.65
Linear CACCu-ideal	1.39	0.66	36.21
ACC	1.95	0.72	38.81
Human driver	N/A	0.91	43.01

It can be seen that A-MPC greatly outperformed linear CACCu when the value of n is unknow to them. The spacing error of linear CACCu is 57% and 88% higher than A-MPC CACCu in the case of $n = 2$ and 3, respectively. Even when the value of n is accurately known to the linear CACCu, there was no apparent winner between the A-MPC CACCu and the linear CACCu in its ideal condition. A-MPC CACCu achieved slightly smaller spacing error but also costed slightly higher fuel. However, considering that A-MPC does not require the n to be known while the performances of CACCu degrades greatly if n is unknow, A-MPC CACCu is apparently a more robust method for the implementation in complicated traffic situations.

4.5 Key findings

This chapter presented the Adaptive Model Predictive Control (A-MPC) approach for CACCu. The simulation with NGSIM data showed that A-MPC CACCu slightly outperformed the linear CACCu when n was accurately known, and largely outperformed linear CACCu when

n was unknown. Therefore, A-MPC CACCu was proven a more robust method for the implementation in complicated traffic situations

Chapter 5. Field Demonstration of CACCu

The proposed CACCu needs to be validated in the field to ensure that it can be implemented to real world. This chapter presents the experiment design, algorithm realization and demonstration results of the linear CACCu. It was confirmed that CACCu can greatly attenuate the traffic disturbance and improve safety, comfort, and fuel efficiency compared to ACC.

5.1 Experimental vehicles

The experiment was conducted through the collaboration with Korea Advanced Institute of Science and Technology (KAIST). The two test vehicles, Hyundai i30 PD and Hyundai Veloster, used in the test are shown in Fig. 5.1. Hyundai i30 PD played as the ego vehicle and Hyundai played as preceding vehicle (s), as elaborated later in the Section Experiment design. Both vehicles have been automated with onboard sensors and electric pedal actuators. The V2V communications between them have been enabled by Wi-Fi modules, which are set to transmit the information every 0.1s.

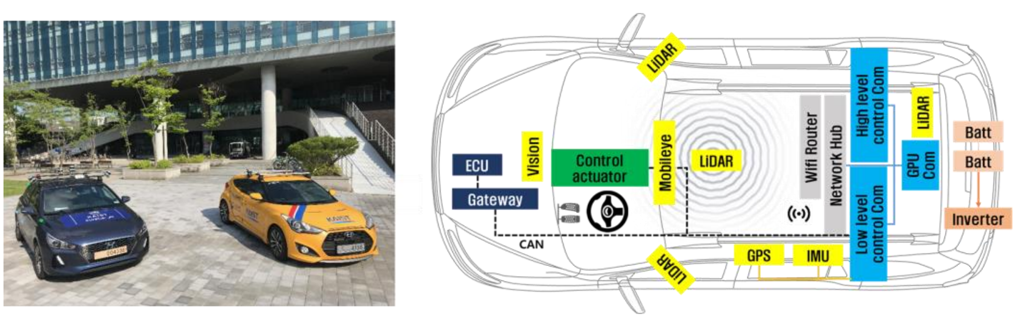


Fig. 5.1 Autonomous vehicle i30 PD(blue)/Veloster(yellow) and System Configure(right)

The long-range radar sensor, which has been commonly adopted by commercial ACC, is not available in these vehicles. Instead, there are two other options for front-view sensing: Lidar and Mobileye (camera). To facilitate the selection of the appropriate sensor that meets the needs, a sensing accuracies experiment was conducted. The distance/speed relative to preceding vehicle, measured by Lidar and Mobileye during an arbitrary run are shown in Fig. 5.2. The real-time-kinetic (RTK) GPS, which is of centimeter accuracy, served as the ground truth. The result showed that the lidar sensor had a detection limit distance of 30m. The reason is that the density of the point-cloud cluster gets too low as the distance increases. It was seen that the Mobileye estimated the depth of the preceding vehicle very well. The root mean square (RMS) values of both sensing errors were found to be 1.10 m for Mobileye and 4.18 m for Lidar. Therefore, Mobileye was chosen to detect the preceding vehicle for CACCu and ACC. The RMS error of estimated velocity by Mobileye was found to be 3.46km/h, with the ground truth provided by CAN bus of preceding vehicle. This is however a notable sensing error compared to state-of-art radar sensors [86].

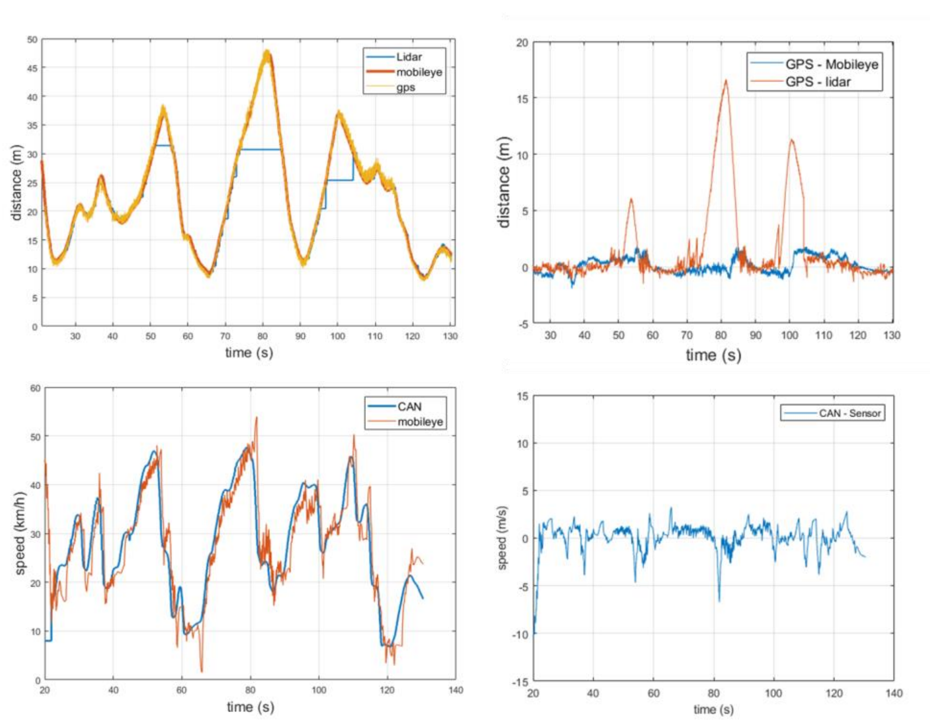


Fig. 5.2 Sensing results (left top/bottom) and sensing error (right top/bottom)

To properly parameterize the feedforward filter in CACCu controller, the longitudinal vehicle dynamics of ego vehicle should be identified. Different from the setting in Chapter 4, the experimental vehicles only accept speed command instead of acceleration command. Another major challenge in this study is that we have neither low-level information (e.g., engine torque map) of the vehicle nor the access to revise its low-level controller. Therefore, the original CACCu high-level controller must be adapted to give speed command, as described in next section. For the same reason, the "vehicle dynamics" now refers to the relationship between the commanded speed and the actual speed. A vehicle dynamics test was conducted for the vehicle dynamics identification. Like in [45], a small step signal (which caused no acceleration saturation) were given as the commanded speed in the test, and the actual speed response of the vehicle was recorded. Using MATLAB system identification toolbox, a second-order model $G_0(s)$ was identified from the collected data:

$$G_0(s) = \frac{L(\dot{x}_0)}{L(v_c)} = \frac{1}{0.8s^2+1.6s+1} e^{-0.7s} \tag{5.1}$$

Where $\dot{x}_0$ and v_c are the speeds of ego vehicle and the commanded speed, respectively. $L(*)$ denotes Laplace transform and s is the Laplace variable. The model response and the actual response of the vehicle are compared in Fig. 5.3.

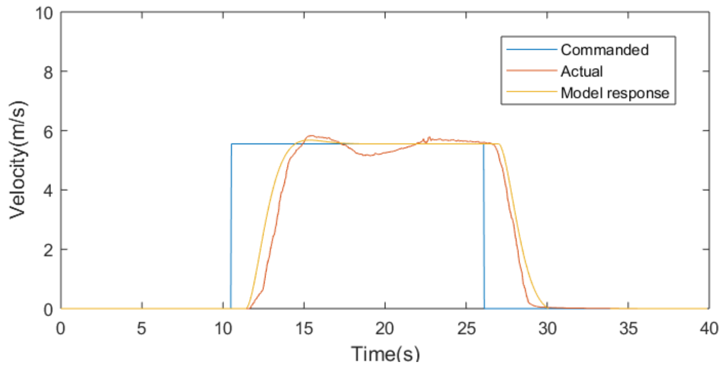


Fig. 5.3 The model response and actual vehicle response during the vehicle dynamics test

5.2 Control algorithm adaptation

To facilitate the speed control in CACCu vehicle, the high-level control law is revised to as below:

$$\begin{aligned} &h_0(t) = x_1(t) - x_0(t) - l_1 \\ &h_{0,d}(t) = t_{0,h}\dot{x}_0(t) + h_{0,st} \\ &e_0(t) = h_0(t) - h_{0,d}(t) \\ &v_c = \dot{x}_0 + k_p e_0 + k_d \dot{e}_0 + f(\ddot{x}_2) \end{aligned} \tag{5.2}$$

Where $x_0(t)$ and $x_1(t)$ are the locations of the ego and 1st preceding vehicle, h_0 is the spacing between the ego and 1st preceding vehicle, with l_1 being the length of the 1st preceding vehicle, $h_{0,d}(t)$ is the desired spacing, $h_{0,st}$ is the standstill spacing, $t_{0,d}$ is the desired time gap, $e_0(t)$ is the spacing error, $k_{0,p}$ and $k_{0,d}$ are the gains of the proportional-derivative (PD) feedback controller, and $f(\ddot{x}_2)$ is the feedforward signal based on the acceleration of 2nd preceding vehicle's acceleration.

Taking Laplace transform of (5.2) and combining it with (5.1), we have:

$$\frac{sX_0}{G} = K(X_1 - HX_0) + sX_0 + Fs^2X_2 \tag{5.3}$$

Where

$F(s) = L(f)$

$K(s) = k_p + k_d s$

$H_0(s) = 1 + t_{0,h} s$

Similar to Control Design in Chapter 4, an ideal feedforward filter $F(s)$ can be derived from (5.3) by making the spacing error $X_1 - HX_0 = 0$:

$$F = \frac{1-G}{sHG} T_1' \tag{5.4}$$

Where $T_1'(s)$ is the transfer function of "virtual preceding vehicle" in the form of OVM:

$$T_1'(s) = T_1'(\alpha_1', \beta_1', \varphi_1', t_{1,h}', s) \tag{5.5}$$

This study adopts controller parameters $k_p = 0.5$, $k_d = 1$ and $t_{0,h} = 1.5$, which shows a balance between control accuracy and smoothness in simulations. Then, $T_1'(s)$ is determined to be $T_1'(1.12, 0.21, 0, 1.62, s)$ by maximizing the string stable ratio (SSR), following the steps in String Stability Analysis, Chapter 4.

5.3 Experiment Design

This field test will be focused on the three-vehicle sandwich scenario (with one unconnected vehicle in between of two connected vehicles), due to the highest probability to occur among the mixed platooning scenarios. ACC and human driving served as performance baseline.

To ensure fair comparisons between CACCu, ACC and human driving, it is required that two preceding vehicles must drive identically each time when testing different control methods. However, a difficulty is that the 1st preceding vehicle is supposed to be a human-driven vehicle while it is almost impossible for the human driver to follow the test path the same way as before. Therefore, we proposed to use the existing NGSIM data to reconstruct the real-traffic scenario. In every test scenario, the NGSIM trajectories of two consecutive vehicles (i.e., 1st and 2nd preceding vehicles) are extracted. As shown in Fig. 5.4, the Hyundai Veloster is set in automated mode instead of manual model. The speed profile of 1st preceding vehicle from NGSIM is given to the Veloster as speed command over time, so that the movement of 1st preceding vehicle can be replicated consistently. In the meanwhile, the Veloster is also responsible for imitating the communications from the 2nd preceding vehicle to ego vehicle. As the ego vehicle does not need to sense the 2nd preceding vehicle and the trajectories of both preceding vehicles are fixed, there is no need to physically add a 2nd preceding vehicle to the test. An easier but equivalent way is making the Veloster send the information of 2nd preceding vehicle to the ego vehicle. Then the Hyundai i30 (ego vehicle) is driven in CACCu/ACC/manual mode following the Veloster.

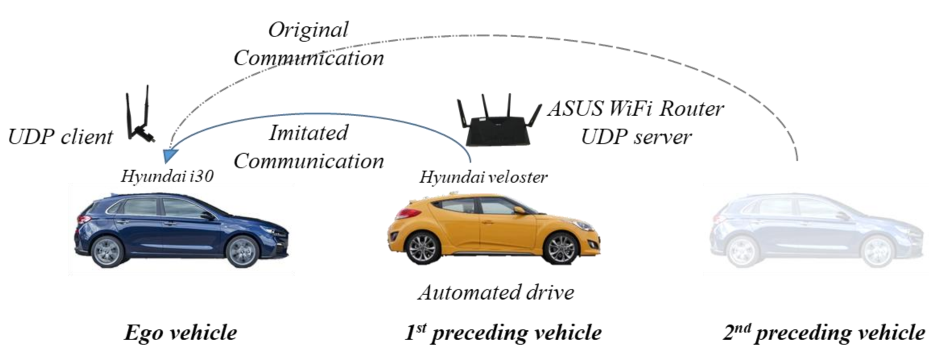
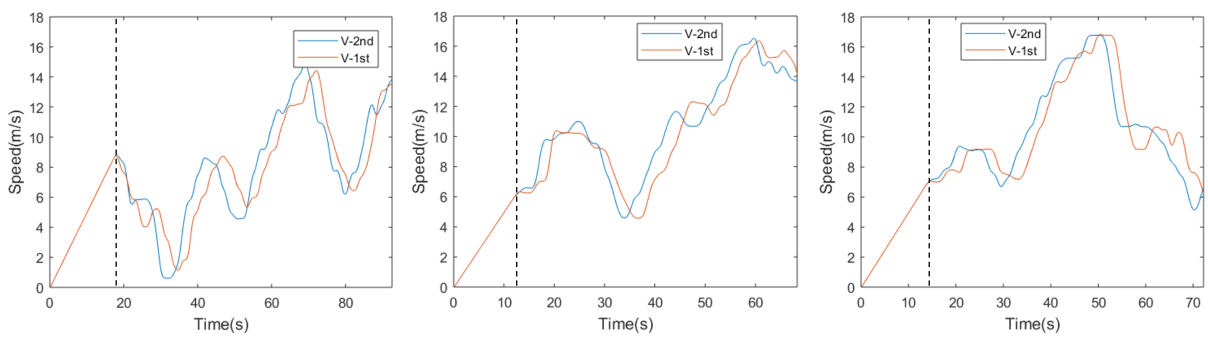


Fig. 5.4 Re-producing three-vehicle-sandwich scenario

Three test scenarios are created with NGSIM data. The speed profiles of the 1st and 2nd preceding vehicles in these scenarios are shown in Fig. 5.5. Note that a moderate acceleration period of $0.5m/s^2$ has been added to the beginning of each NGSIM vehicle speed profile so that the test vehicle can smoothly reach the starting-point speed from the rest. The performance of ego vehicle during this start-up period are not taken into the results. To disperse the random effects of vehicle dynamics nonlinearity, each control method was tested twice for every scenario.



(a) Scenario 1 (b) Scenario 2 (c) Scenario 3

Fig. 5.5 The speed profiles of the preceding vehicles in test scenarios

5.4 Results

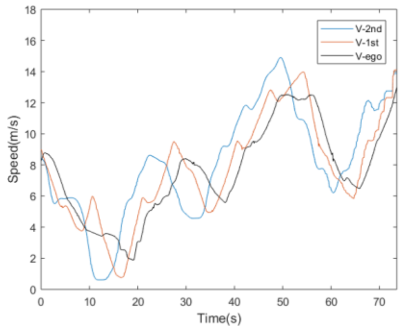
Performances of CACCu/ACC/human driving are summarized in Table 5.1. The acceleration and spacing error of ego vehicle were collected as measures of comfort and safety performances, respectively. Because the fuel consumption in each run was not obtainable, the VT-micro model was used to estimate the fuel consumption based on the vehicle trajectory. Based on the results, the CACCu consistently outperformed ACC and human driver. In average, CACCu reduced 10.82% acceleration RMS, 60.79% spacing error RMS and 6.24% fuel consumption from ACC's. Compared with human driving, CACCu reduced 17.64% acceleration and 13.43% fuel consumption. These benefits are even greater than the simulation results shown in Chapters 4 and 5.

TABLE 5.1
SUMMARY OF TEST RESULTS OF CACCU/ACC/HUMAN DRIVING

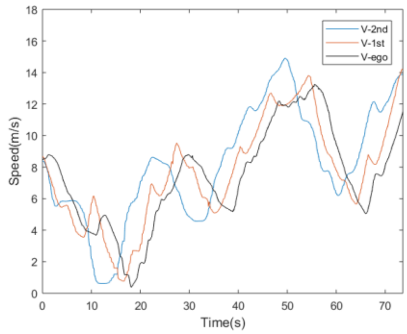
Test number	Control type	Acceleration RMS(m/s)	Spacing error RMS(m)	Fuel consumption (ml)
1.1	CACCu	0.72	1.86	41.80
	ACC	0.85	3.48	42.80
	Human	0.84	N/A	45.50
1.2	CACCu	0.77	1.95	41.70
	ACC	0.87	5.44	41.60
	Human	0.95	N/A	48.90
2.1	CACCu	0.70	2.03	43.10
	ACC	0.75	5.42	43.60
	Human	0.77	N/A	46.80
2.2	CACCu	0.74	1.97	42.50
	ACC	0.74	5.24	44.20
	Human	0.90	N/A	48.40
3.1	CACCu	0.61	2.33	37.30
	ACC	0.86	5.60	48.20
	Human	0.85	N/A	46.20
3.2	CACCu	0.81	1.97	43.00
	ACC	0.80	5.70	45.60
	Human	0.97	N/A	52.30
Average	CACCu	0.72	2.02	41.57
	ACC	0.81	5.15	44.33
	Human	0.88	N/A	48.02
Reduction	From ACC (%)	10.82%	60.79%	6.24%
	From Human (%)	17.64%	N/A	13.43%

The vehicle speed profiles of CACCu/ACC/human driving in test 1.1, 2.1 and 3.1 are displayed in Fig. 5.6. It can be seen that CACCu greatly attenuated the speed fluctuations, while ACC and human driving tended to amplify them. The speed overshootings happening to ACC and human driving were entirely avoided by CACCu. This indicates the improved string stability and explains the smaller acceleration and fuel consumption of CACCu.

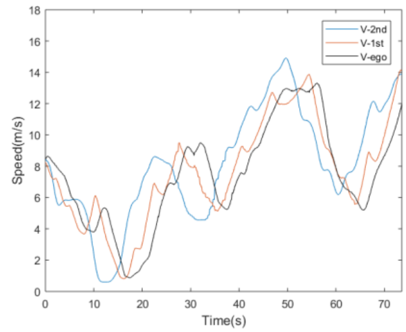
It is also noted that the speed profiles of 1st preceding vehicle were almost identical in the test of CACCu, ACC and human driving, and close to the original NGSIM data shown in Fig. 5.5. Thus, the effectiveness of the experiment design is verified.



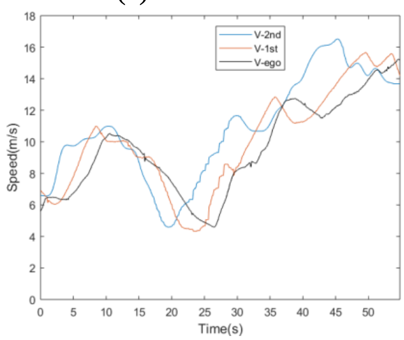
(a) CACCu in test 1.1



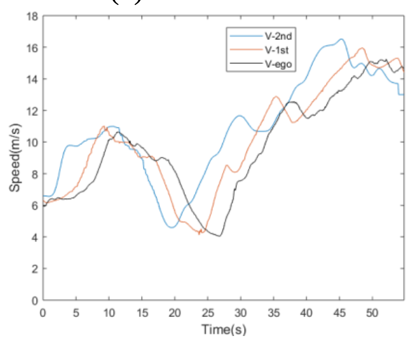
(b) ACC in test 1.1



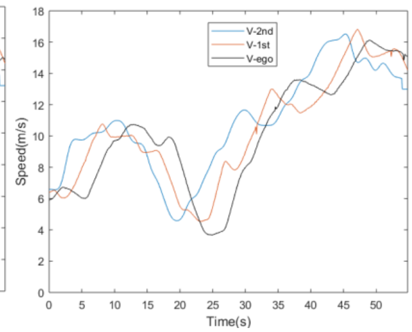
(c) Human driving in test 1.1



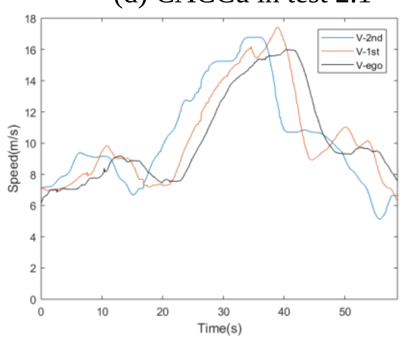
(d) CACCu in test 2.1



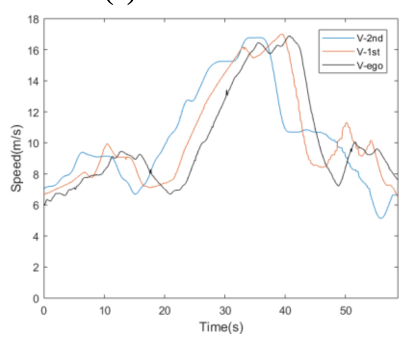
(e) ACC in test 2.1



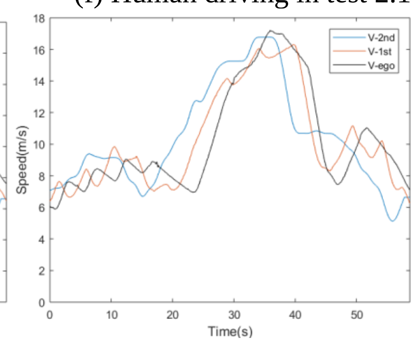
(f) Human driving in test 2.1



(g) CACCu in test 3.1



(h) ACC in test 3.1



(i) Human driving in test 3.1

Fig. 5.6 The vehicle speed profiles in the tests

5.5 Key findings

To test CACCu in the field, this chapter adapted the original CACCu algorithm, making it suitable to be implemented in our test vehicles which only accepts speed command. Then, the experiment was designed to replicate the car-following scenarios with NGSIM data, so that different control methods can be compared fairly. The field test verified the great benefits of CACCu in terms of string stability, safety, comfort and fuel saving.

Chapter 6. Human-in-the-loop CACC (hCACC)

In this chapter, a human-in-the-loop CACC algorithm (hCACC) is proposed with the goal of helping human drivers stabilize vehicles efficiently and safely. Compared to the existing human-in the-loop CCC, the hCACC inherits from CACC the feedback-feedforward control structure and zero-spacing-error rule in control design, instead of simply offering a proportion of preceding vehicle' acceleration. Besides, hCACC would bear the following features:

- Utilizing both the speed and acceleration information of preceding vehicle in pursuit of the best performance
- Taking into consideration of the potential effect of hCACC on the human's behavior
- Less driving load on human
- Less fluctuation in both speed and headway

The effectiveness of hCACC is to be shown by high-fidelity simulations using physics-based vehicle model, real-world vehicle trajectory data, and driving simulator with real human.

6.1 Control Design

The scheme of hCACC system is shown in Fig. 6.1. The ego vehicle is assumed to be connected but not automated, or automated but driven in manual mode. When the ego vehicle is following another CV, the human driver can choose to turn on hCACC and co-pilot the vehicle. On the human side, the driver is still responsible of monitoring the preceding vehicle and giving input to throttle/brake pedals. On the hCACC side, extra acceleration (on top of human actions) would be imposed on the ego vehicle to assist the human driver. Due to the nonlinearity of vehicle dynamics, bi-level control design is needed. The high-level controller decides the desired extra acceleration according to the received information of preceding vehicle, and the low-level controller determines how to adjust the throttle and brake to achieve this extra acceleration. The final inputs to the ego vehicle would be the summation of human's on-pedal throttle/brake and the adjustment made by low-level controller.

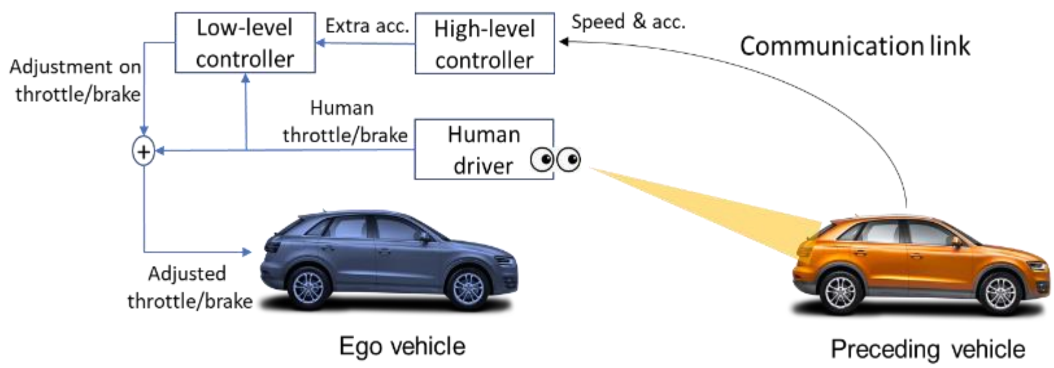


Fig. 6.1 Framework of hCACC system

The human behavior is still modelled by OVM with stochastic parameters, as has been described in Chapter 4. The other two key components of hCACC design, i.e., designs of high-level and low-level controls, are described respectively in the rest of this section.

High-level control

Since human driving has already included feedback control in terms of spacing and speed, a natural approach is giving an additional acceleration feedforward to the ego vehicle (which is similar to upgrading ACC to CACC). Besides, as human's feedback control has long delay, it will be desirable to have an automatic speed feedback which can more timely capture the speed difference from the preceding vehicle. Therefore, the car-following behavior of hCACC vehicle is decided as:

$$h(t) = x_2(t) - x_1(t) - l$$
$$\ddot{x}_1(t) = \alpha\left(\frac{1}{t_h}h(t-\varphi) - \dot{x}_1(t-\varphi)\right) + \beta\dot{h}(t-\varphi) + g(u) \tag{6.1}$$
$$u = \beta'(x_2(t-\theta) - x_1(t)) + f(\ddot{x}_2)$$

Where φ, α, β and t_h are human parameters, u is the acceleration command made by the high-level controller; $g(u)$ represents the actual acceleration achieved by the longitudinal vehicle dynamics; β' is the control gain for the automatic speed feedback; $f(\ddot{x}_2)$ denotes a linear feedforward filter that generates commands based on the received acceleration of preceding vehicle; θ is the communication delay.

By taking Laplace transform of (6.1), a control diagram of hCACC can be depicted in Fig. 6.2, where:

$$K_a(s) = \frac{\alpha}{t_h}e^{-\varphi s}$$

$$K_b(s) = \beta s e^{-\varphi s}$$

$$H(s) = 1 + t_h s$$

$$D(s) = e^{-\theta s}\,.$$

And $G(s)$ and $F(s)$ are Laplace transforms of $g(u)$ and $f(\ddot{x}_2)$; s is the Laplace variable. Accompanied by a proper low-level controller as shown later, the longitudinal vehicle dynamics can be approximated by a first-order system:

$$g\big(u(t)\big) + \tau\dot{g}\big(u(t)\big) = u(t - \emptyset) \tag{6.2}$$

In time domain and a transfer function:

$$G(s) = \frac{1}{1+\tau s} e^{-\emptyset s} \tag{6.3}$$

In Laplace domain.

Where τ and $\emptyset$ are the response lag and actuator delay of the ego vehicle, respectively.

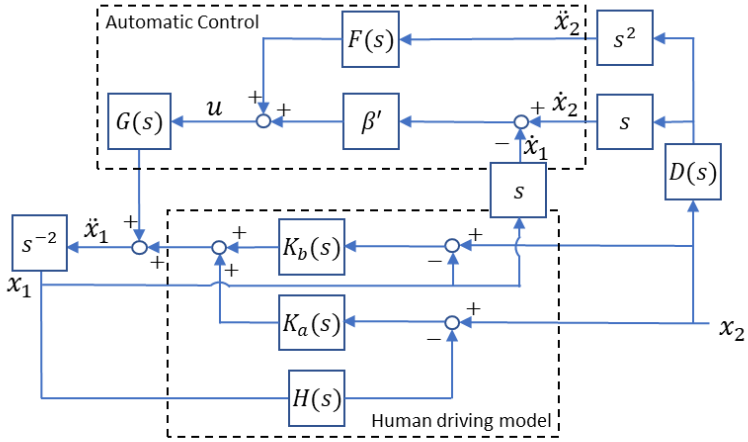


Fig. 6.2 Control diagram of hCACC

The feedforward filter $F(s)$ is designed to pre-compensate spacing error introduced by the speed perturbation of the preceding vehicle. We denote spacing error:

$$e_h(t) = h(t) - t_h \dot{x}_1(t) \tag{6.4}$$

Then the Laplace transform of spacing error can be derived by combining (6.1) and (6.4):

$$E_h(s) = \left(1 - \frac{H(s)(K_b(s)+K_a+G(S)D(s)\left(s^2F(s)+K_b'(s)\right))}{s^2+K_b(s)+GK_b'(s)+H(s)K_a(s)}\right)X_2(s) \tag{6.5}$$

Where $K_b' = \beta' s$; E_h(s) and $X_2(s)$ are the Laplace transforms of spacing error and location of the preceding vehicle, respectively. To make $E_h(s) = 0$ for any $X_2(s)$, the ideal feedforward filter $F(s)$ is:

$$F(s) = \frac{1 - t_h(\beta e^{-\varphi s} + \beta' G(s))}{H(s)G(s)D(s)}$$

Considering that the exact values of human parameters t_h, β, φ, and communication delay θ cannot be obtained, an approximated feedforward filter is given by:

$$F(s) = \frac{1 - \bar{t}_h \beta' G(s)}{(1 + \bar{t}_h s)G(s)} \tag{6.6}$$

Where $\bar{t}_h$ is the mean value of t_h over the past time. It can be estimated using the method proposed in [61]. An important assumption applied to the derivation of (6.6) is that the human driver tends to deactivate his own speed feedback control when an automatic speed feedback control is present, i.e., $\beta \to 0$ if $\beta' > 0$. This assumption is based on our observations in preliminary experiments of hCACC and to be further verified later.

For comparison, the car-following behavior of existing human-in the-loop CCC [58] is given by:

$$\begin{aligned} &h(t) = x_2(t) - x_1(t) - l \\ &\ddot{x}_1(t) = \alpha\left(\frac{1}{t_h} h(t - \varphi) - \dot{x}_1(t - \varphi)\right) + \beta \dot{h}(t - \varphi) + g(u) \\ &u = \gamma \ddot{x}_2(t - \theta) \end{aligned} \tag{6.7}$$

Where γ is the constant gain for acceleration feedforward in CCC, and $\gamma = 0.5$ is usually chosen to obtain the best performance [58]. Accordingly, the control diagram of this CCC can be given by replacing F with γ, and letting $\beta' = 0$ in Fig. 6.2.

Low-level control

The output of high-level controller is the extra desired acceleration of the vehicle. However, the longitudinal motion of vehicle is directly controlled by the throttle and brake.

Thus, a low-level controller is needed to convert the desired acceleration to proper throttle and brake action so that the command from high-level controller can be accurately achieved. A well-accepted version of low-level controller [80] for ACC/CACC utilizes the inverse engine torque map and a set of feedforward signals (i.e., vehicle speed, engine speed, transmission ratio) to pre-compensate the nonlinearity of the engine, transmission system, air drag and rolling resistance, leading to a first-order linear relationship between desired acceleration and actual acceleration. However, the case of hCACC is slightly different. The desired acceleration of the controller should be added onto human's action, not actuating the vehicle alone. Therefore, the throttle and brake generated by the existing low-level controller cannot be directly used. Instead, the modification on human's throttle/brake input should be further decided.

The goal of low-level controller is to make the vehicle acceleration as close as possible to the summed demands of human driver and high-level controller:

$$\ddot{x}_1 \to u_h + u \tag{6.8}$$

Where u_h is the intended acceleration by human driver, and u is the desired extra acceleration by hCACC high-level control. To achieve (6.8), we need:

$$gl(th_h + \Delta th, br_h + \Delta br) = u_h + u \tag{6.9}$$

Where $gl(\cdot)$ denotes the low-level vehicle dynamics model that maps throttle/brake input to the vehicle acceleration. th_h/br_h is the throttle/brake input by the human; $\Delta th/\Delta br$ is the modification on throttle/brake to be determined. While $gl(\cdot)$ is a nonlinear function, it can be known as the inverse of low-level controller in [80]. Since th_h and br_h can be sensed through throttle and brake pedal, human's intention u_h can also be computed:

$$u_h = gl(th_h, br_h) \tag{6.10}$$

Then, the new low-level control law can be derived combining (6.9) and (6.10):

$$(\Delta th, \Delta br) = gl^{-1}(gl(th_h, br_h) + u) - (th_h, br_h) \tag{6.11}$$

Where gl^{-1} is right the low-level controller in [80].

In the following analysis and evaluations, the ego vehicle will be presented by an Audi A8 sedan model from PreScan [83]. This physics-based vehicle model consists of engine, automatic gear box, 2-D chassis and other typical vehicle components. With the low-level controller (6.11), the first-order vehicle dynamics can be identified from the vehicle's response given step acceleration commands [95]. MATLAB system identification toolbox is adopted to accomplish this identification. The identification result is the same with (3.15):

$$G(s) = \frac{1}{1+0.12s} e^{-0.2s} \tag{6.12}$$

6.2 String Stability Analysis

As aforementioned, the human parameters $\alpha, \beta, t_h, \varphi$ are likely to vary over time. For a better robustness, hCACC should be able to work properly when there is a discrepancy between the expected $\alpha, \beta, t_h, \varphi$ and the actual values.

According to (3.1), the transfer function of human-driven vehicle without any automatic control is:

$$T(s) = \frac{K_a + K_b}{s^2 + K_b + H * K_a} \tag{6.13}$$

For CCC, the transfer function can be derived from (6.7):

$$T(s) = \frac{K_a + K_b + 0.5s^2GD}{s^2 + K_b + H * K_a} \tag{6.14}$$

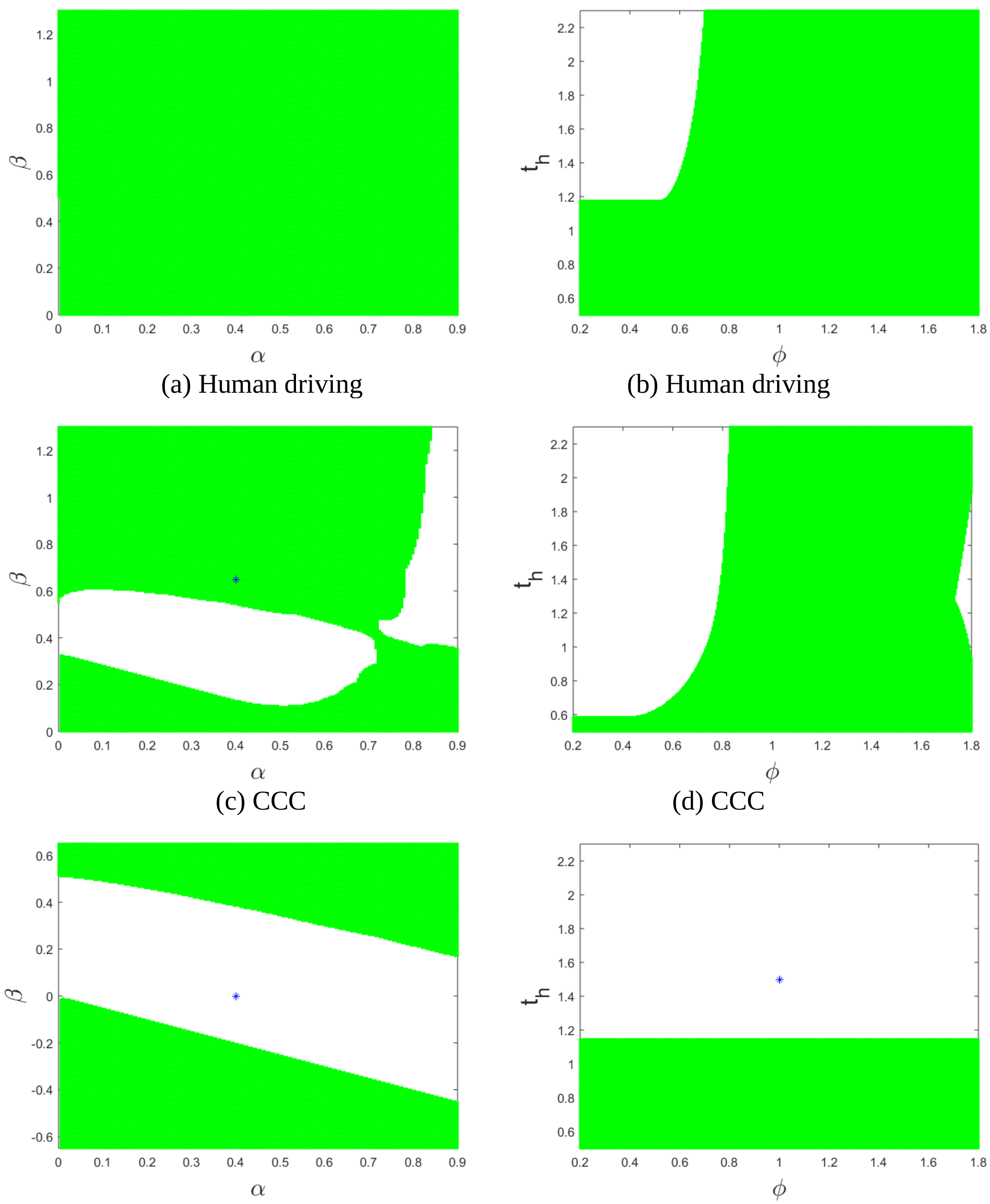
For hCACC, the transfer function can be derived from (6.1) and (6.6):

$$T(s) = \frac{H(K_a + K_b) + (s^2 + GK_b')D}{H(s^2 + K_b + GK_b' + HK_a)} \tag{6.15}$$

Next, we demonstrate the theoretical performance of human driving alone, CCC and hCACC by showing the ranges of human parameters which fulfill string stability. Such ranges can be computed based on (6.13)~(6.15), in which the vehicle dynamics $G(s)$ follow (6.12) and an average communication delay of 100ms [74] is assumed. While other vehicle dynamics and communication delays can also be used for this analysis, they show the similar pattern with the presented results. Generally speaking, broader string-stable ranges of human parameters indicate better chance to make the vehicle stay string-stable under various human behaviors.

By fixing desired time gap and human delay at their average values, i.e., $(t_h, \varphi) = (1.5, 1)$, Fig. 6.3 (a), (c), (e) show string-stable ranges (blank area) of α, β for human driving, CCC and hCACC, respectively. Then, fixing human gains at their average values, i.e., $(\alpha, \beta) = (0.4, 0.65)$, Fig. 6.3 (b), (d) show string-stable ranges of t_h, φ for human driving and CCC. Lastly, Fig. 6.3 (f) shows string-stable ranges of t_h, φ for hCACC when fixing $(\alpha, \beta) = (0.4, 0)$.

In human driving (Fig. 6.3(a), (b)), no positive α or β can be found to fulfill string stability when $(t_h, \varphi) = (1.5, 1)$. To make the vehicle string-stable with $(\alpha, \beta) = (0.4, 0.65)$, human delay should be no longer than 0.7s.



(a) Human driving (b) Human driving

(c) CCC (d) CCC

(e) hCACC (f) hCACC

Fig. 6.3 String-stable ranges of human parameters under different controls

According to Fig. 6.3 (c), (d), CCC offers broader string-stable ranges of α, β than human driving does. Besides, shorter time gap (e.g. 0.6s) is allowed but only in case of very small human delay (e.g. 0.5s).

For hCACC (Fig. 6.3(e), (f)), $\beta' = 0.65$ and $\bar{t}_h = 1.5$ are used in the high-level controller. Obviously, hCACC owns much broader string-stable ranges of human parameters than CCC does. Fig. 6.3 (e) shows that α can be almost any positive value when $\beta = 0$. In fact, residual β does not have to be exactly 0, because a broad “buffer area” around $(\alpha, \beta) = (0.4, 0)$ is provided. Fig. 6.3 (f) further shows that a vast range of time gaps can be used.

In addition, although hCACC requires the estimated $\bar{t}_h$ in (6.6), it actually has significant tolerance on the discrepancy between the estimated $\bar{t}_h$ and the actual t_h. This makes the possibility to further loose the operating condition of hCACC. Since $\bar{t}_h$ is the only human parameter needed in the configuration of hCACC, it is worth exploring whether hCACC can perform similarly well when replacing estimated $\bar{t}_h$ with a pre-tuned constant t_c. If this works, hCACC can be a generic control design instead of being user-specific, and also become more suitable for large-scale implementation due to the cancellation of human parameters estimation which requires minutes to be done [61].

A desirable t_c should give hCACC vehicle a good chance to stay string-stable when co-piloting under various human driving behaviors. To do so, String Stability Ratio (SSR) defined in Chapter 4 is utilized as a performance measure of the hCACC’s robustness against human parameters variation. For the reader’s convenience, we re-state the definition of SSR below:

$$SSR = \int\int\int\int f(\alpha, \beta, \varphi, t_h)\xi(SS)d\alpha d\beta d\varphi dt_h \tag{6.16}$$

Where

$$\xi(SS) = \begin{cases} 1 & if\ SS \leq 1 \\ 0 & if\ SS > 1 \end{cases} \tag{6.17}$$

SS is the string stability determinant that can be calculated through (6.13) to (6.15). $f(\alpha, \beta, \varphi, t_h)$ is the joint probability density function (PDF) of human parameters. Assuming independent distributions of human parameters [96], we have

$$f(\alpha, \beta, \varphi, t_h) = f(\alpha)f(\beta)f(\varphi)f(t_h) \tag{6.18}$$

Where $f(\alpha), f(\beta), f(\varphi), f(t_h)$ are the PDF of $\alpha, \beta, \varphi, t_h$, respectively. To obtain $f(\alpha), f(\beta), f(\varphi), f(t_h)$ and thus complete the configuration of pre-tuned hCACC, the distributions of human parameters under the effect of hCACC need to be collected. Note that the distributions of human parameters stated in *Assumption 2*, Chapter 3 are not directly usable, because the human's behavior with hCACC may be different from that when driving alone.

6.3 Evaluation

The purpose of the evaluation is threefold:

- Verify hCACC's performance over human driver and CCC;
- Confirm the validity of the assumption that human drivers would tend to deactivate their own speed feedback control during the onset of hCACC;
- Explore the effectiveness of derivative designs of hCACC.

There are two derivative designs of hCACC considered here: the aforementioned pre-tuned hCACC and semi-hCACC which only adjusts throttle but not brake of the vehicle. This semi- hCACC is to incorporate the fact that there are still many old or low-cost vehicles equipped with only electronic throttle but no electronic brake or ESP/ESC that support programmed brake for driver assistance system. Therefore, semi-hCACC is proposed to lower down the hardware threshold of hCACC for such vehicles.

Two rounds of driving simulator tests need to be conducted based on the evaluation purposes. In the first round, the participants drive with hCACC, CCC, and no-automation, respectively, through which the performance of hCACC can be quantified. Meanwhile, the distributions of human parameters in the hCACC runs are estimated, in order to validate the assumption on human's feedback, and to complete the configuration of pre-tuned hCACC. In the second round of tests, the pre-tuned hCACC and semi-hCACC are compared with the standard hCACC, to investigate how worse or better hCACC can do without human parameter estimation or automatic brake, respectively.

The measures of effectiveness (MOEs) are defined in three aspects:

- Safety. Time Exposed Time-to-collision (TET) [97] is adopted as a surrogate measure for safety performance. TET is calculated by accumulating the time periods when the vehicle is exposed to an unsafe Time-to-Collision (TTC):

$$\begin{aligned} TET &= \int \delta_i(t)dt \\ \delta_i(t) &= \begin{cases} 1, if\ TTC < TTC^* \\ 0, otherwise \end{cases} \\ TTC &= h(t)/\dot{h}(t) \end{aligned} \tag{6.19}$$

Where TTC^* is a threshold for unsafe TTC. According to NHTSA, TTC < 2s is considered a situation dangerous enough to activate Forward Collision Warning (FCW) system [98]. Therefore, $TTC^*$is chosen be to 2s in this study.

- Energy efficiency. Fuel consumption of the vehicle in each run is estimated using VT-micro fuel consumption model [84].

- Traffic disturbance. The root mean square (RMS) of acceleration and standard deviation (STD) of time gap are collected to quantify the speed/spacing disturbances the ego vehicle undergoes;

In addition to the numerical measures, string stability of the vehicle can be directly judged by comparing the speed/acceleration profile with preceding vehicle's and checking for speed/acceleration overshootings.

Experiment set-up

As shown in Fig. 6.4, the experiment combines an off-the-shelf software PreScan, the real traffic data from Next Generation SIMulation (NGSIM) program, and a driving simulator with Logitech hardware.

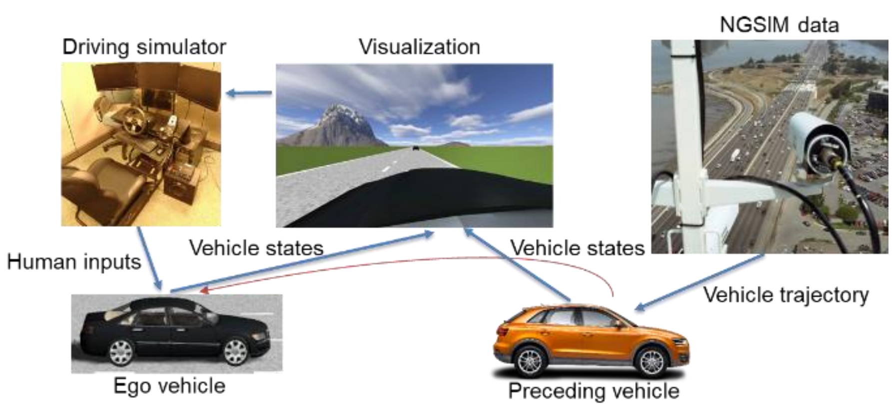


Fig. 6.4. Overview of the experiment

1) PreScan and control systems

PreScan is a simulation platform designed for evaluating Advanced Driver Assistance Systems (ADAS). In this study, it offers simulation environment and physics-based vehicle dynamics model. A straight-highway car-following scenario is developed and visualized in driver's view via PreScan. This driver's view is updated in 20Hz and projected to the screen (windshield) of the driving simulator so that the participant can perceive the situation and take actions on throttle/brake pedal and steering wheel. These actions are then fed back to PreScan. Using the vehicle dynamics model embedded, PreScan can calculate how the driver's actions, along with hCACC/CCC, change the status of the ego vehicle and reflect the change in the driver's view.

Because PreScan's vehicle models are built in a form of MATLAB Simulink, it is convenient to develop the control systems of these vehicles in Simulink. As noted, the control system for ego vehicle is divided into high-level and low-level controllers. There are two high-level controllers (i.e., hCACC and CCC) to be evaluated, while the low-level system remains the same.

2) Predecessor driving behavior

To make the evaluation more realistic, the speed profile of preceding vehicle is derived from the real-world vehicle trajectory data of Next Generation Simulation (NGSIM) program [82], which was launched by the Federal Highway Administration (FHWA). NGSIM used high-resolution cameras to record trajectories of the vehicles on real roads. The US Highway 101 (US 101) dataset was one dataset that reflected highway traffic condition. It contains the location and speed profiles of vehicles in all 6 lanes within the 640-meter long study area during 45 minutes. Due to the limited length of the single vehicle's speed profile in the NGSIM data, we link 4 short speed profiles of different vehicles, by constant deceleration of $1m/s^2$, into a 4-minute speed profile as shown in Fig. 6.5. It should be emphasized that this speed profile is not directly given to the preceding vehicle in the evaluation; instead, it is “tracked” through a PID controller and vehicle dynamics model. This setting is designed to eliminate the inconsistent speeds and unrealistic jerks (i.e., derivative of acceleration) that frequently occur in the original NGSIM data [85]. The PID controller and the vehicle dynamics model together play as a filter that smooths the trajectory and makes sure the movements are mechanically realistic. To fairly compare the performances of different controls, the trajectory of the preceding vehicle is set to be identical in all the runs.

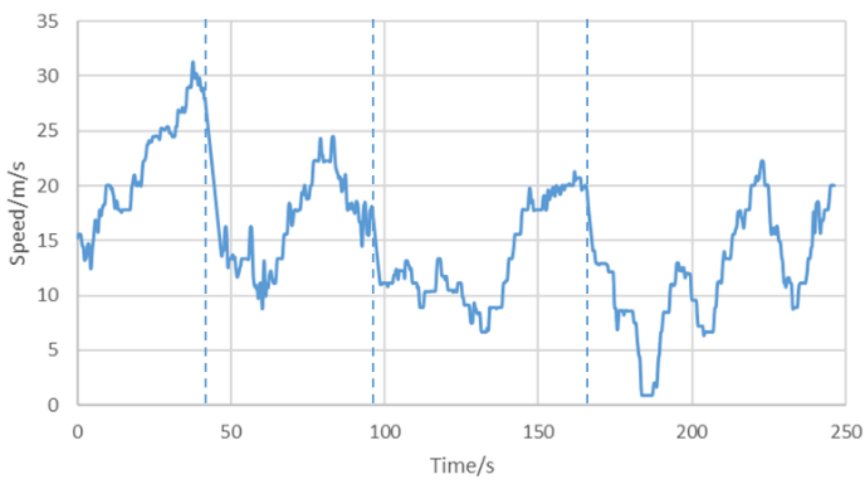


Fig. 6.5 Desired speed profile of the preceding vehicle

3) Human drivers

There are 8 participants in the first round of tests, and 4 in the second round of tests. All of them are college/graduate students. In the first round, each participant is required to drive the ego vehicle and track the preceding vehicle for 4 runs (each run lasts for 4 minutes): warm-up run, hCACC, CCC, and human driving. The purpose of warm-up run is to familiarize the driver with driving simulator, and estimate the human parameters using the method from [61], after which hCACC can be configured. The warm-up run is always placed the first, while the other 3 runs are randomly sequenced to disperse the driver's learning and fatigue effects. In the second round, there are 4 runs besides warm-up for each participant: human driving, hCACC, pre-tuned hCACC, and semi-hCACC, in a randomized sequence.

Results

1) First round of tests

The results of the first round of tests are listed in Table 6.1. hCACC reduced 36.8% acceleration, 31.2% time-gap fluctuation, 81.2% exposure time to unsafe driving situations (TET), and 15.8% fuel consumption from those of human driving, respectively. Paired t-test indicates all these benefits are statistically significant. This means hCACC has great potential to mitigate traffic disturbances, avoid unsafe driving condition and save energy. It is noted that these benefits were achieved by hCACC at an even shorter gap than human driving did. It is also observed that under hCACC, human drivers took 31.2% and 64.3% less control effort on throttle and brake levels than in human driving alone, indicating a decrease in labor intensity of the drivers.

TABLE 6.1
SUMMARY OF EVALUATION RESULTS

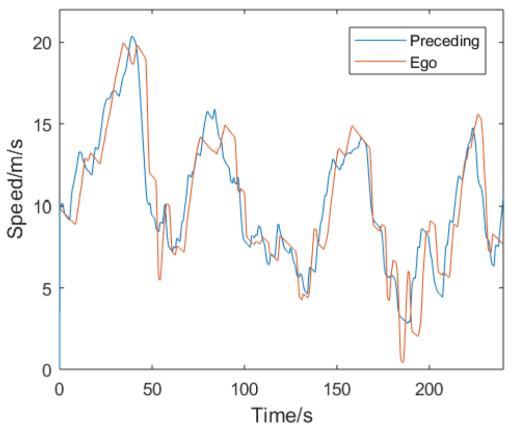
# of Participant		1	2	3	4	5	6	7	8	Mean
hCACC	RMS Acceleration/m/s^2	0.7	0.67	0.73	0.78	0.648	0.57	0.65	0.645	0.67
	Time gap (mean±STD)/s	1.7±0.55	2.03±0.72	1.87±0.56	0.97±0.34	1.55±0.55	1.3±0.55	1.95±0.97	1.74±0.5	1.64±0.59
	TET/s	0	0	0	1.8	0	3.9	0	0	0.7125
	Fuel consumption/L	0.157	0.151	0.161	0.165	0.147	0.14	0.149	0.153	0.15
	RMS throttle/brake	10%/4%	10%/5%	11%/6%	13%/5%	9%/2%	9%/2%	13%/2%	11%/2%	11%/4%
CCC	RMS Acceleration/m/s^2	1.02	0.8	1.24	0.91	1.31	0.83	1.04	1.25	1.05
	Time gap (mean±STD)/s	1.48±1.15	2.24±1.27	1.61±1.08	1.14±0.47	1.39±0.86	1.78±0.81	2.19±1.02	1.56±0.72	1.67±0.93

	TET/s	2.2	6.6	5.3	6.2	8.7	4.6	2.8	5.9	5.3
	Fuel consumption/L	0.171	0.147	0.187	0.16	0.174	0.155	0.163	0.207	0.17
	RMS throttle/brake	15%/13%	13%/9%	16%/16%	15%/11%	16%/16%	14%/9%	16%/14%	17%/16%	15%/13%
Human driving	RMS Acceleration/m/s^2	0.98	0.86	1.36	0.99	1.08	0.99	1.21	1.07	1.07
	Time gap (mean± STD)/s	1.86±1.04	2.74±1.31	2.31±0.85	0.92±0.45	1.93±0.69	1.69±0.73	1.59±1.12	1.39±0.65	1.80±0.86
	TET/s	2.1	3.9	2.6	4.5	1.1	0.6	8.3	7.3	3.8
	Fuel consumption/L	0.154	0.154	0.219	0.163	0.174	0.189	0.197	0.19	0.18
	RMS throttle/brake	14%/12%	13%/10%	16%/17%	17%/11%	15%/13%	17%/11%	17%/16%	16%/12%	16%/13%

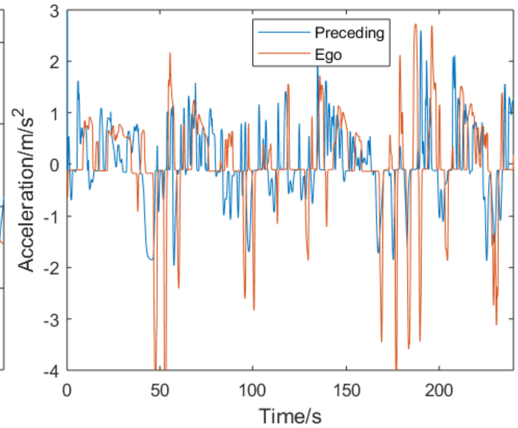
When compared with CCC, hCACC reduced 35.8% acceleration, 36.6% time-gap fluctuation, 86.5% exposure time to unsafe driving situations (TET), and 10.3% fuel consumption, showing consistently large improvements.

In contrast, paired t-test indicates that all the resulted MOEs of CCC show no statistically significant difference from those of human driving alone. Thus, there was no performance improvement over human driving achieved by CCC.

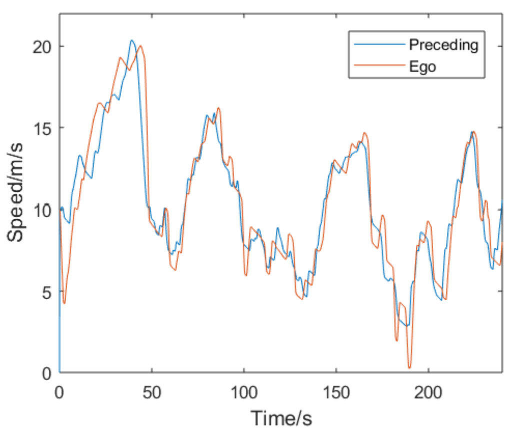
Fig. 6.6 shows the speed and acceleration profiles of the participant 1. Fig. 6.6 (a), (b) are for human driving; (c), (d) are for CCC; and (e), (f) are for hCACC. It is noticed that many speed overshootings occurring in human driving and CCC runs were avoided in hCACC, and the acceleration of hCACC vehicle was mostly smaller than that of the preceding vehicle, indicating an improved string stability. It is noted that string stability is not the only contributing factor to the good performance of hCACC. The results are also largely determined by “how poorly” the hCACC/CCC/human driving performed in string-unstable conditions. When comparing Fig. 6.6(c) with (a) and (e), it is clear that hCACC not only had the better chance to avoid overshootings, but also greatly suppressed the magnitude of the overshooting when it happens. This finding highlights the importance of experiments with real drivers instead of only looking at theoretical analysis.



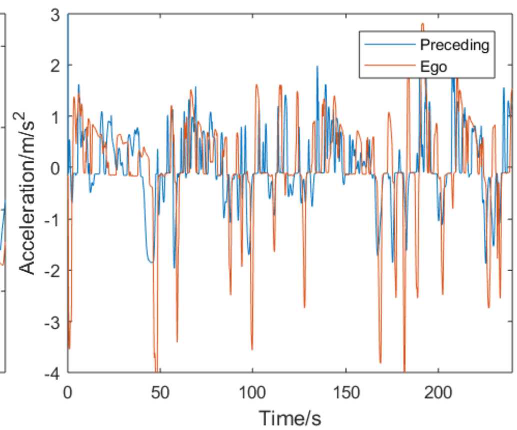
(a) Human driving



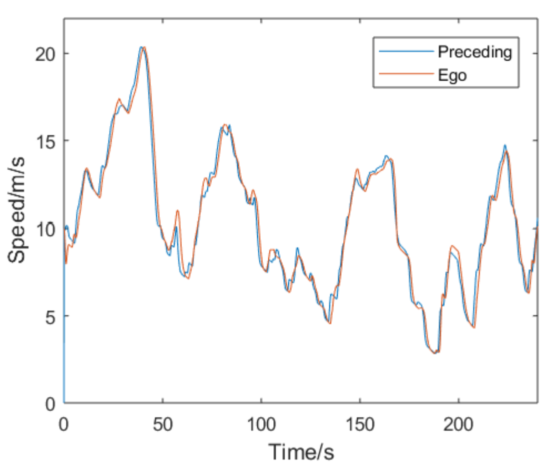
(b) Human driving



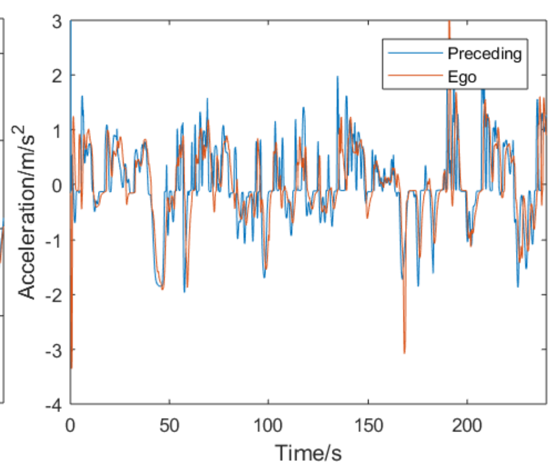
(c) CCC



(d) CCC



(e) hCACC



(f) hCACC

Fig. 6.6. Speed and acceleration profiles of the participant 1

2) Driving behavior under hCACC

Another purpose of the first round of tests was to obtain human parameters under the effect of hCACC so that the assumption can be tested that human's speed feedback can be checked and pre-tuned.

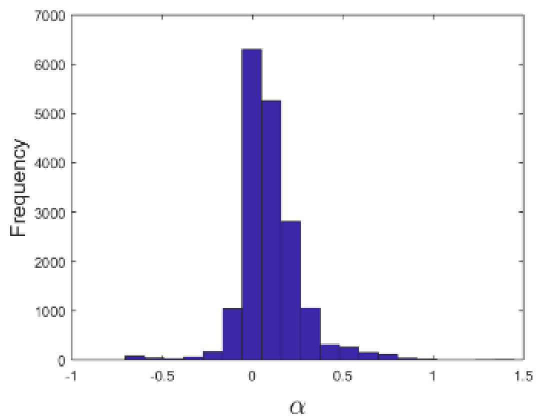
The human parameters in all the runs are estimated using the method in [61] and their mean values (i.e., $\bar{\alpha}$, $\bar{\beta}$, $\bar{\varphi}$ and $\bar{t}_h$) are listed in Table 6.2. In average, $\bar{\beta}$ under hCACC is only 33% of what it used to be in the human driving. This proves our assumption reasonable and explains the favorable performance of hCACC, which needs the residual β to be small. In addition, each driver showed different human parameters with or without CCC. It does challenge the fundamental assumption CCC adopted that human behaves the same even with the help of ADAS.

TABLE 6.2
SUMMARY OF ESTIMATED HUMAN PARAMETERS IN ALL RUNS

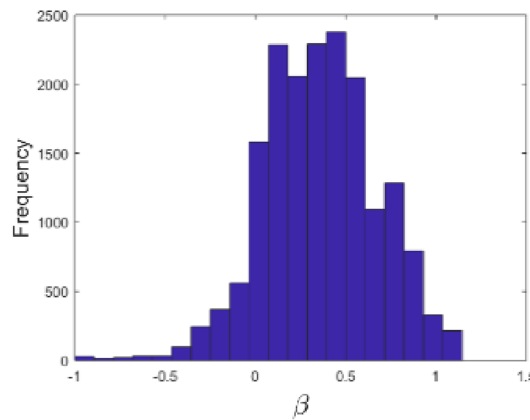
# of participant		1	2	3	4	5	6	7	8	Mean
hCACC	$\bar{\alpha}$	0.05	0.03	0.06	0.02	0.05	0.03	0.03	0.04	0.04
	$\bar{\beta}$	0.25	0.08	0.14	0.25	0.08	0.06	0.02	0.05	0.12
	$\bar{\varphi}$	1.8	1.66	1.69	1.08	1.38	1.31	1.92	1.63	1.56
	$\bar{t}_h$	1.08	1.46	1.29	0.51	1.16	0.65	1.11	1.08	1.04
CCC	$\bar{\alpha}$	0.08	0.06	0.09	0.04	0.11	0.1	0.09	0.14	0.09
	$\bar{\beta}$	0.45	0.17	0.47	0.44	0.42	0.25	0.3	0.27	0.35
	$\bar{\varphi}$	1.31	1.44	1.19	1.11	1.32	1.19	1.56	1.29	1.30
	$\bar{t}_h$	0.76	1.52	0.85	0.69	0.69	1.2	1.48	0.93	1.02
Human driving	$\bar{\alpha}$	0.1	0.06	0.2	0.06	0.1	0.1	0.09	0.13	0.11
	$\bar{\beta}$	0.3	0.2	0.34	0.62	0.31	0.38	0.37	0.3	0.35
	$\bar{\varphi}$	1.24	1.68	1.37	1.04	1.29	1.25	1.18	1.25	1.29
	$\bar{t}_h$	1.16	2.09	1.77	0.48	1.21	1.1	0.95	0.9	1.21

As noted, the pre-tuning of hCACC requires the probability density function (PDF) of human parameters under the effect of hCACC. Fig. 6.7 shows the distributions of estimated $\alpha, \beta, t_h, \varphi$ in the human driving runs (in Fig. 6.7(a)~(d)), and those in the hCACC runs (in Fig.

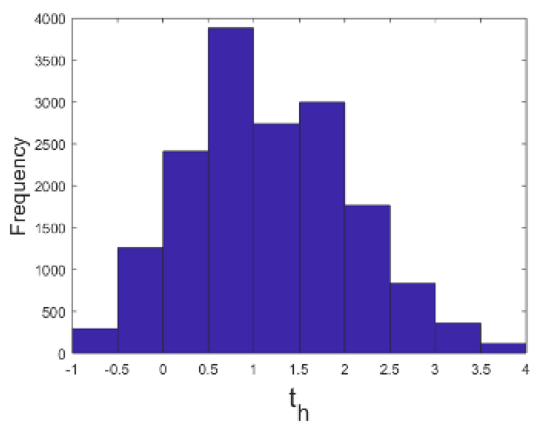
6.7(e)~(h)), respectively. Generally speaking, the human behavior under hCACC was quite different from driving alone. It can be seen that hCACC shifts the distributions of human gains α, β to the left, and α, β have significantly high frequencies to be zero. In addition, the human delay under hCACC tends to be either extremely short or extremely long, while in human driving runs the human delay is concentrated between 0.5s and 1.5s.



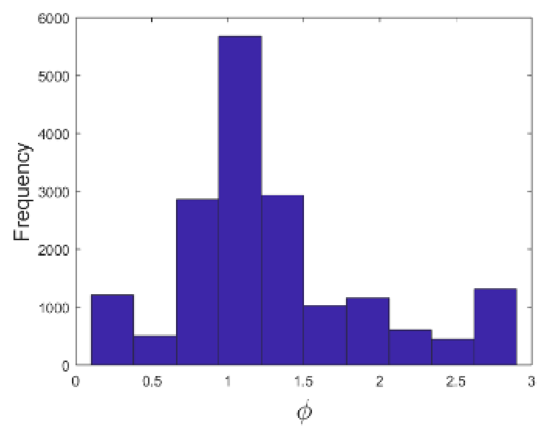
(a)



(b)



(c)



(d)

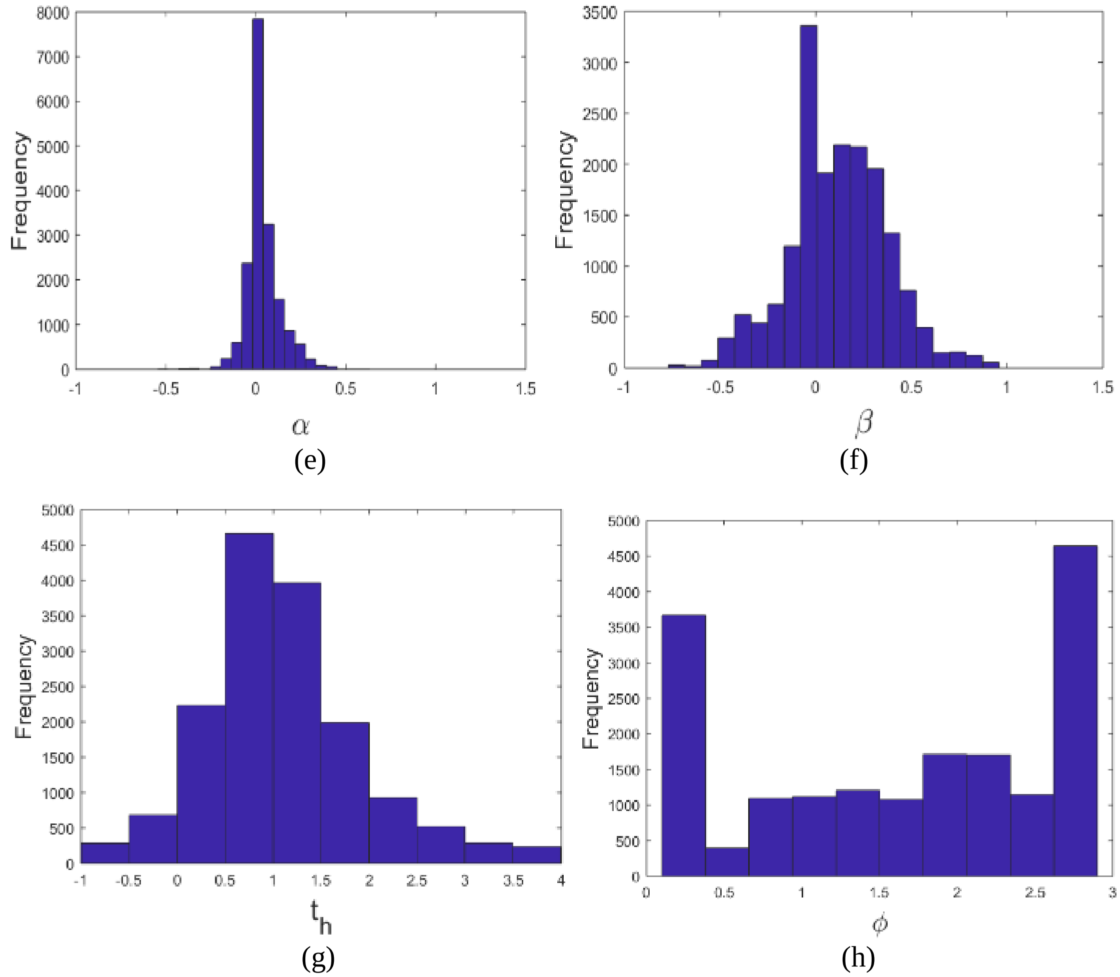


Fig. 6.7. Distribution of estimated human parameters in human driving and hCACC

While different distributions of human parameters and t_c may be obtained when more data is available, the estimated $\alpha, \beta, t_h, \varphi$ in the 8 hCACC runs are fitted into PDFs of kernel distributions. With these PDFs, the optimal $t_c = 1s$ can be found to achieve the maximum SSR=28%, which is the probability to secure string stability.

3) Second round of tests

The evaluation of semi-hCACC and pre-tuned hCACC were conducted with another four participants, and the results are summarized in Table 6.3. The results of human driving and standard hCACC (with human parameters estimation) were also reported to facilitate easier comparison.

TABLE 6.3
EVALUATION RESULTS OF PRE-TUNED HCACC AND SEMI-HCACC

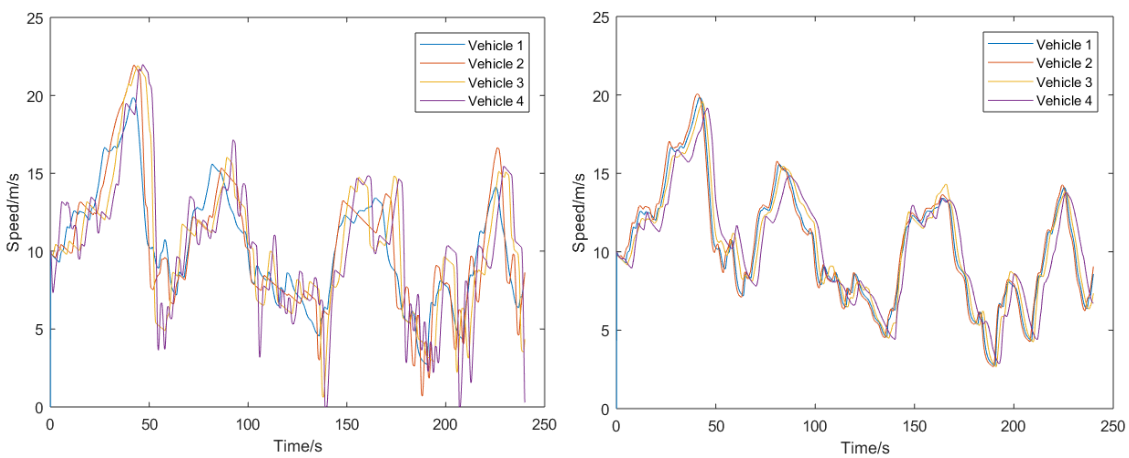
# of participant		1	2	3	4	Mean
Standard hCACC	RMS acceleration/m/m^2	0.87	0.63	0.78	0.96	0.81
	Time gap STD/s	0.65	0.42	0.41	0.39	0.47
	TET/s	0	0	0	7.7	1.93
	Fuel consumption/L	0.155	0.119	0.129	0.146	0.14
Pre-tuned hCACC	RMS acceleration/m/s^2	0.93	0.65	0.82	0.86	0.82
	Time gap STD/s	0.5	0.48	0.44	0.34	0.44
	TET/s	0	0	0.6	1.4	0.50
	Fuel consumption/L	0.157	0.121	0.127	0.147	0.14
Semi-hCACC	RMS acceleration/m/s^2	1.11	0.74	0.97	1.08	0.98
	Time gap STD/s	0.67	0.48	0.39	0.26	0.45
	TET/s	0	1.7	0.2	4.7	1.65
	Fuel consumption/L	0.163	0.124	0.125	0.171	0.15
Human driving	RMS acceleration/m/s^2	1.76	0.87	2.04	0.98	1.41
	Time gap STD/s	0.95	0.51	0.8	0.31	0.64
	TET/s	0	1.2	9	4.9	3.78
	Fuel consumption/L	0.273	0.145	0.28	0.154	0.21

It can be seen that there were no significant differences between the performances of hCACC and pre-tuned hCACC in all aspects, which means the time-consuming human parameters estimation can be omitted in the implementation of hCACC. This finding makes hCACC easier to be implemented.

Meantime, semi-hCACC had similar performances in time gap STD, TET, and fuel consumption with hCACC. Although its RMS acceleration is 20% higher than standard hCACC, it is still 30% lower than the human driving baseline. Therefore, semi-hCACC is a good alternative for vehicles without electronically controllable brake.

Finally, we conducted experiments to demonstrate the platoon-wise performance of 4 human-driving-alone vehicles vs. 4 hCACC vehicles. In these tests, the leading vehicle followed

NGSIM data, while the following three vehicles were driven by volunteers through driving simulator (one by one), with or without the aid of hCACC. The speed profiles of human driving and hCACC platoons are shown in Fig. 6.8. It can be seen that the traffic disturbances were amplified by the human-driving-alone platoon but mitigated by hCACC platoon. When looking at the last vehicle in the platoons, hCACC reduced 50% fuel consumption and 100% TET from human driving. These improvements (especially in fuel saving) are much greater than that in the individual-vehicle tests, indicating that the cumulative effects of individual vehicles can lead to great difference in the traffic quality.



(a) Human driving (b) hCACC

Fig. 6.8. Speed profiles of the 4-vehicle platoon under human driving and hCACC

6.4 Key findings

In this chapter, a human-in-the-loop CACC algorithm (hCACC) was developed for connected human-driven vehicle. By allowing coexistence of the automatic control and driver's actions in a beneficial way, hCACC helps the human driver stabilize the vehicle more efficiently and safely. String stability analysis showed that hCACC can offer broader string-stable ranges of human parameters than human driving alone or the existing human-in-the loop Connected Cruise Control (CCC), indicating a better capability to mitigate traffic disturbance with the uncertain human behaviors. The performance of hCACC was investigated in driving simulator experiments, with CCC and human driving being baselines. Compared with human driving alone, hCACC reduced 36.8% acceleration, 31.2% time-gap fluctuation, 81.2% exposure time to unsafe driving situations, and 15.8% fuel consumption, while CCC achieved no significant improvements. In addition, two derivative designs of hCACC, i.e., pre-tuned hCACC and semi-hCACC are proposed and proven similarly effective, further lowering down the practice threshold of hCACC.

7. Conclusions and Future Research

This chapter summarizes the research efforts made in the design and validation of quasi-CACC systems. Based on the key findings through this dissertation, it is recommended to implement quasi-CACC in the near future so that CAVs and CHVs can generate the maximum benefits under the imperfect market penetration. The potential enhancements of the proposed approaches are discussed in the end.

7.1 Conclusions

To fully harvest the benefits of vehicular automation and connectivity in the mixed traffic, Quasi-CACC applications were proposed in this dissertation to extend the usability of cooperative longitudinal control of vehicles. These applications are expected to be important complement to the currently prevailing method, i.e., CACC, which is seriously limited by relatively high threshold for operations in mixed traffic of both CAVs, CVs, and unconnected vehicles.

First, a new control algorithm for CAV, CACC with unconnected vehicle (CACCu), is developed to enable closely following an unconnected preceding vehicle. When the 1st (nearest) preceding vehicle is unconnected, CACCu utilizes the communication with the further (connected) preceding vehicle to improve the response of ego vehicle. A linear time-invariant controller of CACCu inheriting the feedback-feedforward control structure of typical CACC was designed. An additional filter of "virtual preceding vehicle(s)" is inserted before the original feedforward filter of CACC, to compensate for the effects of n unconnected preceding vehicle(s) in between. String stability analysis in frequency domain was conducted to investigate the theoretical performance of CACCu. The controller parameters of CACCu were tuned to maximize the probability of being string-stable. The performance of CACCu was evaluated and compared with ACC and acceleration-based CCC, using real vehicle trajectory data from NGSIM and physics-based vehicle model from PreScan.

Furthermore, to address the remaining limitations in the CACCu (e.g., requiring the number of unconnected vehicles (n) to be known for achieving the ideal performance) and optimize its performance, we proposed an adaptive model predictive control (A-MPC) approach. In A-MPC CACCu, the system states are estimated and predicted in rolling horizon by an adaptive system model, based on which a constrained multi-objective optimization is solved for determining the optimal control command. An initial system model is derived from *a priori* knowledge as a starting point of the control, and online model estimators are utilized to update the model parameters based on the actual motions of the unconnected preceding vehicle and the further connected vehicle. To comprehensively compare the performances of the A-MPC-based CACCu and the previously proposed linear CACCu, the entire NGSIM data from the US101 innermost lane (with the fewest lane-changes) were used to simulate various car-following scenarios.

The proposed CACCu was validated in the field to ensure that it can be implemented to real world. The experiment was conducted with two automated vehicles equipped with Mobileye sensors and WIFI modules. The original CACCu algorithm is adapted to comply with speed-command-based control in these vehicles. By commanding the 1st preceding to follow a NGSIM real-traffic trajectory, and simultaneously spread the information of the 2nd preceding vehicle in NGSIM data, CACCu was able to be tested in the three-vehicle-sandwich with only two actual vehicles. ACC and human driving served as performance baseline.

The key findings in the design and validation of CACCu are:

- The theoretical analysis indicated that a proper filter of "virtual preceding vehicle" inserted to the original CACC feedforward filter can make CACCu vehicle stay string-stable at a gap significantly shorter than that required by ACC. Such capability is robust against the uncertainty in unconnected vehicle's car-following behaviors, thus no beforehand identification process or extra information on the unconnected vehicles' behaviors is required;
- The high-fidelity simulation results showed that CACCu avoided most of speed overshootings happening to ACC and CCC, indicating improved string stability. CACCu also led to overall 6~9% acceleration reduction, 35~49% spacing error reduction and 3~7% fuel saving from ACC. Compared with CCC, CACCu achieved 5~8% acceleration reduction, 26~38% spacing error reduction and 2~3% fuel saving. These numbers indicated benefits of CACCu in safety, ride comfort and energy efficiency;
- The comprehensive comparison between linear CACCu and A-MPC CACCu showed that A-MPC CACCu slightly outperformed the linear CACCu when n (i.e., the number of unconnected preceding vehicles) was accurately known, and largely outperformed linear CACCu when n was unknown. Therefore, A-MPC CACCu is proven a more robust method for the implementation in complicated traffic situations;
- In the field experiment, it was found that CACCu reduced 10.82% acceleration RMS, 60.79% spacing error RMS and 6.24% fuel consumption from ACC's. Compared with human driving, CACCu reduced 17.64% acceleration and 13.43% fuel

consumption. The speed profiles showed that CACCu greatly attenuated the traffic disturbances while ACC and human driving tended to amplify them. It was confirmed that CACCu can greatly attenuate the traffic disturbance and improve safety, comfort, and fuel efficiency.

To take a full advantage of vehicular connectivity, a human-in-the-loop CACC algorithm (hCACC) was further developed for human-driven connected vehicle. In hCACC, the human driver remains engaged in the longitudinal control of the vehicle, and hCACC controller applies additional acceleration/deceleration on top of human actions according to the received status of preceding vehicle. By allowing coexistence of the automatic control and driver's actions in a beneficial way, hCACC helps the human driver stabilize the vehicle more efficiently and safely. The proposed hCACC inherited the feedback-feedforward control structure and velocity-dependent spacing policy from typical CACC. The hCACC along with human driving and CCC were evaluated by driving simulator experiments, in which the NGSIM data and PreScan vehicle model were utilized for the high fidelity. In addition, two derivative designs of hCACC, i.e., pre-tuned hCACC and semi-hCACC were proposed and tested.

The key findings in the design and validation of hCACCu are:

- String stability analysis showed that hCACC can offer broader string-stable ranges of human parameters than human driving alone or the existing human-in-the loop Connected Cruise Control (CCC), indicating a better capability to mitigate traffic disturbance under the uncertain human behaviors;

- Driving simulator experiments showed that hCACC reduced 36.8% acceleration, 31.2% time-gap fluctuation, 81.2% exposure time to unsafe driving situations, and 15.8% fuel consumption from those of human driving alone;
- Humans drive differently under the influence of ADAS. When aided by hCACC, human drivers tended to partially deactivate their feedback control, and the resulted reductions in human gains were in favor of hCACC;
- Pre-tuned hCACC, which does not require any information about the driver, showed almost the same performance with a standard hCACC in the experiments. Semi-hCACC, for vehicle without automated brake ability, were proven to enjoy most of hCACC's benefits. These findings further lowering down the practice threshold of hCACC.

This dissertation introduced quasi-CACC as a transformative technology that resolves the limitation of current CACC (i.e., CACC does not work efficiently in mixed traffic). On the one hand, CACCu makes the CAV capable of performing feedforward control solely using the received information from a further preceding vehicle when the immediately preceding vehicle is unconnected. This new feature would allow CAV to closely follow an unconnected vehicle instead of falling back to Adaptive Cruise Control (ACC) which requires much longer headway to be stable. On the other hand, a human-in-the-loop CACC algorithm is designed to co-pilot the human driver based on received information from preceding connected vehicle and help stabilize the vehicle more smoothly and safely in the traffic turbulence. A notable technical merit is that the design of quasi-CACC adopted highly robust control strategies to handle uncertainties of human driver's behaviors in both traditional vehicles and CHVs, canceling the need for behavior pattern identification or extra information about the human drivers. The findings from high-fidelity simulation and field test indicated that quasi-CACC systems have great potential in improving the string stability, safety, ride comfort and fuel efficiency performances of CAV or CHV.

Since it would likely to be decades before the market penetration rate of connected automated vehicles reaches 100%, the proposed quasi-CACC systems are recommended to be implemented soon in CAVs and CHVs, by which the vehicular connectivity and automation could generate the maximum benefits to the mixed traffic.

7.2 Future Research

For CACCu, although the A-MPC approach addressed most of obstacles on the way to implementation, there is still room for enhancement in terms of the accuracy of prediction model. As a starting point, this study adopted a simple second-order linear model to represent the dynamics between any two preceding vehicles. However, a neural-network-based adaptive MPC [99] may lead to even better performance, considering that neural network has been more accurate in predicting human's car-following behaviors than traditional car-following models [100].

For hCACC, it is important to ensure that the drivers "feel" the hCACC helpful and expectable instead of disturbing, to build up human's trust in hCACC. The future investigation should look into whether and when hCACC may be conflicting with human's intentions, and driver's subjective satisfaction on hCACC. Then the control logic/algorithm may need to be revised and an activation/deactivation mechanism of hCACC should be designed to resolve the conflict if the drivers report it as a notable problem. To this end, tests with real vehicles, or at least motion-enabled driving simulator are necessary to given the users “real feeling”. In addition, the demographic factors in the performance of hCACC is also worth further investigation.

Lastly, the presented simulations and field tests were focused on the individual vehicle’s behaviors in small-scale scenarios involving only limited number of vehicles. It is still unclear how quasi-CACC would affect the performance (e.g., mobility and sustainability) of transportation networks. Therefore, an important future research is to quantify the impacts of quasi-CACC on transportation efficiency and sustainability via network-wide microscopic simulations. The results could help identifying market penetration breaking points, if any, for supporting policy decision.

References

[1] World Health Organization, "Global status report on road safety 2018," 2019.

[2] The Texas A&M Transportation Institute, "2019 Urban Mobility Report," 2019.

[3] U.S. Department of Transportation, "Traffic Safety Facts: Critical reasons for crashes investigated in the National Motor Vehicle Crash Causation Survey," 2015.

[4] SAE International, "Summary of SAE International's Levels of Driving Automation for On-Road Vehicles," *SAE International*. 2014.

[5] K. Bengler, K. Dietmayer, B. Farber, M. Maurer, C. Stiller, and H. Winner, "Three decades of driver assistance systems: Review and future perspectives," *IEEE Intell. Transp. Syst. Mag.*, vol. 6, no. 4, pp. 6–22, 2014.

[6] X. Mosquet, M. Andersen, and A. Arora, "A Roadmap to Safer Driving Through Advanced Driver Assistance Systems," *Auto Tech Rev.,* 2016.

[7] E. R. Teoh and D. G. Kidd, "Rage against the machine? Google's self-driving cars versus human drivers," *J. Safety Res.*, 2017.

[8] A. Mukhtar, L. Xia, and T. B. Tang, "Vehicle Detection Techniques for Collision Avoidance Systems: A Review," *IEEE Transactions on Intelligent Transportation Systems*. 2015.

[9] B. M. Masini, A. Bazzi, and A. Zanella, "A survey on the roadmap to mandate on board connectivity and enable V2V-based vehicular sensor networks," *Sensors (Switzerland),* vol. 18, no. 7, 2018.

[10] A. Festag, "Standards for vehicular communication—from IEEE 802.11p to 5G," *Elektrotechnik und Informationstechnik,* 2015.

[11] V. Vukadinovic *et al.*, "3GPP C-V2X and IEEE 802.11p for Vehicle-to-Vehicle communications in highway platooning scenarios," *Ad Hoc Networks*, 2018.

[12] S. Ahmadi, *5G NR: Architecture, technology, implementation, and operation of 3GPP new radio standards*. 2019.

[13] A. A. Malikopoulos, S. Hong, B. B. Park, J. Lee, and S. Ryu, "Optimal Control for Speed Harmonization of Automated Vehicles," *IEEE Trans. Intell. Transp. Syst.,* 2019.

[14] J. Lee and B. Park, "Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment," *IEEE Transactions on Intelligent Transportation Systems*. 2012.

[15] Z. Zhao, Z. Wang, G. Wu, F. Ye, and M. J. Barth, "The State-of-the-Art of Coordinated Ramp Control with Mixed Traffic Conditions," in *2019 IEEE Intelligent Transportation Systems Conference, ITSC 2019,* 2019.

[16] Z. Wang, Y. Bian, S. E. Shladover, G. Wu, S. E. Li, and M. J. Barth, "A Survey on Cooperative Longitudinal Motion Control of Multiple Connected and Automated Vehicles," *IEEE Intelligent Transportation Systems Magazine*. 2020.

[17] D. Bevly *et al.*, "Lane change and merge maneuvers for connected and automated vehicles: A

survey," *IEEE Trans. Intell. Veh.*, 2016.

[18] M. Aramrattana *et al.*, "Team Halmstad Approach to Cooperative Driving in the Grand Cooperative Driving Challenge 2016," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 4, pp. 1248–1261, 2018.

[19] R. Hult *et al.*, "Design and Experimental Validation of a Cooperative Driving Control Architecture for the Grand Cooperative Driving Challenge 2016," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 4, pp. 1290–1301, 2018.

[20] P. Bansal and K. M. Kockelman, "Forecasting Americans' long-term adoption of connected and autonomous vehicle technologies," *Transp. Res. Part A Policy Pract.*, vol. 95, pp. 49–63, 2017.

[21] N. H. T. S. A. DEPARTMENT OF TRANSPORTATION, *Federal Motor Vehicle Safety Standards; V2V Communications A Proposed Rule by the National Highway Traffic Safety Administration*. 2017.

[22] V. Milanés, S. E. Shladover, and J. Spring, "Cooperative Adaptive Cruise Control in Real Traffic Situations," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 1, pp. 296–305, 2014.

[23] S. Shladover, D. Su, and X.-Y. Lu, "Impacts of Cooperative Adaptive Cruise Control on Freeway Traffic Flow," *Transp. Res. Rec. J. Transp. Res. Board*, vol. 2324, no. January, pp. 63–70, 2012.

[24] J. Lioris, R. Pedarsani, F. Y. Tascikaraoglu, and P. Varaiya, "Platoons of connected vehicles can double throughput in urban roads," *Transp. Res. Part C Emerg. Technol.*, vol. 77, pp. 292–305, 2017.

[25] J. Ma, F. Zhou, and M. Demetsky, "Evaluating mobility and sustainability benefits of cooperative adaptive cruise control using agent-based modeling approach," in *Systems and Information Design Symposium (SIEDS), 2012 IEEE*, 2012, pp. 74–78.

[26] E. Talavera, A. Díaz-Álvarez, F. Jiménez, and J. Naranjo, "Impact on Congestion and Fuel Consumption of a Cooperative Adaptive Cruise Control System with Lane-Level Position Estimation," *Energies*, vol. 11, no. 1, p. 194, 2018.

[27] S. Shladover *et al.*, "Cooperative Adaptive Cruise Control (CACC) For Partially Automated Truck Platooning: Final Report," 2018.

[28] R. Rajamani and S. E. Shladover, "Experimental comparative study of autonomous and co-operative vehicle-follower control systems," *Transp. Res. Part C Emerg. Technol.*, vol. 9, no. 1, pp. 15–31, 2001.

[29] J. Ploeg, B. T. M. Scheepers, E. Van Nunen, N. Van De Wouw, and H. Nijmeijer, "Design and experimental evaluation of cooperative adaptive cruise control," in *Conference on Intelligent Transportation Systems (ITSC)*, 2011, pp. 260–265.

[30] G. J. L. Naus, R. P. A. Vugts, J. Ploeg, M. J. G. Van De Molengraft, and M. Steinbuch, "String-stable CACC design and experimental validation: A frequency-domain approach," *IEEE Trans. Veh. Technol.*, vol. 59, no. 9, pp. 4268–4279, 2010.

[31] V. Milanés, S. E. Shladover, and J. Spring, "Cooperative Adaptive Cruise Control in Real Traffic Situations," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 1, pp. 296–305, 2013.

[32] E. van Nunen, M. R. J. A. E. Kwakkernaat, J. Ploeg, and B. D. Netten, "Cooperative Competition

for Future Mobility," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1018–1025, 2012.

[33] S. E. Shladover, C. Nowakowski, X.-Y. Lu, and R. Ferlis, "Cooperative adaptive cruise control: definitions and operating concepts," *Transp. Res. Rec.*, 2015.

[34] J. Ploeg, E. Semsar-Kazerooni, G. Lijster, N. Van De Wouw, and H. Nijmeijer, "Graceful degradation of cooperative adaptive cruise control," *IEEE Trans. Intell. Transp. Syst.*, vol. 16, no. 1, pp. 488–497, 2015.

[35] R. Rajamani, *Vehicle dynamics and control*. Springer Science & Business Media, 2011.

[36] C. Flores and V. Milanés, "Fractional-order-based ACC/CACC algorithm for improving string stability," *Transp. Res. Part C Emerg. Technol.*, vol. 95, pp. 381–393, 2018.

[37] L. E. Peppard, "String Stability of Relative-Motion PID Vehicle Control Systems," *IEEE Trans. Automat. Contr.*, 1974.

[38] Y. Zhou, M. Wang, and S. Ahn, "Distributed model predictive control approach for cooperative car-following with guaranteed local and string stability," *Transp. Res. Part B Methodol.*, 2019.

[39] L. Xiao and F. Gao, "Practical string stability of platoon of adaptive cruise control vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 4, pp. 1184–1194, 2011.

[40] G. Gunter *et al.*, "Are commercially implemented adaptive cruise control systems string stable?," vol. 19122, pp. 1–22, 2019.

[41] V. L. Knoop, M. Wang, I. Wilmink, D. M. Hoedemaeker, M. Maaskant, and E. J. Van der Meer, "Platoon of SAE Level-2 Automated Vehicles on Public Roads: Setup, Traffic Interactions, and Stability," *Transp. Res. Rec.*, 2019.

[42] B. Van Arem, C. J. G. Van Driel, and R. Visser, "The impact of cooperative adaptive cruise control on traffic-flow characteristics," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 429–436, 2006.

[43] Z. Wang, G. Wu, and M. J. Barth, "A Review on Cooperative Adaptive Cruise Control (CACC) Systems: Architectures, Controls, and Applications," in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 2018.

[44] W. S. Levine and M. Athans, "On the Optimal Error Regulation of a String of Moving Vehicles," *IEEE Trans. Automat. Contr.*, vol. 11, no. 3, pp. 355–361, 1966.

[45] V. Milanes, S. E. Shladover, J. Spring, C. Nowakowski, H. Kawazoe, and M. Nakamura, "Cooperative adaptive cruise control in real traffic situations," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 1, pp. 296–305, 2014.

[46] J. Mårtensson *et al.*, "The Development of a Cooperative Heavy-Duty Vehicle for the GCDC 2011: Team Scoop," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1033–1049, 2012.

[47] S. J. Qin and T. A. Badgwell, "A survey of industrial model predictive control technology," *Control Eng. Pract.*, 2003.

[48] M. Wang, W. Daamen, S. P. Hoogendoorn, and B. van Arem, "Rolling horizon control framework for driver assistance systems. Part II: Cooperative sensing and cooperative control," *Transp. Res. Part C Emerg. Technol.*, 2014.

[49] C. Englund *et al.*, “The Grand Cooperative Driving Challenge 2016: Boosting the introduction of cooperative automated vehicles,” *IEEE Wirel. Commun.*, 2016.

[50] H. Xing, J. Ploeg, and H. Nijmeijer, “Compensation of Communication Delays in a Cooperative ACC System,” *IEEE Trans. Veh. Technol.*, 2020.

[51] C. Wang, S. Gong, A. Zhou, T. Li, and S. Peeta, “Cooperative adaptive cruise control for connected autonomous vehicles by factoring communication-related constraints,” *Transp. Res. Part C Emerg. Technol.*, 2019.

[52] L. Cui, J. Hu, B. B. Park, and P. Bujanovic, “Development of a simulation platform for safety impact analysis considering vehicle dynamics, sensor errors, and communication latencies: Assessing cooperative adaptive cruise control under cyber attack,” *Transp. Res. Part C Emerg. Technol.*, 2018.

[53] Z. A. Biron, S. Dey, and P. Pisu, “Real-Time Detection and Estimation of Denial of Service Attack in Connected Vehicle Systems,” *IEEE Trans. Intell. Transp. Syst.*, pp. 1–10, 2018.

[54] Z. Chen and B. B. Park, “Preceding Vehicle Identification for Cooperative Adaptive Cruise Control Platoon Forming,” *IEEE Trans. Intell. Transp. Syst.*, pp. 1–13, 2019.

[55] Q. Wang, X. (Terry) Yang, Z. Huang, and Y. Yuan, “Multi-Vehicle Trajectory Design During Cooperative Adaptive Cruise Control Platoon Formation,” *Transp. Res. Rec. J. Transp. Res. Board*, 2020.

[56] A. Duret, M. Wang, and A. Ladino, “A hierarchical approach for splitting truck platoons near network discontinuities,” *Transp. Res. Part B Methodol.*, 2020.

[57] R. A. Singer, “Estimating Optimal Tracking Filter Performance for Manned Maneuvering Targets,” *IEEE Trans. Aerosp. Electron. Syst.*, 1970.

[58] J. I. Ge and G. Orosz, “Dynamics of connected vehicle systems with delayed acceleration feedback,” *Transp. Res. Part C Emerg. Technol.*, vol. 46, pp. 46–64, 2014.

[59] L. Zhang, “Motif-Based Design for Connected Vehicle Systems in Presence of Heterogeneous Connectivity Structures and Time Delays,” vol. 17, no. 6, pp. 1638–1651, 2016.

[60] L. Zhang and G. Orosz, “Beyond-Line-of-Sight Identification by Using Vehicle-to-Vehicle Communication,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 6, pp. 1962–1972, 2018.

[61] J. I. Ge and G. Orosz, “Connected cruise control among human-driven vehicles: Experiment-based parameter estimation and optimal control design,” *Transp. Res. Part C Emerg. Technol.*, vol. 95, no. August 2017, pp. 445–459, 2018.

[62] J. I. Ge, S. S. Avedisov, C. R. He, W. B. Qin, and M. Sadeghpour, “Experimental validation of connected automated vehicle design among human-driven vehicles,” *Transp. Res. Part C*, vol. 91, no. April, pp. 335–352, 2018.

[63] N. Chen, M. Wang, T. Alkim, and B. Van Arem, “A Robust Longitudinal Control Strategy of Platoons under Model Uncertainties and Time Delays,” *J. Adv. Transp.*, vol. 2018, pp. 1–13, 2018.

[64] H. Liu, X. (David) Kan, S. E. Shladover, X. Y. Lu, and R. E. Ferlis, “Modeling impacts of Cooperative Adaptive Cruise Control on mixed traffic flow in multi-lane freeway facilities,” *Transp. Res. Part C Emerg. Technol.*, vol. 95, no. December 2017, pp. 261–279, 2018.

[65] A. Talebpour and H. S. Mahmassani, “Influence of connected and autonomous vehicles on traffic flow stability and throughput,” *Transp. Res. Part C Emerg. Technol.*, 2016.

[66] I. J. Reagan, D. G. Kidd, and J. B. Cicchino, “Driver acceptance of adaptive cruise control and active lane keeping in five production vehicles,” in *Proceedings of the Human Factors and Ergonomics Society*, 2017.

[67] H. Rakha and R. K. Kamalanathsharma, “Eco-driving at signalized intersections using V2I communication,” in *IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC*, 2011.

[68] B. L. Smith and H. Park, “Investigating Benefits of IntelliDrive in Freeway Operations: Lane Changing Advisory Case Study,” *J. Transp. Eng.*, 2012.

[69] H. S. Tan and J. Huang, “DGPS-based vehicle-to-vehicle cooperative collision warning: Engineering feasibility viewpoints,” *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 415–427, 2006.

[70] J. Deur, D. Pavković, N. Perić, M. Jansz, and D. Hrovat, “An electronic throttle control strategy including compensation of friction and limp-home effects,” *IEEE Trans. Ind. Appl.*, 2004.

[71] Bosch, “ESP Generation 9.” [Online]. Available: https://www.bosch-mobility-solutions.com/en/products-and-services/passenger-cars-and-light-commercial-vehicles/driving-safety-systems/electronic-stability-program/esp-generation-9/. [Accessed: 12-Mar-2019].

[72] W.-D. Jonner, H. Winner, L. Dreilich, and E. Schunck, “Electrohydraulic Brake System - The First Approach to Brake-By-Wire Technology,” in *SAE Technical Paper Series*, 1996.

[73] W. Xiang, P. C. Richardson, C. Zhao, and S. Mohammad, “Automobile brake-by-wire control system design and analysis,” *IEEE Trans. Veh. Technol.*, 2008.

[74] A. M. H. Al-Jhayyish and K. W. Schmidt, “Feedforward Strategies for Cooperative Adaptive Cruise Control in Heterogeneous Vehicle Strings,” *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 1, pp. 113–122, 2018.

[75] J. I. Ge and G. Orosz, “Optimal Control of Connected Vehicle Systems With Communication Delay and Driver Reaction Time,” *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 8, pp. 2056–2070, 2017.

[76] T. J. Ayres, L. Li, D. Schleuning, and D. Young, “Preferred time-headway of highway drivers,” in *ITSC 2001. 2001 IEEE Intelligent Transportation Systems. Proceedings (Cat. No.01TH8585)*, 2001, pp. 826–829.

[77] G. Orosz, R. E. Wilson, and G. Stepan, “Traffic jams: dynamics and control,” *Philos. Trans. R. Soc. A Math. Phys. Eng. Sci.*, vol. 368, no. 1928, pp. 4455–4479, 2010.

[78] A. Mehmood and S. M. Easa, “Modeling Reaction Time in Car-Following Behaviour Based on Human Factors,” *Civil, Environ. Struct. Constr. Archit. Eng.*, vol. 3, no. 9, pp. 325–333, 2009.

[79] J. I. Ge, G. Orosz, D. Hajdu, T. Insperger, and J. Moehlis, *Time Delay Systems*, vol. 7. 2017.

[80] R. Rajamani, *Vehicle Dynamics and Control*. 2006.

[81] M. R. I. Nieuwenhuijze, T. van Keulen, S. Öncü, B. Bonsen, and H. Nijmeijer, “Cooperative Driving With a Heavy-Duty Truck in Mixed Traffic: Experimental Results,” *IEEE Trans. Intell.*

Transp. Syst., vol. 13, no. 3, pp. 1026–1032, 2012.

[82] V. Alexiadis, J. Colyar, J. Halkias, R. Hranac, and G. McHale, "The next generation simulation program," *ITE J. (Institute Transp. Eng.*, vol. 74, no. 8, pp. 22–26, 2004.

[83] M. Tideman, "Scenario-Based Simulation Environment for Assistance Systems," *ATZautotechnology*, vol. 10, no. 1, pp. 28–32, Jan. 2010.

[84] H. A. Rakha, K. Ahn, K. Moran, B. Saerens, and E. Van Den Bulck, "Virginia Tech Comprehensive Power-Based Fuel Consumption Model : Model development and testing," *Transp. Res. Part D*, vol. 16, no. 7, pp. 492–503, 2011.

[85] V. Punzo, M. T. Borzacchiello, and B. Ciuffo, "On the assessment of vehicle trajectory data accuracy and application to the Next Generation SIMulation (NGSIM) program data," *Transp. Res. Part C Emerg. Technol.*, vol. 19, no. 6, pp. 1243–1262, 2011.

[86] J. Hasch, E. Topak, R. Schnabel, T. Zwick, R. Weigel, and C. Waldschmidt, "Millimeter-wave technology for automotive radar sensors in the 77 GHz frequency band," *IEEE Trans. Microw. Theory Tech.*, vol. 60, no. 3 PART 2, pp. 845–860, 2012.

[87] S. Li, K. Li, R. Rajamani, and J. Wang, "Model predictive multi-objective vehicular adaptive cruise control," *IEEE Trans. Control Syst. Technol.*, 2011.

[88] T. Stanger and L. Del Re, "A model predictive Cooperative Adaptive Cruise Control approach," in *Proceedings of the American Control Conference*, 2013.

[89] H. Fukushima, T. H. Kim, and T. Sugie, "Adaptive model predictive control for a class of constrained linear systems based on the comparison model," *Automatica*, 2007.

[90] G. Welch and G. Bishop, "An Introduction to the Kalman Filter," *In Pract.*, 2006.

[91] V. Milanés and S. E. Shladover, "Modeling cooperative and autonomous adaptive cruise control dynamic responses using experimental data," *Transp. Res. Part C Emerg. Technol.*, 2014.

[92] F. l. Lewis, L. Xie, and D. Popa, *Optimal and Robust Estimation: With an Introduction to Stochastic Control Theory*. 2007.

[93] C. Schmid and L. T. Biegler, "Quadratic programming methods for reduced hessian SQP," *Comput. Chem. Eng.*, 1994.

[94] A. Bemporad, M. Morari, V. Dua, and E. N. Pistikopoulos, "The explicit linear quadratic regulator for constrained systems," *Automatica*, 2002.

[95] R. Kianfar *et al.*, "Design and Experimental Validation of a Cooperative Driving System in the Grand Cooperative Driving Challenge," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 994–1007, 2012.

[96] J. I. Ge and G. Orosz, "Data-driven parameter estimation for optimal connected cruise control," in *2017 IEEE 56th Annual Conference on Decision and Control, CDC 2017*, 2018, vol. 2018-Janua, no. 1351456, pp. 3739–3744.

[97] M. M. Minderhoud and P. H. L. Bovy, "Extended time-to-collision measures for road traffic safety assessment," *Accid. Anal. Prev.*, vol. 33, no. 1, pp. 89–97, 2001.

[98] N. H. T. S. Administration, "Forward collision warning system confirmation test," *Off. Veh.*

Safety, Off. Crash Avoid. Stand. Natl. Highw. Traffic Saf. Adm. Washington, DC, 2013.

[99] V. A. Akpan and G. D. Hassapis, “Nonlinear model identification and adaptive model predictive control using neural networks,” *ISA Trans.*, 2011.

[100] A. Khodayari, A. Ghaffari, R. Kazemi, and R. Braunstingl, “A modified car-following model based on a neural network model of the human driver effects,” *IEEE Trans. Syst. Man, Cybern. Part ASystems Humans*, 2012.