A CAV Cooperative Lane Change Protocol With CTH Safety Guarantee on Dedicated Highways

Xueli Fan, Jieming Chen, Qixin Wang, and Edward Chung

***Abstract*—Autopiloting *Connected and Autonomous Vehicles* (CAVs) is an important application for mobile computing. A promising context to realize autopiloting CAVs is cooperative driving on dedicated highways. For such a context, an indispensable driving scenario is *Cooperative Lane Change* (CLC). Due to the safety concerns of this driving scenario, a verifiably safe solution is needed (at least, the solution design should be formally provably safe). However, this demand is complicated by the inherently unreliable wireless communications between the CAVs. In this paper, we focus on the well-adopted *Constant Time Headway* (CTH) safety rule. We propose a CLC protocol, and formally prove its guarantee of the CTH safety and liveness, even under arbitrary wireless packet losses. These theoretical claims are further confirmed by our simulations. The simulation results also show that our proposed protocol significantly improves lane change success rates (by $5.3\% \sim +\infty\%$) than other alternatives under adverse conditions. Furthermore, the sensitivity study results also show our protocol can tolerate reasonable disturbances.**

***Index Terms*—CPS, hybrid automata, lane change, verifiable safety.**

I. Introduction

AUTOPILOTING *Connected and Autonomous Vehicles* (CAVs) is an important application of mobile computing. To realize autopiloting CAVs, *cooperative driving* of CAVs on *dedicated highways* is a (if not the most) promising context [1], [2], [3]. Even so, in this context, the realization of cooperative CAV driving is still non-trivial, and must be built upon individual cooperation protocols for specific driving scenarios. In this paper, we focus on one of such scenarios: *Cooperative Lane Change* (CLC).

Lane change is one of the most common driving scenarios, hence CLC of CAVs has naturally become a hot research topic [4], [5], [6]. However, how to guarantee the CLC safety is challenged by the inherently unreliable wireless communications [4], [5], [7], [8]. Particularly, wireless packets can be arbitrarily lost due to various reasons, such as malicious or unintentional jamming, large scale path-loss, multipath, hand-over, contention etc. This is evidenced by many field studies [9], [10], [11], [12], [13], [14], [15]. Solutions to address this challenge heavily depend on the *targeted safety rule* and the *chosen wireless communications paradigm*.

For the targeted safety rule, in this paper, we focus on the widely adopted *Constant Time Headway* (CTH) safety rule [16], [17], [18]. This rule requires any two consecutive vehicles on a same lane (referred to respectively as the "*leader*" and the "*follower*") maintain a spatial distance proportional to the follower's current speed.

For the wireless communications paradigm, there are two major alternatives: *Vehicle to Vehicle* (V2V) or *Vehicle to Infrastructure* (V2I). In this paper, we focus on the V2V paradigm, due to its lower demand on infrastructure investment.

In summary, this paper shall propose a V2V CAV CLC protocol on dedicated highways. This protocol shall guarantee the CTH safety under arbitrary wireless packet losses: a challenge still lacks attention nowadays. In the large volume of literature on CLC [4], [5], [6], works focusing on the challenge of wireless packet losses are relatively few.

Specifically, in Tsugawa et al.'s paper on the Demo 2000 cooperative driving of automated vehicles [19], wireless packet losses are reported. However, how to deal with the wireless packet losses is not the focus of the paper, hence is not elaborated.

Sakr et al. [20] propose to use open-loop Kalman filter to predict the position of remote vehicles when wireless packets are lost. But the focus is to detect intention of lane change, instead of guaranteeing CTH safety.

Wang et al. [21] present two CLC protocols (PerLC and ConLC) for V2V CAVs. The main idea of the protocols is to use simple acknowledgements (ACKs) to fight against arbitrary wireless packet losses. If an ACK is not received within a predefined period, the wireless packet will be retransmitted. The

Received 19 April 2024; revised 13 March 2025; accepted 26 March 2025. Date of publication 8 April 2025; date of current version 6 August 2025. The work was supported in part by HK RGC under Grant T22-505/19-N (aka P0031331, RBCR, P0031259, RBCP), Grant PolyU 152002/18E (aka P0005550, Q67V), Grant PolyU 152164/14E (aka P0004750, Q44B), Grant GRF 15207324 (aka P0051926, B-QCFM), Grant G-PolyU503/16, in part by HKSAR Government and HKJCCT under Grant P0041424 (aka ZB5A), and in part by the HK PolyU under Grant P0042701 (aka CE09), Grant P0046487 (aka CE0F), Grant P0047916 (aka TACW), Grant P0042699 (aka CE55), Grant P0045578 (aka CE1C), Grant P0043884 (aka CD6R), Grant P0047965 (aka TAEB), Grant P0047964 (aka TAEA), Grant P0033695 (aka ZVRD), Grant P0013879 (aka BBWH), Grant P0036469 (aka CDA8), Grant P0043634 (aka 1-TAB2), Grant P0043647 (aka 1-TABF), Grant P0042721 (aka 1-ZVG0), Grant LTG22-25/IICA/33 (aka TDG 2022-25), and Grant TDG22-25/SMS-11. Recommended for acceptance by M. Segata. *(Xueli Fan and Jieming Chen contributed equally to this work.) (Corresponding author: Qixin Wang.)*

Xueli Fan is with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR, China, and also with the School of Cyber Science and Technology, Shenzhen Campus of Sun Yat-sen University, Shenzhen 518107, China (e-mail: xue.li.fan@connect.polyu.hk).

Jieming Chen and Edward Chung are with the Department of Electrical and Electronic Engineering, The Hong Kong Polytechnic University, Hong Kong SAR, China.

Qixin Wang is with the Department of Computing, The Hong Kong Polytechnic University, Hong Kong SAR, China (e-mail: qixin.wang@polyu.edu.hk).

This article has supplementary downloadable material available at https://doi.org/10.1109/TMC.2025.3558922, provided by the authors.

Digital Object Identifier 10.1109/TMC.2025.3558922

mechanical lane change routine cannot start unless all the needed ACKs are received. Although arbitrary wireless packet losses are considered in [21], how to adapt the protocols to guarantee the CTH safety is still an open problem.

Different from the above works, this paper focuses on guaranteeing the CTH safety under *arbitrary* wireless packet losses. This is achieved via exploiting the timeout (aka leasing [22], [23]) design philosophy for distributed systems. The basic idea is to properly configure a set of timeout deadlines on each participant at the start of each cooperation. During the cooperation, if the needed wireless packets cannot arrive before the timeout deadlines, the participants will independently reset themselves, hence implicitly reset the whole system. Specifically, we made the following contributions.

1) We propose a timeout based CLC protocol for V2V CAVs on dedicated highways.
2) We formally prove the CTH safety guarantee and liveness of our proposed protocol, even under arbitrary wireless packet losses.
3) We carry out simulations to verify our proposed protocol. The results show that our protocol can always fulfill the CTH safety and liveness despite of arbitrary wireless packet losses; and compared to other alternatives, our protocol can achieve significantly higher ($5.3\% \sim +\infty\%$ improvements[1]) lane change success rates under adverse conditions. Furthermore, the sensitivity study results also show that our protocol can tolerate reasonable disturbances.

The rest of the paper is organized as follows. Section II presents related work. Section III formulates the problem. Section IV proposes the protocol and formally proves its properties. Section V discusses some important observations. Section VI evaluates our proposed protocol. Section VII concludes the paper.

II. Related Work

Despite of the works by Tsugawa et al. [19], Sakr et al. [20], and Wang et al. [21], which are elaborated in Section I, there are many other works on CAV CLC.

Li et al. [24] propose a CLC trajectory planning workflow, where the CAVs dynamically form groups, plan trajectories (of both the requesting vehicle and the surrounding vehicles) group-by-group as optimization problems. The planning needs to be periodically updated, to adapt to runtime changes. Because of this, this workflow assumes wireless communications are always available; i.e. it does not deal with arbitrary wireless packet losses. Also, CTH safety guarantee is not the focus, and is not discussed.

Similar to Li et al. [24], Luo et al. [25] also formulate the CAV CLC problem as a periodical trajectory planning optimization problem. The solution supports multiple requesting vehicles (i.e. multiple vehicle requesting to change lane); and it proposes a model predictive controller to track the planned trajectories. Like Li et al. [24], Luo et al. [25] also assume wireless communications are always available, i.e. there is no wireless packet loss; and CTH safety guarantee is not discussed.

Many works model CLC as convoy formation control problems [26], [27], [28]. These works typically take a global view, grouping all states (which can be position, velocity, heading, angular velocity, etc.) of involved CAVs as the state of a global control problem; and group all control outputs (which can be throttle, acceleration, steering, etc.) of the involved CAVs as the control output of the global control problem. These works may differ on the controller type, control objectives, and control constraints. But what is common is that the global control is conducted in real-time: the global control problem's states and control outputs are collected/distributed from/to the involved CAVs periodically, and the period is orders of magnitude smaller than the time head ways needed for safety. For such real-time control to work, all the involved CAVs must remain connected to each other. Therefore, wireless communications are assumed to be always available. To our best knowledge, none of these solutions discusses how to guarantee CTH safety under arbitrary wireless packet losses.

Also among the works that model CLC as convoy formation control problems, a new trend is to use machine learning to train the controller for the control problem [29], [30]. These solutions depend even more on the availability of the wireless communications. In addition to collect/distribute real-time states/controls from/to the involved CAVs, the wireless communications also need to collect/distribute data for runtime machine learning. To our best knowledge, how to guarantee CTH safety under arbitrary wireless packet losses is still an open problem for these solutions.

The concern for wireless packet loss is real.

Specifically, the V2X (Vehicle-to-Everything: including V2V and V2I) standardization communities have published many use cases of CAV cooperative driving (including CAV CLC) [5], [31], [7]. These use cases specify the demands on the V2X wireless communications qualities. Particularly, the *3 rd Generation Partnership Project* (3GPP) standards specify that to support fully automated CLC, the V2X wireless communications must guarantee a *Packet Error Rate* (PER) of no more than 0.01% and a delay of no more than 10ms [32], [33]. However, the standards do not tell us how to achieve these demanded wireless communications qualities. In practice, nowadays V2X implementations still *cannot always guarantee* such qualities.

For example, in the field study by Hüsges et al. [9], a pair of LTE V2X [7] *On-Board Units* (OBUs) (Cohda Wireless MK6C EVK [34]) are mounted respectively on two vehicles to carry out V2V communications. Even in the most benign scenario, where the two vehicles are placed on a straight road in open space and the two OBUs' antennas have *Line-of-Sight* (LOS), the PER can still exceed 10% when the inter-vehicle distance is between 85 m to 95 m. This PER is 1000 times of the worst acceptable CLC PER (0.01%) specified by the 3GPP standard [32], [33]. When the two antennas have *Non-Line-of-Sight* (NLOS) (this can happen when a big enough lane changing CAV cuts between two CAVs), PER of 100% (i.e. 10000 times the worst acceptable CLC PER) can even happen at some distances below 100 m.

[1]The $+\infty\%$ improvement is achieved when the other alternatives achieve 0 success rates, while ours achieves positive success rates, see Fig. 7.

In the field study by Lusvarghi et al. [10], a pair of LTE V2X OBUs (Quectel AG15 [35]) are mounted respectively on two vehicles to carry out V2V communications. In the urban road settings, the PER can exceed 4% (i.e. 400 times of the worst acceptable CLC PER of 0.01%) when the inter-vehicle distance is within 150 m.

In the field study by Bréhon-Grataloup et al. [11], a campus road network test site with various V2X technologies is implemented using the Cohda Wireless MK6 [36] *Road Side Units* (RSUs) and OBUs. The measured packet delays can often exceed 100 ms (i.e. 10 times the worst acceptable CLC V2X communication delay, 10 ms, specified by the 3GPP standard [32], [33]).

In the field study by Kim et al. [12], the authors implemented their own 5G NR [37], [38] compatible RSUs and OBUs, and tested in a highway setting. The probability that the handover delay exceeding 10 ms (the worst acceptable CLC V2X communication delay) is more than 50%. Sometimes the delay may even exceed 22 ms.

The above field study results are also corroborated by field *Signal-to-Noise Ratio* (SNR) measurements [13], [14], lab experiment studies [39], [40], and large volume of analytical/simulation based studies [41], [42], [43]. Particularly, as per An et al. [41]'s theoretical analysis (and the corresponding simulation), when SNR drops below 9.44 dB, 5G NR [37], [38] V2V communications' PER can already exceed 1% (i.e. 100 times the worst acceptable CLC PER). When SNR drops below 1.67 dB, the PER can even reach 100% (i.e. 10000 times the worst acceptable CLC PER). Meanwhile, as per Mikami et al. [14]'s field study on a truck platoon on open space highway, the 5G NR V2V communications SNR in the field can often drop below 9.44 dB, sometimes even below 0 dB (meaning the signal is weaker than the noise).

In fact, theoretically speaking, no matter how robust a wireless communication link is, it can always fail given enough strong jamming noises; and such failure can last for as long as the jamming noises persist (i.e. a jamming attack, a kind of Denial-of-Service attack) [15], [40].

This paper, therefore, focuses on the context when the V2X (V2V) wireless communications cannot meet the qualities demanded by the standards [32], [33], [31]. Hence it is different from all other works that assume the standards' demanded wireless communication qualities hold.

The content of this paper has been included in the PhD dissertation of the first author [44].

III. Problem Formulation

In this section, we specify the lane change scenario, related assumptions, CTH safety rule, and the research problem.

A. Lane Change Scenario

Fig. 1 illustrates this paper's focus lane change scenario.

This scenario includes two neighboring lanes of interest along a dedicated highway. Without loss of generality, name one of the lanes as the "*current lane*," and the other lane as the "*target lane*" for the lane change. These two lanes are abstracted as two parallel axes in Fig. 1, whose positive direction defines the X-axis direction of our global fixed-to-ground X-Y coordinates. In this coordinate system, for simplicity and for the time being, let us abstract every CAV as a point mathematically (see the $\circ$ dots in Fig. 1), and let $\vec{p}(x,t)$ denote the location of CAV x at *world time instance* (simplified as "*time instance*" in the following unless otherwise denoted) t. When not conducting the mechanical lane change routine, a CAV point should drive along either the "current lane" axis or the "target lane" axis. Considerations on the vehicle body length are discussed in Section III-C (the paragraph right after Def. III-C) and Section V-B.

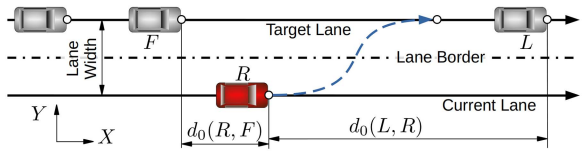


Fig. 1. Lane change scenario.

Suppose our lane change scenario starts from time instance t_0, when all CAVs on the dedicated highway are driving at the stable default speed v_{lim}, where v_{lim} is a preconfigured constant. Note adopting such a v_{lim} is a popular practice by many cooperative CAV driving schemes [45], [46], [47], [48], particularly in the large volume of literature on *Cooperative Adaptive Cruise Control* (CACC) [49], [50]; this practice prevails not only for its simplicity, but also for its safety and energy efficiency [47], [48], [49], [50].

On the target lane, at t_0, we assume there are n ($n \in \mathbb{N}^0$) CAVs, consecutively denoted as V_i ($i = 1, 2, \ldots, n$, with V_1 as the front most CAV on the target lane). We call the V_is the "*target lane CAVs*."

Meanwhile, on the current lane, at a time instance $t'_0 \geqslant t_0$ ($t'_0 < +\infty$), a CAV R intends to change to the target lane. We call R the "*requesting CAV*." At t'_0, R recognizes the closest (along the X-axis) target lane CAV before R, denote that target lane CAV as L ($L \stackrel{\mathsf{def}}{=} \varnothing$ if it does not exist); and recognizes the closest (along the X-axis) target lane CAV not before (i.e. after or at the same X-coordinate as) R, denote that target lane CAV as F ($F \stackrel{\mathsf{def}}{=} \varnothing$ if it does not exist). We call L the "*leader CAV*," and F the "*follower CAV*." We denote the distances from R to L and F along the X-axis (i.e. distance projections on the X-axis) at t'_0 as $d_0(L,R)$ and $d_0(R,F)$ respectively ($d_0(L,R) \stackrel{\mathsf{def}}{=} +\infty$ if $L = \varnothing$, and $d_0(R,F) \stackrel{\mathsf{def}}{=} +\infty$ if $F = \varnothing$).

B. Assumptions on CAV Driving Dynamics

Vehicular driving dynamics modeling is nontrivial (interested readers can refer to the classic textbook of [51]). Fortunately, we can make the following reasonable and generic enough assumptions for our CAVs.

1) Mechanical Lane Change Routine: Suppose at time t^0_{lc}, a CAV starts the mechanical lane change routine with an initial speed of $v^0_{\mathsf{lc}} \in [v^{\mathsf{low}}_{\mathsf{lc}}, v^{\mathsf{high}}_{\mathsf{lc}}]$ (from the initial orientation of pointing toward the positive direction of the X-axis), where $v^{\mathsf{high}}_{\mathsf{lc}} > v^{\mathsf{low}}_{\mathsf{lc}} > 0$ are respectively the upper bound and lower bound of v^0_{lc}.

Based on the dynamic model of CAV in the classic textbook of [51], suppose currently the mechanical lane change routine has been going on for τ seconds ($\tau \geq 0$) and has not yet finished, then the CAV's current velocity (i.e. speed and direction) is determined by a function of v_{lc}^0 and τ. Denote this function as $\mathsf{lc}(v_{\mathsf{lc}}^0, \tau)$. This in turn implies the total duration and projected X-axis distance experienced by a completed mechanical lane change routine are respectively determined by functions of v_{lc}^0, which can be respectively denoted as $\delta_{\mathsf{lc}}(v_{\mathsf{lc}}^0)$ and $d_{\mathsf{lc}}^{\mathsf{X}}(v_{\mathsf{lc}}^0)$. Note the total projected Y-axis distance experienced by a completed mechanical lane change routine is always the lane width.

Furthermore, according to [51], we can make the following reasonable and generic enough assumption:

Assumption 1: We assume throughout a mechanical lane change routine of a CAV, the CAV's speed (i.e. velocity magnitude) is constant, and the total duration $\delta_{\mathsf{lc}}(v_{\mathsf{lc}}^0)$ needed to complete the mechanical lane change routine is a strictly monotonically decreasing function of the initial speed v_{lc}^0. That is, the faster the initial speed, the faster the mechanical lane change routine completes. As $v_{\mathsf{lc}}^0 \in [v_{\mathsf{lc}}^{\mathsf{low}}, v_{\mathsf{lc}}^{\mathsf{high}}]$, we have $\delta_{\mathsf{lc}}(v_{\mathsf{lc}}^{\mathsf{high}}) < \delta_{\mathsf{lc}}(v_{\mathsf{lc}}^{\mathsf{low}})$ and $\delta_{\mathsf{lc}}(v_{\mathsf{lc}}^0) \in [\delta_{\mathsf{lc}}(v_{\mathsf{lc}}^{\mathsf{high}}), \delta_{\mathsf{lc}}(v_{\mathsf{lc}}^{\mathsf{low}})]$. ■

The above assumption also implies the projected X-axis speed of the mechanical lane change routine first decreases from v_{lc}^0, and then increases back to v_{lc}^0, hence

$$v_{\mathsf{lc}}^0 \delta_{\mathsf{lc}}(v_{\mathsf{lc}}^0) > d_{\mathsf{lc}}^{\mathsf{X}}(v_{\mathsf{lc}}^0). \tag{1}$$

2) CAV Acceleration Routine: Besides the accelerations/decelerations conducted during mechanical lane change routines, a CAV may also conduct straight line (i.e. along the X-axis) accelerations/decelerations. *To simplify narration, in the following, unless explicitly denoted, the term "acceleration" and "deceleration" shall only refer to such straight line accelerations/decelerations; correspondingly, in the straight line driving context, we do not differentiate the term "speed" and "velocity."*

We assume in dedicated highway cooperative driving of CAVs, all CAVs' acceleration strategies can be fixed, hence can be called "*acceleration routines*." Specifically, given the initial speed $v_{\mathsf{a}}^{\mathsf{low}}$ and the target speed $v_{\mathsf{a}}^{\mathsf{high}}$, suppose currently the acceleration routine has been going on for τ seconds ($\tau \geqslant 0$) and has not yet finished, then the CAV's current acceleration value is fixed, and is a function of $v_{\mathsf{a}}^{\mathsf{low}}$, $v_{\mathsf{a}}^{\mathsf{high}}$, and τ. Denote this function as $\mathsf{acc}(v_{\mathsf{a}}^{\mathsf{low}}, v_{\mathsf{a}}^{\mathsf{high}}, \tau)$. This function in turn implies that the current speed of the CAV is a function of $v_{\mathsf{a}}^{\mathsf{low}}$, $v_{\mathsf{a}}^{\mathsf{high}}$, and τ, which can be denoted as $v_{\mathsf{a}}(v_{\mathsf{a}}^{\mathsf{low}}, v_{\mathsf{a}}^{\mathsf{high}}, \tau)$. This in turn implies that the total duration and distance needed to accelerate from $v_{\mathsf{a}}^{\mathsf{low}}$ to $v_{\mathsf{a}}^{\mathsf{high}}$ is a function of $v_{\mathsf{a}}^{\mathsf{low}}$ and $v_{\mathsf{a}}^{\mathsf{high}}$. We can denote this duration and this distance to be respectively $\delta_{\mathsf{a}}(v_{\mathsf{a}}^{\mathsf{low}}, v_{\mathsf{a}}^{\mathsf{high}})$ and $d_{\mathsf{a}}(v_{\mathsf{a}}^{\mathsf{low}}, v_{\mathsf{a}}^{\mathsf{high}})$ (for more sophisticated details on vehicle acceleration, interested readers can refer to [51]).

Furthermore, we have the following assumption:

Assumption 2: Given $0 \leqslant v_{\mathsf{a}}^{\mathsf{low}} < v_{\mathsf{a}}^{\mathsf{high}}$, the acceleration routine as per $\mathsf{acc}(v_{\mathsf{a}}^{\mathsf{low}}, v_{\mathsf{a}}^{\mathsf{high}}, \tau)$ is always nonzeno (i.e. $\delta_{\mathsf{a}}(v_{\mathsf{a}}^{\mathsf{low}}, v_{\mathsf{a}}^{\mathsf{high}}) > 0$) and the speed will strictly monotonically increase from $v_{\mathsf{a}}^{\mathsf{low}}$ to $v_{\mathsf{a}}^{\mathsf{high}}$. ■

3) CAV Deceleration Routine: Similar to acceleration, we also assume the CAV deceleration strategies are fixed, hence can be called "*deceleration routines*." Specifically, given the initial speed $v_{\mathsf{d}}^{\mathsf{high}}$ and the target speed $v_{\mathsf{d}}^{\mathsf{low}}$, suppose currently the deceleration routine has been going on for τ seconds ($\tau \geqslant 0$) and has not yet finished, then the CAV's current acceleration value is fixed, and is a function of $v_{\mathsf{d}}^{\mathsf{high}}$, $v_{\mathsf{d}}^{\mathsf{low}}$, and τ. Denote this function as $\mathsf{dec}(v_{\mathsf{d}}^{\mathsf{high}}, v_{\mathsf{d}}^{\mathsf{low}}, \tau)$. This function in turn implies that the current speed of the CAV is a function of $v_{\mathsf{d}}^{\mathsf{high}}$, $v_{\mathsf{d}}^{\mathsf{low}}$, and τ, which can be denoted as $v_{\mathsf{d}}(v_{\mathsf{d}}^{\mathsf{high}}, v_{\mathsf{d}}^{\mathsf{low}}, \tau)$. This in turn implies that the total duration and distance needed to decelerate from $v_{\mathsf{d}}^{\mathsf{high}}$ to $v_{\mathsf{d}}^{\mathsf{low}}$ is a function of $v_{\mathsf{d}}^{\mathsf{high}}$ and $v_{\mathsf{d}}^{\mathsf{low}}$. We can denote this duration and this distance to be respectively $\delta_{\mathsf{d}}(v_{\mathsf{d}}^{\mathsf{high}}, v_{\mathsf{d}}^{\mathsf{low}})$ and $d_{\mathsf{d}}(v_{\mathsf{d}}^{\mathsf{high}}, v_{\mathsf{d}}^{\mathsf{low}})$.

Furthermore, we have the following assumption:

Assumption 3: Given $v_{\mathsf{d}}^{\mathsf{high}} > v_{\mathsf{d}}^{\mathsf{low}} \geqslant 0$, the deceleration routine as per $\mathsf{dec}(v_{\mathsf{d}}^{\mathsf{high}}, v_{\mathsf{d}}^{\mathsf{low}}, \tau)$ is always nonzeno (i.e. $\delta_{\mathsf{d}}(v_{\mathsf{d}}^{\mathsf{high}}, v_{\mathsf{d}}^{\mathsf{low}}) > 0$) and the speed will strictly monotonically decrease from $v_{\mathsf{d}}^{\mathsf{high}}$ to $v_{\mathsf{d}}^{\mathsf{low}}$. ■

C. Other Assumptions, CTH Safety Rule, and the Research Problem

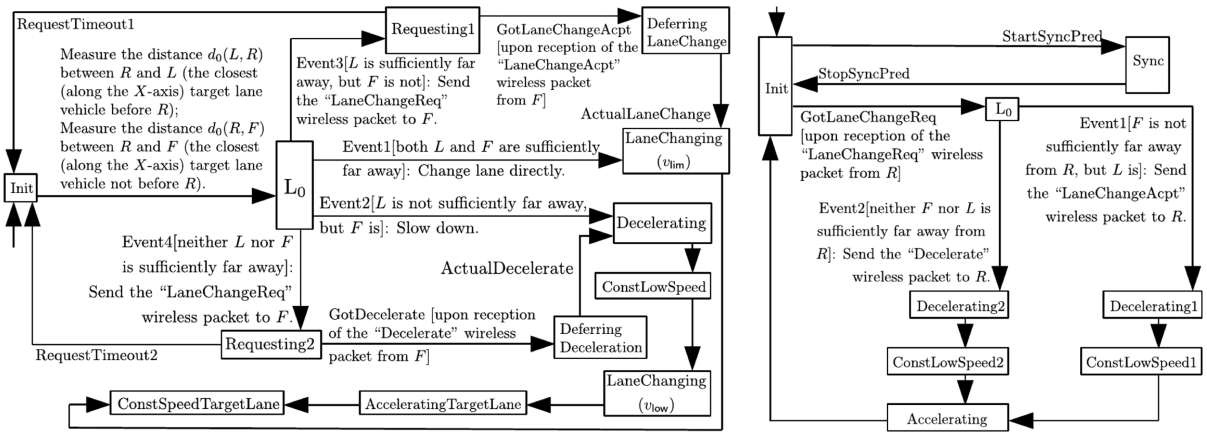
Besides the mechanical driving capabilities, CAVs should also be able to sense their surrounding environments. Particularly, we assume the following.

Assumption 4: Each CAV is equipped with redundant ranging sensors (e.g., laser, radar, ultrasonic, computer vision, and human driver as the last resort), so that starting from any time instance t_0, there will always be a future (or t_0 itself) time instance t_0' ($0 \leqslant t_0' - t_0 < +\infty$), when the requesting CAV R can recognize the leader CAV L (or that $L = \varnothing$, when it does not exist) and the follower CAV F (or that $F = \varnothing$, when it does not exist) on the target lane, and can sense the distance $d_0(L, R)$ and $d_0(R, F)$ to L and F ($d_0(L, R) \stackrel{\text{def}}{=} +\infty$ when $L = \varnothing$, and $d_0(R, F) \stackrel{\text{def}}{=} +\infty$ when $F = \varnothing$). Meanwhile, for any two consecutive CAVs x and y along a *same* lane (suppose x precedes y), as long as there is line-of-sight from y to x, y can always instantly detect x's speed (e.g. based on y's own speed and the relative velocity to x detected by y's ranging sensors).■

For the above lane change scenario, we aim to guarantee the *Constant Time Headway* (CTH) safety rule [50] as specified in the following.

Definition 1 (CTH Safety Rule): Suppose two vehicles (in math point abstraction, see Fig. 1) x and y are driving in the same direction along a same lane (in math axis abstraction, see Fig. 1). Suppose x precedes y at time instance t. Denote the distance between x and y at t as $d(t)$, and y's speed at t as $v_y(t)$. Then we call $\delta(t) \stackrel{\text{def}}{=} d(t)/v_y(t)$ as the *time headway* of y (relative to x) at t. If $\delta(t) \geqslant \Delta^*$, where $\Delta^* > 0$ is a given constant, aka the *desired time headway*, then we say the ordered tuple (x, y) is CTH-Δ^* safe at t. In other words, if $d(t) \geqslant v_y(t)\Delta^*$, then we say (x, y) is CTH-Δ^* safe at t. ■

The importance of the CTH-Δ^* safety rule can be explained by the following fact. Suppose a lane has both a minimum speed limit v^{min} and a maximum speed limit v^{max}, we can set Δ^* to $\Delta^{**} + \frac{D_0}{v^{\mathsf{min}}}$, where Δ^{**} is the maximum duration needed to stop a CAV at any speed $v \in [v^{\mathsf{min}}, v^{\mathsf{max}}]$ using emergency braking



(a) CAV R (b) CAVs V_i

Fig. 2. Automata Sketches. Rectangles are modes, arrows between modes are events, and the arrow without source mode indicates the initial mode in the respective automata sketches. Texts in "[ ]" are the triggering conditions (aka *guards*) for the corresponding events; texts after the ":" are the *actions* to be carried out once the corresponding events happen.

(which should be monotonically decelerating, and can be different from the normal deceleration dec), and D_0 is the maximum vehicle body length. Under this setting, simple analysis can show that CTH-Δ^* safety guarantees y will never hit x, even if x suddenly stops at anytime on the lane.

We hence assume the following, which implies the safety of the target lane before the lane change.

Assumption 5: $\forall i \in \{1, 2, \ldots, n-1\}$, (V_i, V_{i+1}) is CTH-Δ^* safe at t_0. ■

This naturally leads to our research problem: whether the lane change requested by R (see Fig. 1) will disrupt the safety of the target lane? Furthermore, as cooperation between CAVs need communications of wireless packets, and wireless communications are well-known unreliable, can we guarantee the CTH safety when wireless packets are arbitrarily lost? More rigorously,

Research Problem 1 (CTH Safety Guarantee): How to coordinate the CAV lane change under arbitrary wireless packet losses, so that CTH-Δ^* safety always holds in the target lane? ■

Related, we are also concerned about the liveness and performance of our proposed CLC protocol. Formally,

Research Problem 2 (Liveness Guarantee): Under arbitrary wireless packet losses, can the CLC protocol reset itself within bounded time? ■

Research Problem 3 (Performance): Under arbitrary wireless packet losses, what are the success rate and time cost of lane change? ■

IV. Solution

In this section, we propose our protocol to answer Research Problem 1 to 3.

A. Heuristics

The heuristics of our proposed protocol is illustrated by the automata sketches of Fig. 2.

Initially (i.e. at t_0), the target lane CAVs V_i $(i = 1, 2, \ldots, n)$ and the requesting CAV R all reside in their respective "Init" mode. At t_0' $(0 \leqslant t_0' - t_0 < +\infty)$, R intends to change lane and triggers the first action: it recognizes (with its ranging sensors) the closest (along the X-axis) target lane CAV before it (i.e. the *leader CAV*, denoted as L) and the closest (along the X-axis) target lane CAV not before (i.e. after or at the same X coordinate as) it (i.e. the *follower CAV*, denoted as F). If the leader (follower) CAV does not exist, we denote $L \stackrel{\text{def}}{=} \varnothing$ ($F \stackrel{\text{def}}{=} \varnothing$). Using the ranging sensors, R can measure its current distances to L and F along the X-axis: denote them respectively as $d_0(L, R)$ and $d_0(R, F)$ (see Fig. 1). Note $d_0(L, R) \stackrel{\text{def}}{=} +\infty$ if $L = \varnothing$, and $d_0(R, F) \stackrel{\text{def}}{=} +\infty$ if $F = \varnothing$. R then enters the intermediate mode of "L_0" to take further actions based on $d_0(L, R)$ and $d_0(R, F)$. Specifically,

1) If R is sufficiently far away from both L and F (see "Event1" in Fig. 2(a)), then it will directly start the mechanical lane change routine with the initial speed of v_{lim}.
2) If R is close to L but sufficiently far away from F (see "Event2" in Fig. 2(a)), then it will decelerate to v_{low} and maintain this speed to reach a safe distance from L. Then it will start the mechanical lane change routine with the initial speed of v_{low}. By the end of the routine, R will reach the target lane, and will then accelerate back to v_{lim}.
3) If R is sufficiently far away from L but close to F (see "Event3" in Fig. 2(a)), then it will first send a "LaneChangeReq" wireless packet to F. Upon receiving this packet, F will reply a "LaneChangeAcpt" wireless packet to R, and decelerate to v_{low} to reach a safe distance from R (see "Event1" in Fig. 2(b)). At R, upon reception of the "LaneChangeAcpt" wireless packet, R will wait for F to reach the safe distance, and then start the mechanical lane change routine with the initial speed of v_{lim}. By the end of the routine, R will reach the target lane, and F will accelerate back to v_{lim}. In addition, the target lane CAVs V_i after F will maintain the convoy by synchronizing their speeds with their respective predecessors if needed.
4) Otherwise (R is close to both L and F, see "Event4" in Fig. 2(a)), R will first send a "LaneChangeReq" wireless

packet to F. Upon reception of this packet, F will reply a "Decelerate" wireless packet to R, and decelerate to v_{low} to reach a safe distance from R (see "Event2" in Fig. 2(b)). At R, upon reception of the "Decelerate" wireless packet, R will wait for F to reach the safe distance, and then also decelerate to v_{low} to reach a safe distance from L. Then R will start the mechanical lane change routine with the initial speed of v_{low}. By the end of the routine, R will reach the target lane, and will then accelerate back to v_{lim}. Meanwhile, F will also accelerate back to v_{lim}. In addition, the target lane CAVs V_i after F will maintain the convoy by synchronizing their speeds with their respective predecessors if needed.

For the above cases, how "far" is "sufficiently far" and how to configure the parameters to achieve the CTH-Δ^* safety are non-trivial questions. We will clarify them in the detailed protocol design and analysis (see Sections IV-B and IV-C).

Another challenge is the possibility of arbitrary wireless packet losses. What if the "LaneChangeReq," "Decelerate," "LaneChangeAcpt," wireless packets are lost? Can the CTH-Δ^* safety still sustain? Can the CAVs still reset themselves instead of stuck in a mode forever?

To address these concerns, we propose to deploy the "*lease*" design pattern for distributed systems [22]. A "*lease*" is an agreement on timeout, contracted since the early stage of a cooperation. After the lease is contracted, if wireless packets are lost, the affected entities can reset themselves when the agreed timeout is reached (by looking at their respective local clocks, hence need no more communications). In Fig. 2, nearly every mode has its timeout configuration. The exact configurations to choose are also non-trivial problems that affect the CTH-Δ^* safety, system liveness, and efficiency. The details and analysis are also elaborated in Sections IV-B and IV-C. The efficiency is evaluated in Section VI.

B. Proposed Protocol

We propose our detailed protocol by expanding the automata sketches of Fig. 2 with the heuristics described in Section IV-A.

1) Protocol Symbols: The protocol specifications (and follow up analyses) may use the following symbols.

First, as our proposed protocol deals with two and only two given steady state speeds: v_{lim} and v_{low}, where

$$v_{\mathsf{lim}} > v_{\mathsf{low}} > 0, \tag{2}$$

all straight line acceleration and decelerations are between these two speeds. Hence we adopt the following simplification notations:

$$\delta_{\mathsf{a}}^{\ddagger} \stackrel{\mathsf{def}}{=} \delta_{\mathsf{a}}(v_{\mathsf{low}}, v_{\mathsf{lim}}), \quad d_{\mathsf{a}}^{\ddagger} \stackrel{\mathsf{def}}{=} d_{\mathsf{a}}(v_{\mathsf{low}}, v_{\mathsf{lim}}),$$
$$\delta_{\mathsf{d}}^{\ddagger} \stackrel{\mathsf{def}}{=} \delta_{\mathsf{d}}(v_{\mathsf{lim}}, v_{\mathsf{low}}), \quad d_{\mathsf{d}}^{\ddagger} \stackrel{\mathsf{def}}{=} d_{\mathsf{d}}(v_{\mathsf{lim}}, v_{\mathsf{low}}). \tag{3}$$

Other simplification notations include

$$\widetilde{v} \stackrel{\mathsf{def}}{=} v_{\mathsf{lim}} - v_{\mathsf{low}} > 0 \text{ (due to Ineq. (2))}, \tag{4}$$

$$\widetilde{d}_{\mathsf{a}}(v_{\mathsf{lim}}) \stackrel{\mathsf{def}}{=} v_{\mathsf{lim}}\delta_{\mathsf{a}}^{\ddagger} - d_{\mathsf{a}}^{\ddagger} > 0 \text{ (due to Assumption 2)}, \tag{5}$$

$$\widetilde{d}_{\mathsf{a}}(v_{\mathsf{low}}) \stackrel{\mathsf{def}}{=} d_{\mathsf{a}}^{\ddagger} - v_{\mathsf{low}}\delta_{\mathsf{a}}^{\ddagger} > 0 \text{ (due to Assumption 2)}, \tag{6}$$

$$\widetilde{d}_{\mathsf{d}}(v_{\mathsf{lim}}) \stackrel{\mathsf{def}}{=} v_{\mathsf{lim}}\delta_{\mathsf{d}}^{\ddagger} - d_{\mathsf{d}}^{\ddagger} > 0 \text{ (due to Assumption 3)}, \tag{7}$$

$$\widetilde{d}_{\mathsf{d}}(v_{\mathsf{low}}) \stackrel{\mathsf{def}}{=} d_{\mathsf{d}}^{\ddagger} - v_{\mathsf{low}}\delta_{\mathsf{d}}^{\ddagger} > 0 \text{ (due to Assumption 3)}, \tag{8}$$

$$\widetilde{d_{\mathsf{lc}}}(v_{\mathsf{lim}}) \stackrel{\mathsf{def}}{=} v_{\mathsf{lim}}\delta_{\mathsf{lc}}(v_{\mathsf{lim}}) - d_{\mathsf{lc}}^{\mathsf{X}}(v_{\mathsf{lim}}) > 0 \text{ (due to Ineq. (1))}, \tag{9}$$

$$\widetilde{d_{\mathsf{lc}}}(v_{\mathsf{low}}) \stackrel{\mathsf{def}}{=} v_{\mathsf{low}}\delta_{\mathsf{lc}}(v_{\mathsf{low}}) - d_{\mathsf{lc}}^{\mathsf{X}}(v_{\mathsf{low}}) > 0 \text{ (due to Ineq. (1))}. \tag{10}$$

Our proposed protocol adopts the following configurable constants:

1) $\Delta^* > 0$, the desired time headway of CTH safety, see Definition 1.
2) $\Delta_{\mathsf{nonzeno}} > 0$, maximum waiting time for wireless reply.

These configurable constants further imply the following constants used in the protocol.

$$D_1 \stackrel{\mathsf{def}}{=} v_{\mathsf{lim}}\Delta^* - \widetilde{d_{\mathsf{lc}}}(v_{\mathsf{lim}}), \tag{11}$$

$$D_2 \stackrel{\mathsf{def}}{=} 2v_{\mathsf{lim}}\Delta^* + \widetilde{d}_{\mathsf{d}}(v_{\mathsf{lim}}) + \widetilde{d_{\mathsf{lc}}}(v_{\mathsf{low}}) + \widetilde{v}(\delta_{\mathsf{lc}}(v_{\mathsf{low}}) + \delta_{\mathsf{a}}^{\ddagger}), \tag{12}$$

$$D_3 \stackrel{\mathsf{def}}{=} v_{\mathsf{lim}}\Delta^* + \widetilde{d_{\mathsf{lc}}}(v_{\mathsf{lim}}), \tag{13}$$

$$D_{\mathsf{Sync}} \stackrel{\mathsf{def}}{=} \max\{D_{\mathsf{Sync}}^{\mathsf{Event1,min}}, D_{\mathsf{Sync}}^{\mathsf{Event2,min}}\}, \tag{14}$$

where

$$D_{\mathsf{Sync}}^{\mathsf{Event1,min}} \stackrel{\mathsf{def}}{=} v_{\mathsf{lim}}\Delta^* + D_3 + \widetilde{d}_{\mathsf{d}}(v_{\mathsf{lim}}) + \widetilde{d}_{\mathsf{d}}(v_{\mathsf{low}}) + \widetilde{v}(\delta_{\mathsf{lc}}(v_{\mathsf{lim}}) + \delta_{\mathsf{a}}^{\ddagger}), \tag{15}$$

$$D_{\mathsf{Sync}}^{\mathsf{Event2,min}} \stackrel{\mathsf{def}}{=} v_{\mathsf{lim}}\Delta^* + D_2 + \widetilde{d}_{\mathsf{d}}(v_{\mathsf{lim}}) + \widetilde{v}\delta_{\mathsf{a}}^{\ddagger}, \tag{16}$$

$$\Delta_{\mathsf{Coop}}^{\mathsf{Event1,max}} \stackrel{\mathsf{def}}{=} \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{a}}^{\ddagger} + \frac{D_3 + \widetilde{d}_{\mathsf{d}}(v_{\mathsf{low}})}{\widetilde{v}} + \delta_{\mathsf{lc}}(v_{\mathsf{lim}}), \tag{17}$$

$$\Delta_{\mathsf{Coop}}^{\mathsf{Event2,max}} \stackrel{\mathsf{def}}{=} \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{a}}^{\ddagger} + \frac{D_2}{\widetilde{v}}, \tag{18}$$

$$\Delta_{\mathsf{Coop}}^{\mathsf{max}} \stackrel{\mathsf{def}}{=} \max\{\Delta_{\mathsf{Coop}}^{\mathsf{Event2,max}} + \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{lc}}(v_{\mathsf{low}}) + \Delta^*, \Delta_{\mathsf{Coop}}^{\mathsf{Event1,max}}\}, \tag{19}$$

$$\Delta_{\mathsf{reset}} \stackrel{\mathsf{def}}{=} \Delta_{\mathsf{Coop}}^{\mathsf{max}} + \Delta_{\mathsf{nonzeno}}. \tag{20}$$

Table I summarizes the symbols used in this paper.

2) Requesting CAV R Protocol Behaviors: The requesting CAV R's protocol behaviors are formally defined by a hybrid automaton [52] A_R, which is graphically specified by Fig. 3. For readers unfamiliar with the graphical specification conventions of hybrid automata, the following paragraphs give a brief tutorial; a more comprehensive tutorial can be found in [52].

As exemplified in Fig. 3, each rectangle box represents a *hybrid automaton mode* (simplified as "*mode*" in the following). At any time instance, the hybrid automaton executes according

TABLE I
SYMBOL LIST (LISTED ALPHABETICALLY: GREEK BEFORE LATIN, UPPER CASE BEFORE LOWER CASE)

Symbol	Intuitive Meaning	Symbol	Intuitive Meaning
Δ^*	The desired time headway. See Def. 1.	R	The requesting CAV. See Sec. 3.1 and Fig. 1.
$\Delta_{\mathsf{Coop}}^{\mathsf{Event1,max}}$, $\Delta_{\mathsf{Coop}}^{\mathsf{Event2,max}}$	Max duration of F's stay in the "Coop" state, respectively in case that F enters the "Coop" state via Event1 or Event2. See Eq. (17), (18) and Fig. 4.	V_i	The ith target lane CAV ($i = 1, 2, \ldots, n$). See Sec. 3.1.
$\Delta_{\mathsf{Coop}}^{\mathsf{max}}$	Max duration (a loose upper bound) of F's stay in the "Coop" state. See Eq. (19).	$\mathsf{acc}(v_{\mathsf{a}}^{\mathsf{low}}, v_{\mathsf{a}}^{\mathsf{high}}, \tau)$	Speed at the τth second of the acceleration routine to accelerate from $v_{\mathsf{a}}^{\mathsf{low}}$ to $v_{\mathsf{a}}^{\mathsf{high}}$. See Sec. 3.2.2.
$\Delta_{\mathsf{nonzeno}}$	Max waiting time for wireless reply. See Sec. 4.2.1-2).	$d_0(L, R)$, $d_0(R, F)$	Respectively the last measured distance between L and R, and between R and F. See Sec. 3.1.
Δ_{reset}	Automatic reset time bound. See Eq. (20).	$d_{\mathsf{a}}^{\ddagger}$	Distance to accelerate from v_{low} to v_{lim}. See Eq. (3).
$\delta_{\mathsf{a}}^{\ddagger}$	Duration to accelerate from v_{low} to v_{lim}. See Eq. (3).	$d_{\mathsf{a}}(v_{\mathsf{a}}^{\mathsf{l}}, v_{\mathsf{a}}^{\mathsf{h}})$	Distance to accelerate from $v_{\mathsf{a}}^{\mathsf{l}}$ to $v_{\mathsf{a}}^{\mathsf{h}}$. See Sec. 3.2.2.
$\delta_{\mathsf{a}}(v_{\mathsf{a}}^{\mathsf{l}}, v_{\mathsf{a}}^{\mathsf{h}})$	Duration to accelerate from $v_{\mathsf{a}}^{\mathsf{l}}$ to $v_{\mathsf{a}}^{\mathsf{h}}$. See Sec. 3.2.2.	$\widetilde{d}_{\mathsf{a}}(v_{\mathsf{lim}})$	$\stackrel{\mathsf{def}}{=} v_{\mathsf{lim}}\delta_{\mathsf{a}}^{\ddagger} - d_{\mathsf{a}}^{\ddagger}$. See Eq. (5).
$\delta_{\mathsf{comm}}^{\mathsf{max}}$	Max one-hop wireless packet request-reply delay. See Thm. 1-(**c6**).	$\widetilde{d}_{\mathsf{a}}(v_{\mathsf{low}})$	$\stackrel{\mathsf{def}}{=} d_{\mathsf{a}}^{\ddagger} - v_{\mathsf{low}}\delta_{\mathsf{a}}^{\ddagger}$. See Eq. (6).
$\delta_{\mathsf{d}}^{\ddagger}$	Duration to decelerate from v_{lim} to v_{low}. See Eq. (3).	$d_{\mathsf{d}}^{\ddagger}$	Distance to decelerate from v_{lim} to v_{low}. See Eq. (3).
$\delta_{\mathsf{d}}(v_{\mathsf{d}}^{\mathsf{h}}, v_{\mathsf{d}}^{\mathsf{l}})$	Duration to decelerate from $v_{\mathsf{d}}^{\mathsf{h}}$ to $v_{\mathsf{d}}^{\mathsf{l}}$. See Sec. 3.2.3.	$d_{\mathsf{d}}(v_{\mathsf{d}}^{\mathsf{h}}, v_{\mathsf{d}}^{\mathsf{l}})$	Distance to decelerate from $v_{\mathsf{d}}^{\mathsf{h}}$ to $v_{\mathsf{d}}^{\mathsf{l}}$. See Sec. 3.2.3.
$\delta_{\mathsf{defer}}^{\mathsf{Rdec}}$	Duration that R should defer its deceleration. See Fig. 3 mode "DeferringDeceleration."	$\widetilde{d}_{\mathsf{d}}(v_{\mathsf{lim}})$	$\stackrel{\mathsf{def}}{=} v_{\mathsf{lim}}\delta_{\mathsf{d}}^{\ddagger} - d_{\mathsf{d}}^{\ddagger}$. See Eq. (7).
$\delta_{\mathsf{defer}}^{\mathsf{lc}}$	Duration that R should defer its lane change. See Fig. 3 mode "DeferringLaneChange."	$\widetilde{d}_{\mathsf{d}}(v_{\mathsf{low}})$	$\stackrel{\mathsf{def}}{=} d_{\mathsf{d}}^{\ddagger} - v_{\mathsf{low}}\delta_{\mathsf{d}}^{\ddagger}$. See Eq. (8).
$\delta_{\mathsf{lc}}(v_{\mathsf{lc}}^0)$	Duration to complete a mechanical lane change routine with initial velocity of v_{lc}^0. See Sec. 3.2.1.	$\mathsf{dec}(v_{\mathsf{d}}^{\mathsf{high}}, v_{\mathsf{d}}^{\mathsf{low}}, \tau)$	Speed at the τth second of the the deceleration routine to decelerate from $v_{\mathsf{d}}^{\mathsf{high}}$ to $v_{\mathsf{d}}^{\mathsf{low}}$. See Sec. 3.2.3.
$\delta_{\mathsf{low}}^{\mathsf{LR}}$	Duration that R should remain at v_{low} to increase its distance to L. See Fig. 3 mode "ConstLowSpeed."	$\widetilde{d_{\mathsf{lc}}}(v_{\mathsf{lim}})$	$\stackrel{\mathsf{def}}{=} v_{\mathsf{lim}}\delta_{\mathsf{lc}}(v_{\mathsf{lim}}) - d_{\mathsf{lc}}^{\mathsf{X}}(v_{\mathsf{lim}})$. See Eq. (9).
$\delta_{\mathsf{low}}^{\mathsf{RF1}}$, $\delta_{\mathsf{low}}^{\mathsf{RF2}}$	Duration that F should remain at v_{low} to increase its distance to R. Respectively see Fig. 4 mode "ConstLowSpeed1" and "ConstLowSpeed2."	$\widetilde{d_{\mathsf{lc}}}(v_{\mathsf{low}})$	$\stackrel{\mathsf{def}}{=} v_{\mathsf{low}}\delta_{\mathsf{lc}}(v_{\mathsf{low}}) - d_{\mathsf{lc}}^{\mathsf{X}}(v_{\mathsf{low}})$. See Eq. (10).
τ	A local clock; or a temporary variable. Particularly, in Sec. 4.2.2 and Fig. 3, it is a local clock used by A_R; in Sec. 4.2.3 and Fig. 4, it is a local clock used by A_i.	$d_{\mathsf{lc}}^{\mathsf{X}}(v_{\mathsf{lc}}^0)$	Projected X-axis distance experienced by a completed mechanical lane change routine (with initial velocity v_{lc}^0). See Sec. 3.2.1.
A_R	Hybrid automaton for the requesting CAV R. See Sec. 4.2.2 and Fig. 3.	$\mathsf{lc}(v_{\mathsf{lc}}^0, \tau)$	Velocity at the τth second of a mechanical lane change routine (with initial velocity of v_{lc}^0). See Sec. 3.2.1.
A_i	Hybrid automaton for the target lane CAV V_i. See Sec. 4.2.3 and Fig. 4.	n	Total number of target lane CAVs. See Sec. 3.1.
D_1	Threshold deciding if L and R are far enough. See Eq. (11).	$\vec{p}(x, t)$	Location of CAV x at time instance t. See Sec. 3.1.
D_2	(When L and R are not far enough) threshold deciding if R and F are far enough. See Eq. (12).	state	State of V_i, can and only can be "Init," "Coop," or "Sync." See Fig. 4.
D_3	(When L and R are far enough) threshold deciding if R and F are far enough. See Eq. (13).	t	A time instance of the world.
D_{Sync}	Threshold deciding if two consecutive target lane vehicles are far enough. See Eq. (14) and Fig. 4 event "StartSyncPred."	t_0	Scenario start time instance. See Sec. 3.1.
$D_{\mathsf{Sync}}^{\mathsf{Event1,min}}$, $D_{\mathsf{Sync}}^{\mathsf{Event2,min}}$	Intermediate variables to calculate D_{Sync}. See Eq. (15), (16).	t_1	Temporary variable. See Thm. 1-**Claim 2**.
F	The follower CAV: the closest (along the X-axis) target lane CAV not before R, or $\varnothing$ if nonexistent. See Fig. 1.	t_2	Temporary variable. See Thm. 1-**Claim 2**.
L	The leader CAV: the closest (along the X-axis) target lane CAV before R, or $\varnothing$ if nonexistent. See Fig. 1.	$\widetilde{v}$	$\stackrel{\mathsf{def}}{=} v_{\mathsf{lim}} - v_{\mathsf{low}}$. See Eq. (4).
P	Wireless packet loss rate. See Sec. 6.1.	v_{lim}, v_{low}	Respectively the max and min steady state speed along a highway lane. See Sec. 3.1 and Ineq. (2).

to one of these modes, aka the hybrid automaton "*dwells*" (aka "*resides*") in one of these modes. Inside a mode rectangle, the top line is the mode's name (which is local to the hybrid automaton, i.e. modes of the same name of different hybrid automata are different modes), the rest in the rectangle specifies the continuous domain dynamics (by default via differential equations with respect to time, e.g. $\dot{\tau} = 1$) and/or constraints (e.g. a dwelling duration constraint like $0 \leqslant \tau < \Delta_{\mathsf{nonzeno}}$) of the mode.

For example, in Fig. 3, the rectangle near the upper left corner represents a mode, whose name is "Init." That is, this rectangle represents the "Init" mode of hybrid automaton A_R. Inside the rectangle, we have the following continuous domain dynamics specified with differential equations:

$$|\dot{\vec{p}}(R, t)| = v_{\mathsf{lim}}, \tag{21}$$

$$\dot{\tau} = 1, \tag{22}$$

where (21) specifies that when executing in the "Init" mode, the time derivative of $\vec{p}(R, t)$ maintains the magnitude of v_{lim}, i.e. vehicle R maintains the speed of v_{lim}; (22) specifies that when executing in the "Init" mode, the continuous variable τ increments at a rate of 1 per second (i.e. τ represents a clock).

An arrow between two modes represents a discrete *event* (the two modes are respectively the *source mode* and *destination mode* of the event). The event arrow is usually annotated. The

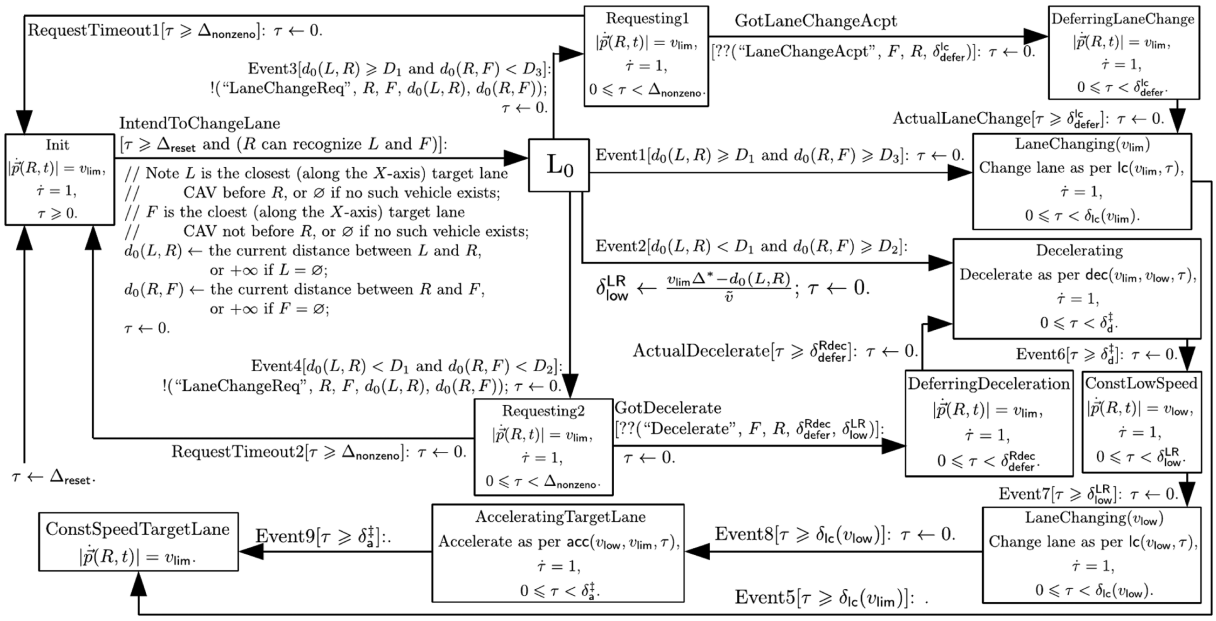


Fig. 3. Hybrid automaton A_R (see Section IV-B-2) for the requesting CAV R. For readers unfamiliar with the hybrid automata graphical specification conventions [52], a tutorial is given in Section IV-B-2. Note mode "L_0" is a dummy mode, whose allowed dwelling duration is 0 s. That is, once entered "L_0," the execution must leave "L_0" immediately via one of the mode's outgoing events.

annotation is usually a string containing a ":" symbol. Before the ":" is the optional *event name* and the event's *guard* (quoted by the brackets "[ ]"). The event guard specifies the triggering condition for the event. Particularly, "??(x)" means 1) the event is triggered upon the *reception* of a wireless packet x (where x is a tuple of three or four elements, respectively the type, sender, intended receiver, and optional data payload of the packet); and 2) the sent wireless packet x is not always received: *it can be lost arbitrarily*. After the ":" are the instant *actions* to be taken when the event happens. Particularly, "!(y)" means a wireless packet y is sent once the event happens.

For example, in Fig. 3, the arrow from mode "L_0" to mode "Requesting1" represents an event named "Event3." This event is triggered when $d_0(L,R) \geqslant D_1$ and $d_0(R,F) < D_3$ (see the event guard of "[$d_0(L,R) \geq D_1$ and $d_0(R,F) < D_3$]"). Once triggered, this event runs the following sequence of actions instantly.

First it sends out a wireless packet of type "LaneChangeReq," whose sender is R, receiver is F, and data payload is a tuple $(d_0(L,R), d_0(R,F))$ (see the action notation "!('LaneChangeReq', R, F, $d_0(L,R)$, $d_0(R,F)$)").

Next it assigns τ with value of 0 (see the action notation "$\tau \leftarrow 0$").

Note the convention uses "$\leftarrow$" for value assignment, and uses "$=$" instead for value equivalence comparison, so as to avoid ambiguity.

As another example, the arrow from mode "Requesting1" to mode "DeferringLaneChange" represents an event named "GotLaneChangeAcpt." This event is triggered upon the *reception* of a wireless packet, whose type is "LaneChangeAcpt," sender is F, receiver is R, and data payload is $\delta^{\text{lc}}_{\text{defer}}$ (see the guard notation "[??('LaneChangeAcpt', F, R, $\delta^{\text{lc}}_{\text{defer}}$)]"). Note "??" means a sent "LaneChangeAcpt" wireless packet is not always received; the packet can be lost arbitrarily; and the event "GotLaneChangeAcpt" is only triggered when the packet is received (i.e. not lost).

Besides the event arrows, each hybrid automaton has one arrow that has only the destination mode (and without the source mode). Such an arrow indicates the starting mode of execution. For example, in Fig. 3, the execution starts from mode "Init" due to such an arrow. This arrow can also specify initialization actions to be carried out before executing the starting mode. For example, in Fig. 3, the initialization action is assigning τ with value Δ_{reset} (see the action notation "$\tau \leftarrow \Delta_{\text{reset}}$" at the source end of the arrow).

3) Target Lane CAV V_i Protocol Behaviors: A target lane CAV V_i's ($i \in \{1, 2, \ldots, n\}$) protocol behaviors are formally defined by a hybrid automaton A_i, which is graphically specified by Fig. 4. Readers unfamiliar with the graphical specification conventions of hybrid automata can refer to the tutorial given in Section IV-B-2, or a more detailed tutorial in [52].

Also note that according to the convention, τ represents a clock local to the corresponding hybrid automaton (i.e. the τ in Fig. 4 is the local clock of hybrid automaton A_i, while τ in Fig. 3 is the local clock of hybrid automaton A_R).

C. Analysis

We now analyze the proposed protocol. Specifically, we claim the following theorem, which answers Research Problem 1 and 2.

Theorem 1: Suppose the following constraints hold:

(**c1**) $v_{\text{lim}} > v_{\text{low}} > 0$, i.e. Ineq. (2);
(**c2**) $\Delta^* > 0$ and $\Delta_{\text{nonzeno}} > 0$;

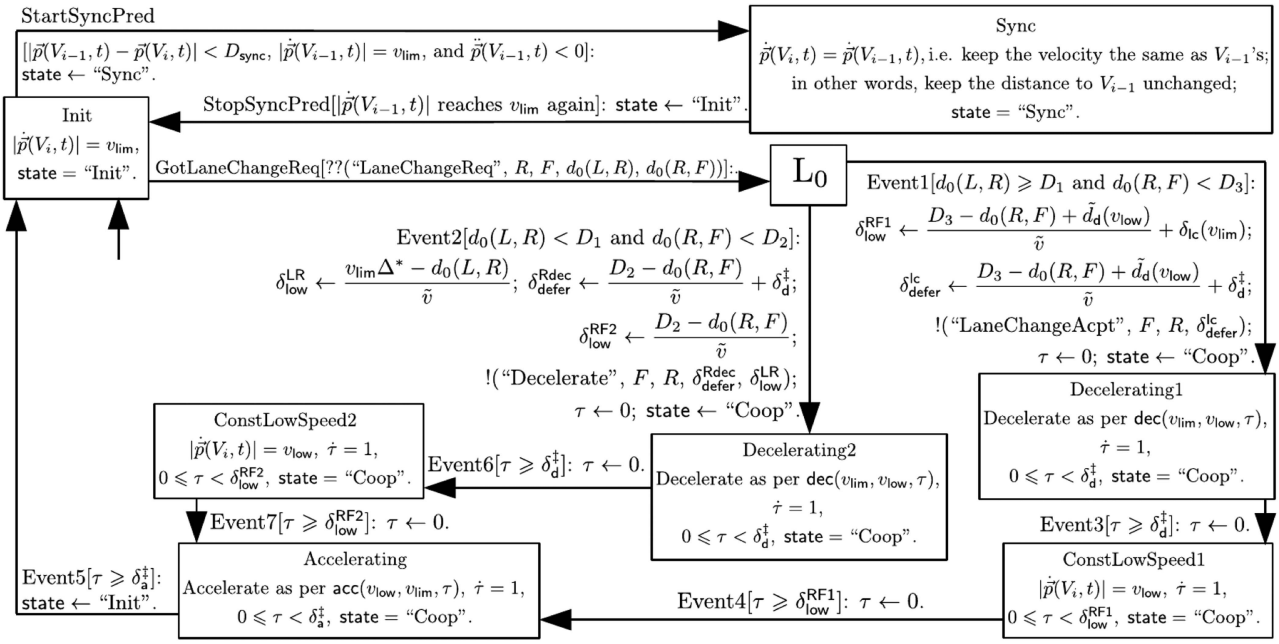


Fig. 4. Hybrid automaton A_i (see Section IV-B-3) for the target lane CAV V_i ($i=1,2,\ldots,n$). For readers unfamiliar with the hybrid automata graphical specification conventions [52], a tutorial is given in Section IV-B-2. Note: 1) As V_0 does not exist, for V_1, the event "StartSyncPred" can never happen. 2) Mode "L_0" is a dummy mode, whose allowed dwelling duration is 0 s. That is, once entered "L_0," the execution must leave "L_0" immediately via one of the mode's outgoing events.

(c3) $\Delta^*>\delta_{\mathsf{lc}}(v_{\mathsf{low}})>\delta_{\mathsf{lc}}(v_{\mathsf{lim}})>\delta^{\ddagger}_{\mathsf{d}}$;
(c4) $\delta^{\ddagger}_{\mathsf{d}}+\delta_{\mathsf{lc}}(v_{\mathsf{low}})\geqslant\delta^{\ddagger}_{\mathsf{a}}$;
(c5) $\Delta_{\mathsf{nonzeno}}<\min\{\Delta^{\mathsf{Event1,max}}_{\mathsf{Coop}},\Delta^{\mathsf{Event2,max}}_{\mathsf{Coop}}\}$;
(c6) when no packet is lost, maximum one-hop wireless packet request-reply delay $\delta^{\mathsf{max}}_{\mathsf{comm}}<<\min\{\Delta_{\mathsf{nonzeno}},\Delta^*,\delta^{\ddagger}_{\mathsf{a}},\delta^{\ddagger}_{\mathsf{d}},\delta_{\mathsf{lc}}(v_{\mathsf{lim}}),\delta_{\mathsf{lc}}(v_{\mathsf{low}})\}$, hence is negligible.

Then we have the following claims.

Claim 1 (Safety): $\forall t\in[t_0,+\infty)$, for any two CAVs x and y on the target lane, one and only one of the following sustains:
(i) (x,y) is CTH-Δ^* safe at t, or
(ii) (y,x) is CTH-Δ^* safe at t.

Claim 2 (Liveness (Automatic Resetting)): Suppose at $t_1\in[t_0,+\infty)$, R leaves hybrid automaton A_R mode "Init," while V_is ($i=1,2,\ldots,n$) are all residing in the respective hybrid automaton A_i mode "Init." Then $\exists t_2\in(t_1,t_1+\Delta_{\mathsf{reset}}]$ s.t.
(**Stable State 1**) at t_2, V_1, V_2, …, V_n, and R are in respective hybrid automata mode "Init", or
(**Stable State 2**) at t_2, $V_1,V_2,\ldots,V_n$ are in respective hybrid automata mode "Init" and R is in A_R's mode "ConstSpeedTargetLane". ■

In order to prove Theorem 1, we need to first propose/prove several definitions and claims.

Definition 2 (Coop-Duration): For a target lane CAV V_i ($i\in\{1,2,\ldots,n\}$), suppose its hybrid automaton (i.e. A_i) variable, state, changes from "Init" to "Coop" at $t_3\in[t_0,+\infty)$, then as per Fig. 4, the state must change back to "Init" at some finite t_4 (where $t_3<t_4\leqslant t_3+\Delta^{\mathsf{max}}_{\mathsf{Coop}}$, see (19), (17), (18) for the definition of $\Delta^{\mathsf{max}}_{\mathsf{Coop}}$). That is, $\forall t\in(t_3,t_4]$, state ="Coop";[2] and at t_4^+, state = "Init". We call $(t_3,t_4]$ a "*coop-duration*". Note as per Fig. 4, it is easy to see that $\Delta^{\mathsf{max}}_{\mathsf{Coop}}$ is a loose upper bound to the time length of a coop-duration. ■

Lemma 1: Any two coop-durations $(t_5,t_6]$ and $(t_7,t_8]$ respectively belonging to two different target lane CAVs can never overlap nor connect. Formally, i.e., $[t_5,t_6]\cap[t_7,t_8]=\varnothing$. ■

Proof: Suppose $[t_5,t_6]\cap[t_7,t_8]\neq\varnothing$ and suppose $t_9\in[t_5,t_6]\cap[t_7,t_8]$. Then $t_5\in[t_9-\Delta^{\mathsf{max}}_{\mathsf{Coop}},t_9]$ and $t_7\in[t_9-\Delta^{\mathsf{max}}_{\mathsf{Coop}},t_9]$, therefore $|t_5-t_7|\leqslant\Delta^{\mathsf{max}}_{\mathsf{Coop}}$. This means R sends two different "LaneChangeReq" packets within $\Delta^{\mathsf{max}}_{\mathsf{Coop}}$. This contradicts $\Delta_{\mathsf{reset}}>\Delta^{\mathsf{max}}_{\mathsf{Coop}}$ (see Fig. 3 "Init" mode and (20)). □

Lemma 2: Any two coop-durations $(t_{10},t_{11}]$ and $(t_{12},t_{13}]$ can never overlap nor connect. Formally, i.e., $[t_{10},t_{11}]\cap[t_{12},t_{13}]=\varnothing$. ■

Proof: In addition to Lemma 1, applying similar reasoning, we can prove coop-durations of a same target lane CAV V_i cannot overlap nor connect. □

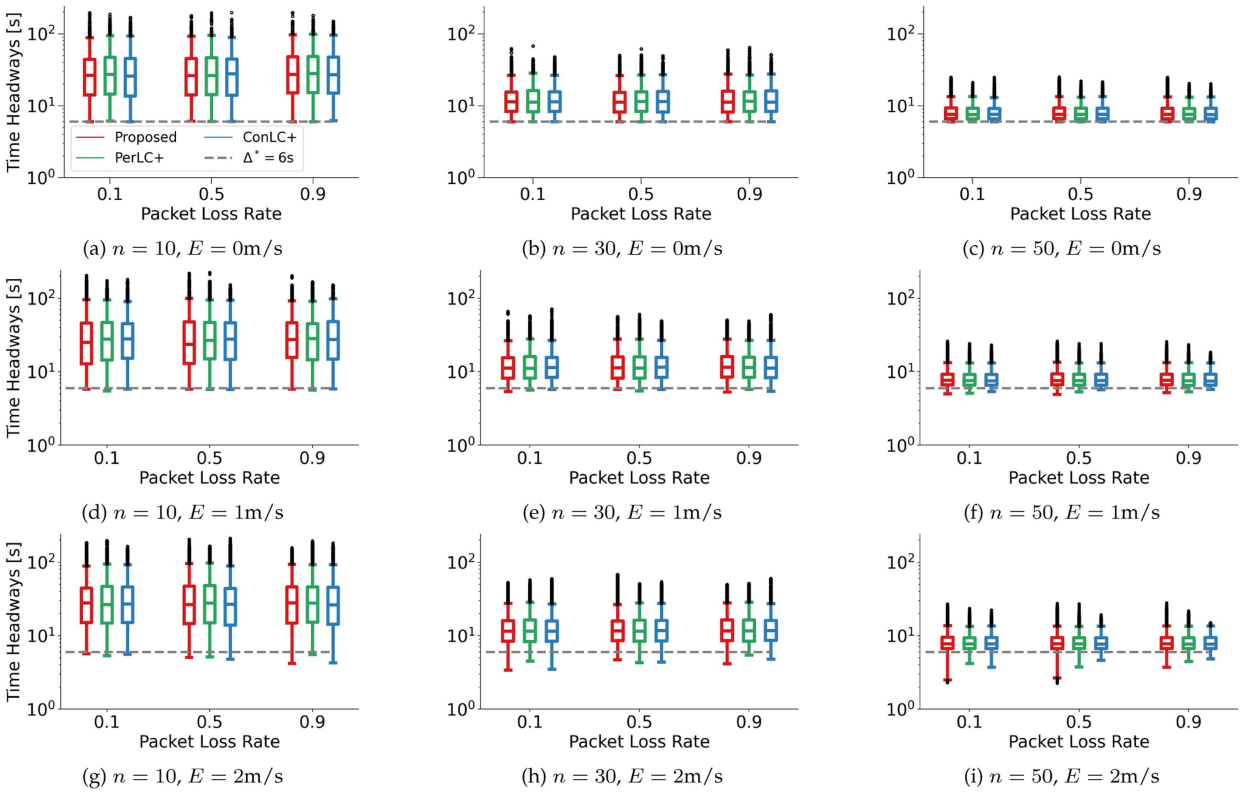
Lemma 3: $\forall t\in[t_0,+\infty)$, if no target lane CAV V_i is in coop-duration at t, then all target lane CAVs (i.e. V_1, V_2, …, V_n) are in "Init" mode at t. ■

Proof: According to Fig. 4, if $\exists V_i$, whose state = "Sync" at t, then there must be an V_j in a coop-duration at t. □

Lemma 4: Suppose $(t_{14},t_{15}]\subseteq[t_0,+\infty)$ is the first ever happened coop-duration, then $\forall t\in[t_0,t_{15}]$, $\forall i\in\{1,2,\ldots,n-1\}$, (V_i,V_{i+1}) is CTH-Δ^* safe at t. ■

Proof: See Appendix A for details. □

[2]Note, if we regard hybrid automaton discrete variables' values are left continuous along time axis, then at t_3, we regard state ="Init".



(a) $n = 10$, $E = 0\text{m/s}$ (b) $n = 30$, $E = 0\text{m/s}$ (c) $n = 50$, $E = 0\text{m/s}$

(d) $n = 10$, $E = 1\text{m/s}$ (e) $n = 30$, $E = 1\text{m/s}$ (f) $n = 50$, $E = 1\text{m/s}$

(g) $n = 10$, $E = 2\text{m/s}$ (h) $n = 30$, $E = 2\text{m/s}$ (i) $n = 50$, $E = 2\text{m/s}$

Fig. 5. Statistics of time headways. X-axes: wireless packet loss rate P (0.1, 0.5, and 0.9, respectively, mean $P = 10\%$, 50%, and 90%). Y-axes: time headway (unit: second). n is the number of target lane CAVs. E is the accumulated velocity disturbance's magnitude bound.

Lemma 5: $\forall t \in [t_0, +\infty), \forall i \in \{1, 2, \ldots, n-1\}, (V_i, V_{i+1})$ is CTH-Δ^* safe at t. ■

Proof: See Appendix B for details. □

Corollary 1: Throughout $[t_0, +\infty)$, there is no spatial swapping between V_i and V_j ($\forall i, j \in \{1, 2, \ldots, n\}, i \neq j$) along the target lane. ■

Proof: Due to Lemma 5, the first swapping never happens. □

Lemma 6: Suppose CAV R finishes a mechanical lane change routine at $t_{16} \in [t_0, +\infty)$, then $\forall i \in \{1, 2, \ldots, n\}$, one and only one of the following claims sustain.

Claim 1 (V_i, R) is CTH-Δ^* safe throughout $[t_{16}, +\infty)$.

Claim 2 (R, V_i) is CTH-Δ^* safe throughout $[t_{16}, +\infty)$. ■

Proof: See Appendix C for details. □

Now we are ready to prove Theorem 1.

Proof of Theorem 1**Claim 1**:

In case $x, y \in \{V_1, V_2, \ldots, V_n\}$, the claim sustains due to Lemma 5 (in case x and y are not consecutive, e.g. $x = V_i$ and $y = V_{i+k}$, where $k > 1$, then the distance between x and y is no less than the distance between V_{i+k-1} and y, hence the CTH-Δ^* safety rule still sustains for (x, y)).

In case $x \in \{V_1, V_2, \ldots, V_n\}$ and $y = R$, or the reverse, the claim sustains due to Lemma 6.

Combining the above two cases, the claim sustains. □ $(*)$

Proof of Theorem 1**Claim 2**:

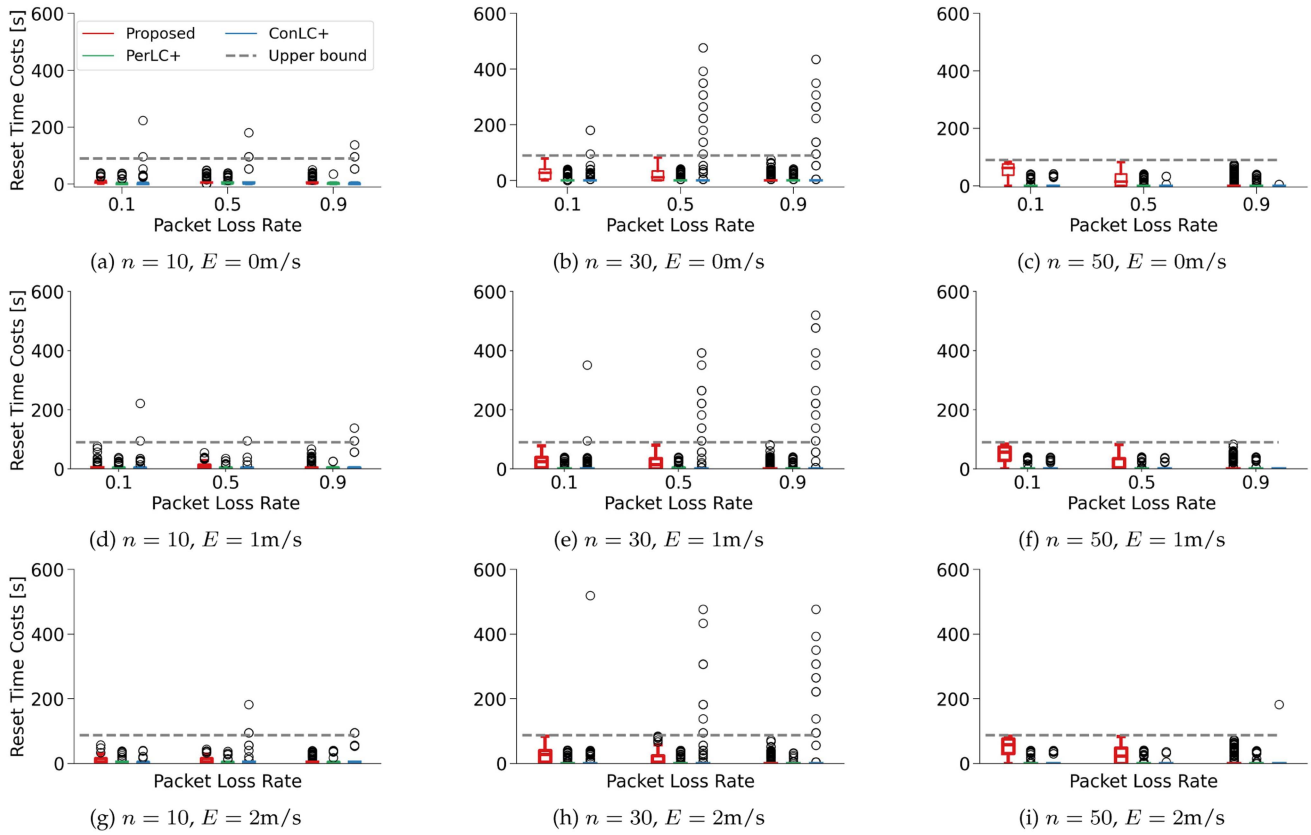
Based on if the requesting CAV R sends the "LaneChangeReq" wireless packet at t_1, there are two cases.

Case 1: R does not send "LaneChangeReq" to the follower CAV $F = V_i$ ($i \in \{1, 2, \ldots, n\}$) at t_1. This includes two further cases.

Case 1.1: R triggers "Event1" of A_R (see Fig. 3) at t_1. Then R will enter the "ConstSpeedTargetLane" mode at $t_{17} \overset{\text{def}}{=} t_1 + \delta_{\text{lc}}(v_{\text{lim}}) < t_1 + \Delta_{\text{reset}}$. Meanwhile, all target lane CAVs $V_1 \sim V_n$ remain in their respective "Init" mode throughout $[t_1, +\infty)$. Therefore, t_{17} is a time instance that satisfies the claim's description (we call such a time instance a "*valid time instance*" in the following).

Case 1.2: R triggers "Event2" of A_R (see Fig. 3) at t_1. Then R will enter the "ConstSpeedTargetLane" mode at

$$
\begin{aligned}
t_{18} &\overset{\text{def}}{=} t_1 + \delta_{\text{d}}^{\ddagger} + \delta_{\text{low}}^{\text{LR}} + \delta_{\text{lc}}(v_{\text{low}}) + \delta_{\text{a}}^{\ddagger} \\
&= t_1 + \delta_{\text{d}}^{\ddagger} + \frac{v_{\text{lim}}\Delta^* - d_0(L, R)}{\widetilde{v}} + \delta_{\text{lc}}(v_{\text{low}}) + \delta_{\text{a}}^{\ddagger} \\
&\leqslant t_1 + \delta_{\text{d}}^{\ddagger} + \frac{v_{\text{lim}}\Delta^*}{\widetilde{v}} + \delta_{\text{lc}}(v_{\text{low}}) + \delta_{\text{a}}^{\ddagger} \\
&= t_1 + \delta_{\text{d}}^{\ddagger} + \delta_{\text{a}}^{\ddagger} + \frac{v_{\text{lim}}\Delta^* + \widetilde{v}\delta_{\text{lc}}(v_{\text{low}})}{\widetilde{v}} \\
&\leqslant t_1 + \delta_{\text{d}}^{\ddagger} + \delta_{\text{a}}^{\ddagger} + \frac{D_2}{\widetilde{v}} \\
&\leqslant t_1 + \Delta_{\text{Coop}}^{\text{Event2,max}} \leqslant t_1 + \Delta_{\text{Coop}}^{\text{max}} < t_1 + \Delta_{\text{reset}}
\end{aligned}
$$



(a) $n = 10$, $E = 0\text{m/s}$ (b) $n = 30$, $E = 0\text{m/s}$ (c) $n = 50$, $E = 0\text{m/s}$

(d) $n = 10$, $E = 1\text{m/s}$ (e) $n = 30$, $E = 1\text{m/s}$ (f) $n = 50$, $E = 1\text{m/s}$

(g) $n = 10$, $E = 2\text{m/s}$ (h) $n = 30$, $E = 2\text{m/s}$ (i) $n = 50$, $E = 2\text{m/s}$

Fig. 6. Statistics of reset time costs. X-axes: wireless packet loss rate P (0.1, 0.5, and 0.9, respectively, mean $P = 10\%$, 50%, and 90%). Y-axes: reset time costs (unit: second). n is the number of target lane CAVs. E is the accumulated velocity disturbance's magnitude bound.

(see (18), (19)).

Meanwhile, all target lane CAVs $V_1 \sim V_n$ remain in their respective "Init" mode throughout $[t_1, +\infty)$. Therefore, t_{18} is a valid time instance.

Combining **Case 1.1** and **Case 1.2**, **Case 1** complies with the claim.

Case 2: R sends "LaneChangeReq" to the follower CAV $F = V_i$ ($i \in \{1, 2, \ldots, n\}$) at t_1.

This also implies that R will not send another "LaneChangeReq" wireless packet during $(t_1, t_1 + \Delta_{\mathsf{reset}}]$ (see Fig. 3) (†)

Meanwhile, at the receiver's end, we can have two cases.

Case 2.1: $F = V_i$ receives the "LaneChangeReq" wireless packet at t_1 (see event "GotLaneChangeReq" in Fig. 4), which leads to two possible cases.

Case 2.1.1: The reception of the "LaneChangeReq" packet triggers "Event1" of A_i (see Fig. 4) at t_1. Then a coop-duration starts at t_1 and ends at $t_{19} \overset{\text{def}}{=} t_1 + \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{low}}^{\mathsf{RF1}} + \delta_{\mathsf{a}}^{\ddagger}$. Let $t_{20} \overset{\text{def}}{=} t_1 + \Delta_{\mathsf{Coop}}^{\mathsf{Event1,max}}$, and $t_{21} \overset{\text{def}}{=} t_{20} + \Delta_{\mathsf{nonzeno}}$. Then we have $t_{19} \leqslant t_{20} < t_{21} \leqslant t_1 + \Delta_{\mathsf{reset}}$ (due to (17), (20), (19) and the definition of $\delta_{\mathsf{low}}^{\mathsf{RF1}}$, see Fig. 4 "Event1").

Meanwhile, due to (†), a second coop-duration will not start during $(t_1, t_1 + \Delta_{\mathsf{reset}}]$. (††)

Due to (††), V_i is in "Init" at $\forall t \in (t_{20}, t_{21})$. Without loss of generality, pick $t_{22} \overset{\text{def}}{=} \frac{t_{20}+t_{21}}{2}$. Meanwhile, (††) also implies no other target lane CAV is in a coop-duration at t_{22}. Hence due to Lemma 3, $V_1 \sim V_n$ are all in "Init" mode at t_{22}. († † †)

In order to determine R's mode at t_{22}, note V_i's "Event1" sends the "LaneChangeAcpt" wireless packet to R. This can have two cases.

Case 2.1.1.1: If R receives the "LaneChangeAcpt" packet at t_1^+, then it will enter the "ConstSpeedTargetLane" mode by $t_1 + \delta_{\mathsf{defer}}^{\mathsf{lc}} + \delta_{\mathsf{lc}}(v_{\mathsf{lim}}) = t_1 + \delta_{\mathsf{low}}^{\mathsf{RF1}} + \delta_{\mathsf{d}}^{\ddagger} \leqslant t_{20} < t_{22}$.

Case 2.1.1.2: If R loses the "LaneChangeAcpt" packet at t_1^+, then at t_{22}, R must be in mode "Init" (see Fig. 3). This is because $t_1 + \Delta_{\mathsf{reset}} > t_{22} > t_{20} = t_1 + \Delta_{\mathsf{Coop}}^{\mathsf{Event1,max}} > t_1 + \Delta_{\mathsf{nonzeno}}$ (due to (**c5**)).

Combining **Case 2.1.1.1**, **Case 2.1.1.2**, and († † †), we see t_{22} is a valid time instance, hence **Case 2.1.1** complies with the claim.

Case 2.1.2 The reception of the "LaneChangeReq" packet triggers "Event2" of V_i (see Fig. 4) at t_1. Then a coop-duration starts at t_1 and ends at $t_{23} \overset{\text{def}}{=} t_1 + \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{low}}^{\mathsf{RF2}} + \delta_{\mathsf{a}}^{\ddagger}$. Let $t_{24} \overset{\text{def}}{=} t_1 + \Delta_{\mathsf{Coop}}^{\mathsf{Event2,max}} + \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{lc}}(v_{\mathsf{low}}) + \Delta^*$ and $t_{25} \overset{\text{def}}{=} t_{24} + \Delta_{\mathsf{nonzeno}}$. Then we have $t_{23} \leqslant t_{24} < t_{25} \leqslant t_1 + \Delta_{\mathsf{reset}}$ (due to (18), (20), (19) and the definition of $\delta_{\mathsf{low}}^{\mathsf{RF2}}$, see Fig. 4 "Event2").

Meanwhile, due to (†), a second coop-duration will not start during $(t_1, t_1 + \Delta_{\mathsf{reset}}]$. († † ††)

Due to († † ††) V_i is in "Init" at $\forall t \in (t_{24}, t_{25})$. Without loss of generality, pick $t_{26} \overset{\text{def}}{=} \frac{t_{24}+t_{25}}{2}$. Meanwhile, († † ††) also implies no other target lane CAV is in a coop-duration at t_{26}. Hence due to Lemma 3, $V_1 \sim V_n$ are all in "Init" mode at t_{26}. († † † † †)

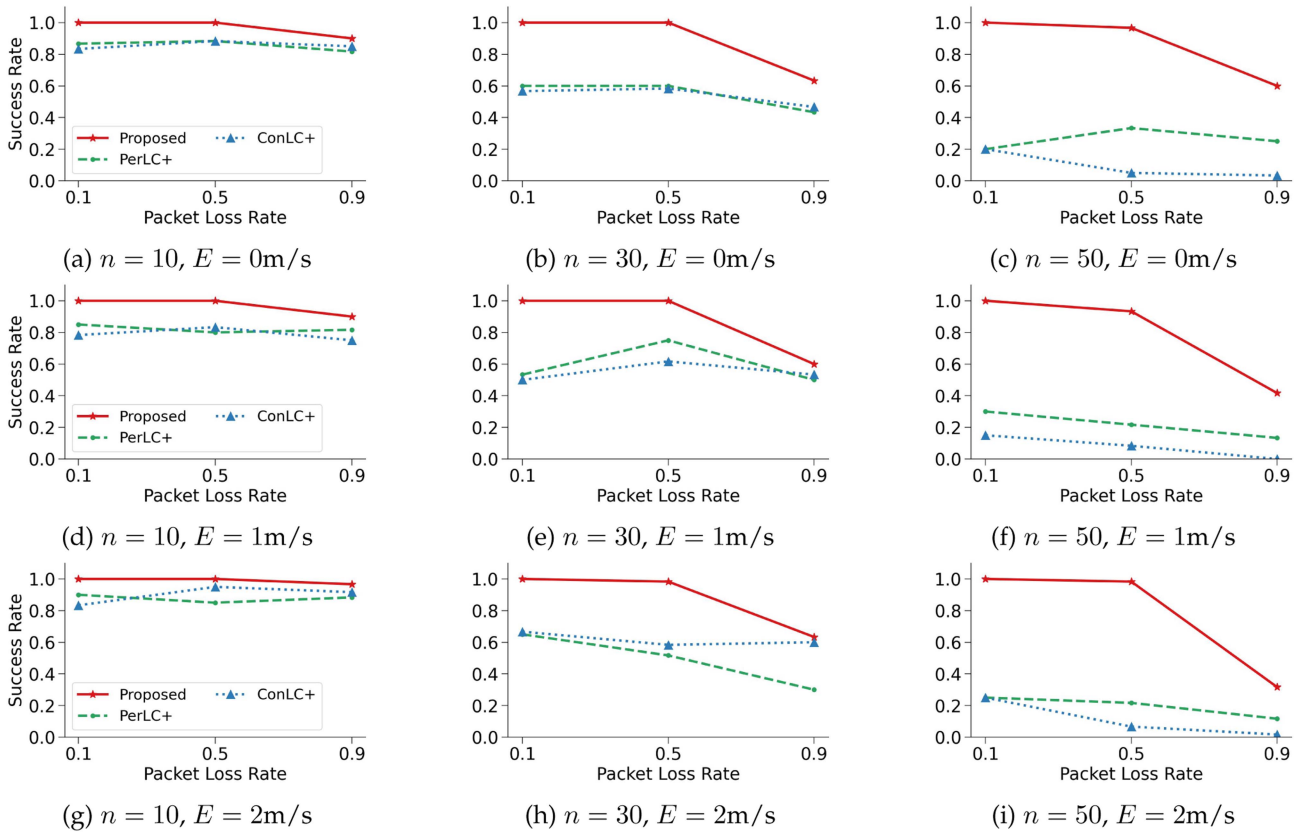


Fig. 7. Statistics of lane change *success rates*. X-axes: wireless packet loss rate P (0.1, 0.5, and 0.9, respectively, mean $P = 10\%$, 50%, and 90%). Y-axes: success rate. n is the number of target lane CAVs. E is the accumulated velocity disturbance's magnitude bound.

To decide R's mode at t_{26}, note V_i's "Event2" sends the "Decelerate" wireless packet to R. This can have two cases.

Case 2.1.2.1: If R receives the "Decelerate" packet at t_1^+, then it will enter the "ConstSpeedTargetLane" mode by

$$\begin{aligned}
&t_1 + \delta_{\mathsf{defer}}^{\mathsf{Rdec}} + \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{low}}^{\mathsf{LR}} + \delta_{\mathsf{lc}}(v_{\mathsf{low}}) + \delta_{\mathsf{a}}^{\ddagger} \\
&= t_1 + \frac{D_2 - d_0(R,F)}{\widetilde{v}} + \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{d}}^{\ddagger} + \frac{v_{\mathsf{lim}}\Delta^* - d_0(L,R)}{\widetilde{v}} \\
&\quad + \delta_{\mathsf{lc}}(v_{\mathsf{low}}) + \delta_{\mathsf{a}}^{\ddagger} \\
&= t_1 + \frac{D_2}{\widetilde{v}} + \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{a}}^{\ddagger} + \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{lc}}(v_{\mathsf{low}}) \\
&\quad + \frac{v_{\mathsf{lim}}\Delta^* - (d_0(R,F) + d_0(L,R))}{\widetilde{v}} \\
&\leqslant t_1 + \Delta_{\mathsf{Coop}}^{\mathsf{Event2,max}} + \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{lc}}(v_{\mathsf{low}}) + \frac{v_{\mathsf{lim}}\Delta^* - v_{\mathsf{low}}\Delta^*}{\widetilde{v}} \\
&\quad \text{(due to Lemma 5 and } F\text{'s speed} \geqslant v_{\mathsf{low}}) \\
&= t_1 + \Delta_{\mathsf{Coop}}^{\mathsf{Event2,max}} + \delta_{\mathsf{d}}^{\ddagger} + \delta_{\mathsf{lc}}(v_{\mathsf{low}}) + \Delta^* = t_{24} < t_{26}.
\end{aligned}$$

Case 2.1.2.2: If R loses the "Decelerate" packet at t_1^+, then at t_{26}, R must be in mode "Init" (see Fig. 3). This is because $t_{26} > t_{24} > t_1 + \Delta_{\mathsf{Coop}}^{\mathsf{Event2,max}} > t_1 + \Delta_{\mathsf{nonzeno}}$ (due to (**c5**)) and $t_{26} < t_{25} \leqslant t_1 + \Delta_{\mathsf{reset}}$.

Combining **Case 2.1.2.1**, **Case 2.1.2.2**, and († † † † †), we see t_{26} is a valid time instance, hence **Case 2.1.2** complies with the claim.

Combining **Case 2.1.1** and **Case 2.1.2**, **Case 2.1** compiles with the claim.

Case 2.2: $F = V_i$ loses the "LaneChangeReq" packet at t_1 (see event "GotLaneChangeReq" in Fig. 4). Then, due to (†), nothing happens to $V_1 \sim V_n$ during $[t_1, t_1 + \Delta_{\mathsf{reset}})$. Due to Lemma 3, $\forall t \in [t_1, t_1 + \Delta_{\mathsf{reset}})$, $V_1 \sim V_n$ are all in "Init" mode at t. († † † † ††)

Let $t_{27} \overset{\mathsf{def}}{=} t_1 + \Delta_{\mathsf{Coop}}^{\mathsf{max}}$. Then R must be in "Init" mode at t_{27}, as $t_1 + \Delta_{\mathsf{nonzeno}} < t_{27} < t_1 + \Delta_{\mathsf{reset}}$. Meanwhile, due to († † † † ††), t_{27} is a valid time instance. Hence **Case 2.2** compiles with the claim.

Combining **Case 2.1** and **Case 2.2**, **Case 2** complies with the claim.

Combining **Case 1** and **Case 2**, the claim sustains. □ (∗∗)

Due to (∗) and (∗∗), the theorem sustains. □

V. Further Discussions

A. V2V Communication Failures Between Target Lane CAVs are Irrelevant

V2V communications between the target lane CAVs (if used) are only used in the "Sync" mode (see Fig. 4); and are only used between two consecutive target lane CAVs (V_i and V_{i+1}, where

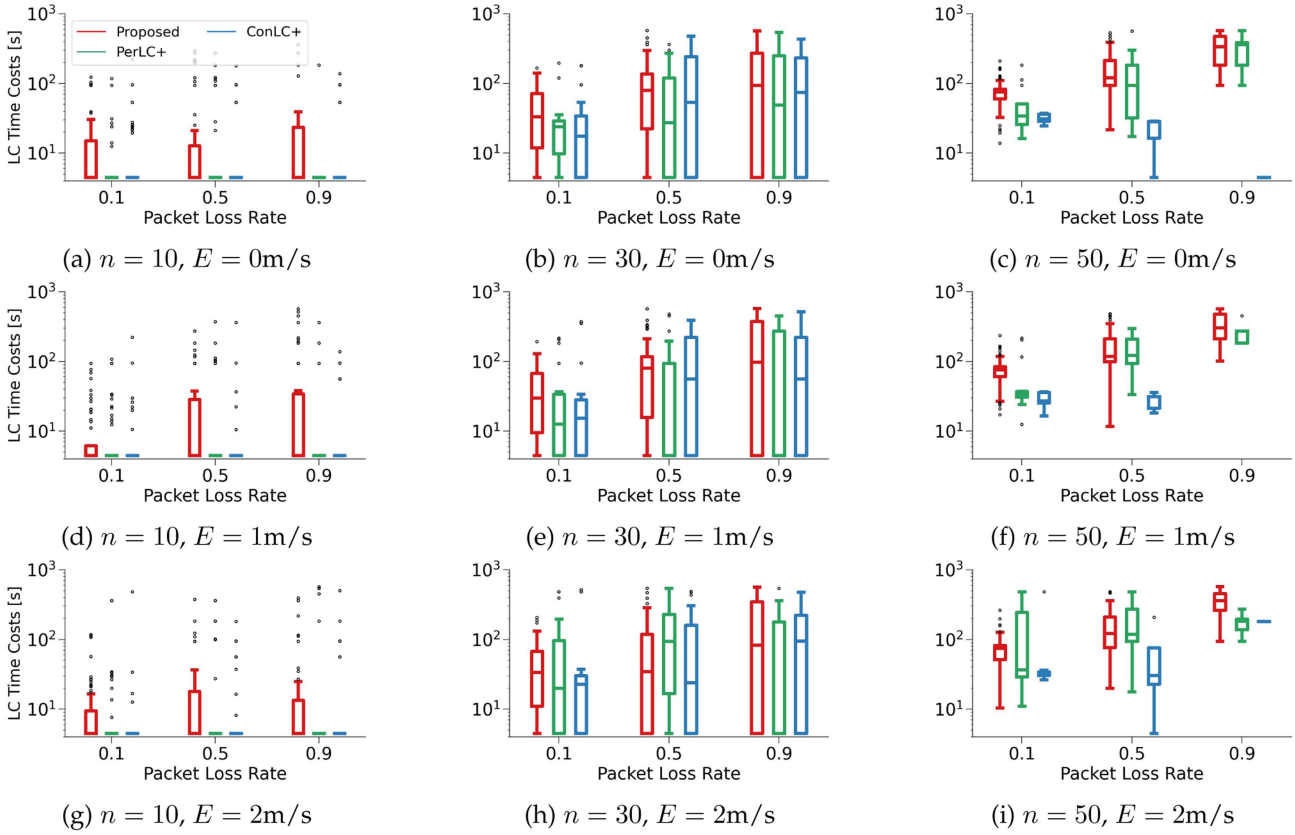


Fig. 8. Statistics of lane change time costs. X-axes: wireless packet loss rate P (0.1, 0.5, and 0.9, respectively, mean $P = 10\%$, 50%, and 90%). Y-axes: lane change time costs (unit: second). n is the number of target lane CAVs. E is the accumulated velocity disturbance's magnitude bound.

$i = 1, 2, \ldots, n-1$) for three possible cases: to trigger the "StartSyncPred" event, to let V_i inform V_{i+1} of the former's current distance/velocity/acceleration, or to trigger the "StopSyncPred" event. For all these three cases, the corresponding V2V communications can be replaced by V_{i+1}'s local ranging sensors (see Assumption 4). Hence the correponding V2V communications failures are irrelevant.

In case the ranging sensors need *line-of-sight*, we have the following observations. All the target lane CAVs that should be in "Sync" at any time instance t must be following a unique target lane CAV V_i ($i \in \{1, 2, \ldots, n\}$) that is in a coop-duration. This implies V_i must be behind R at t, if R is after all on the target lane at t. Therefore, it is impossible that R resides between two *speed synchronizing* target lane CAVs (i.e. the predecessor CAV is in a coop-duration, while the follower CAV is in "Sync"; or both are in "Sync") at t. Therefore, the line-of-sight between two speed synchronizing target lane CAVs is available at t.

B. Vehicle Body Shape

In Section III-C, right after Definition 1, we discussed how to take into consideration of vehicle body length in same lane CTH-Δ^* safety design. Now let us discuss how to consider vehicle body shape in a mechanical lane change routine.

So far, we have abstracted the target lane as an axis (simplified as the "*target lane axis*" in this subsection), and CAVs as points (simplified as the "*CAV points*" in this subsection). Suppose the earliest time instance that the CAV R point touches the target lane axis is t_{45}, and (on the target lane axis) the touch-point's X-coordinate is X_1. Meanwhile, suppose the earliest time instance that R's body touches the lane border between the current and target lane (see Fig. 1) is t_{46}. As we only allow limited number of mechanical lane change routines, there is an upper bound to $|t_{45} - t_{46}|$, denote it as $\Delta_{\mathsf{half\_lc}}$.

Theorem 1 ensures at t_{45}, the requesting CAV R point is at least $v_{\mathsf{low}}\Delta^*$ ahead of the follower CAV F point. Therefore $\forall t \in [t_{46}, t_{45}]$, the follower CAV F point's X-coordinate is no greater than $X_1 - v_{\mathsf{low}}\Delta^* - v_{\mathsf{low}}\delta$, where $\delta \stackrel{\text{def}}{=} t_{45} - t$, hence $0 \leqslant \delta \leqslant \Delta_{\mathsf{half\_lc}}$. Meanwhile, as v_{lim} is also the maximum projected speed on the X-axis during any mechanical lane change routine, at t, any point on the CAV R's body has an X-coordinate of no less than $X_1 - v_{\mathsf{lim}}\delta - D_0$, where D_0 is the maximum vehicle body length.

Hence at t, any point on R's body is at least $X_1 - v_{\mathsf{lim}}\delta - D_0 - (X_1 - v_{\mathsf{low}}\Delta^* - v_{\mathsf{low}}\delta) \geqslant v_{\mathsf{low}}\Delta^* - D_0 - \widetilde{v}\Delta_{\mathsf{half\_lc}}$ ahead of the follower CAV F point along the X-axis.

VI. Evaluation

Next, we carry out simulations to verify our proposed protocol, especially the CTH-Δ^* safety, liveness (automatic resetting), and performance of the lane change protocol.

A. Simulation Configuration

Our simulator adopts the following configurations typical to a vehicle (see the classic textbook of [51]): $\Delta_{\mathsf{nonzeno}} = 0.1$s; $\Delta^* = 6$s; $v_{\mathsf{lim}} = 25$ m/s; $v_{\mathsf{low}} = 20$m/s; lane-change, acceleration, and deceleration strategies are set as per [51], which imply $\delta_{\mathsf{lc}}(v_{\mathsf{lim}}) = 4.51$s, $d^{\mathsf{X}}_{\mathsf{lc}}(v_{\mathsf{lim}}) = 112.5573$m, $\delta_{\mathsf{lc}}(v_{\mathsf{low}}) = 4.72$s, $d^{\mathsf{X}}_{\mathsf{lc}}(v_{\mathsf{low}}) = 94.1975$ m, $\delta^{\ddagger}_{\mathsf{a}} = 4.65$s, $d^{\ddagger}_{\mathsf{a}} = 105.0914$m, $\delta^{\ddagger}_{\mathsf{d}} = 1.97$s, and $d^{\ddagger}_{\mathsf{d}} = 44.955$m. The above further decides other parameters: $D_1 \sim D_3$ (see (11), (12), (13)), D_{Sync} (see (14)), and Δ_{reset} (see (20)).

The simulator can be configured to mimic different traffic loads, wireless packet loss rates, and velocity disturbances. We assume a target lane segment of length 10,000 meters with n target lane CAVs distributed along it. We set n to 10, 30, and 50 respectively for mild, moderate, and loaded[3] traffic loads. The wireless packet loss rate P is set to 0.1 (i.e. 10%), 0.5 (i.e. 50%), and 0.9 (i.e. 90%) respectively for mild, moderate, and severe wireless packet losses. For each CAV, a *velocity disturbance* uniformly distributed in $[-0.1, 0.1]$m/s is added in each simulation step (i.e. 0.01 s of the simulated time); meanwhile, at any simulated time instance, the accumulated velocity disturbance's magnitude is bounded by E, a constant set to 0 m/s, 1 m/s, and 2 m/s respectively for zero, moderate, and severe velocity disturbances.

For each given collaborative driving protocol, n, P, and E combination, we run 60 simulation trials. Each trial simulates 10 minutes (unless in some exception cases, see the notes in Section VI-D) of a lane change scenario. At the beginning of a simulation trial, the n target lane CAVs are generated along the target lane segment $[-5000$m, 5000m] as per pseudo uniform distribution, which takes into consideration of Assumption 5. Specifically, the pseudo code is as follows:

Step 1 Set $\mathcal{V}$ to $\varnothing$;

Step 2 If $|\mathcal{V}| \geqslant n$, terminate; otherwise

Step 2.1 Randomly pick a candidate target lane CAV initial location p on the target lane segment $[-5000$m, 5000m] as per uniform distribution;

Step 2.2 If p does not violate CTH-Δ^* safety rule with the elements already in $\mathcal{V}$, then add p into $\mathcal{V}$; otherwise, ignore p;

Step 2.3 Go back to **Step 2**.

The generated $\mathcal{V}$ is the initial locations for the target lane CAVs for the trial.

B. Comparison Baselines

As mentioned in Sections I and II, few works cover CTH safety guarantee under arbitrary wireless packet losses. Perhaps the closest works are Wang et al. [21]'s PerLC and ConLC protocols for V2V CAVs (see Section I), where ACKs from the target lane leader CAV L and follower CAV F are obligatory for starting the mechanical lane change routine. ConLC improves PerLC in terms of using multicast instead of one by one unicast, and brakes the requesting CAV R to change the relationship to L and F (include letting the current L and F pass, and change to a new pair of L and F).

However, CTH safety guarantee is not covered by Wang et al. [21]; and how to adapt the PerLC and ConLC protocols to guarantee CTH safety under arbitrary wireless packet losses are still open problems. Nevertheless, in order to provide comparison baselines to evaluate our proposed protocol, we propose our version of adaptations to the PerLC and ConLC protocols, respectively named PerLC+ and ConLC+. PerLC+ is basically a subset of our proposed protocol. The target lane CAV hybrid automata A_i ($i = 1, 2, \ldots, n$, see Fig. 4) remain unchanged. The requesting CAV R's hybrid automaton A_R (see Fig. 3) has the "Event2" and "Event4" disabled (returns to "Init" mode with τ reset to 0in case of "Event2" and "Event4"). We call the resulted hybrid automaton A'_R. ConLC+ evolves PerLC+. Specifically, it evolves A'_R to A''_R by replacing the "RequestTimeout1" event with the following routine:

Step 1 Decelerate to v_{low};

Step 2 Wait for $\Delta^{\mathsf{Event1,max}}_{\mathsf{Coop}} + \Delta_{\mathsf{nonzeno}}$, then find the new leader CAV L and follower CAV F and measure the new $d_0(L, R)$ and $d_0(R, F)$;

Step 3 If $d_0(L, R) \geqslant v_{\mathsf{lim}}(\Delta^* - \delta_{\mathsf{lc}}(v_{\mathsf{low}})) + d^{\mathsf{X}}_{\mathsf{lc}}(v_{\mathsf{low}})$ and $d_0(R, F) \geqslant v_{\mathsf{lim}}(\Delta^* + \delta_{\mathsf{lc}}(v_{\mathsf{low}})) - d^{\mathsf{X}}_{\mathsf{lc}}(v_{\mathsf{low}}) + \widetilde{d}_{\mathsf{a}}(v_{\mathsf{lim}})$, change lane via mechanical lane change routine $\mathsf{lc}(v_{\mathsf{low}}, \tau)$ ($\tau \in [0, \delta_{\mathsf{lc}}(v_{\mathsf{low}})]$) and then accelerate to v_{lim}; else go to **Step 2**.

Following the proof for Theorem 1**Claim 1**, we can prove PerLC+ and ConLC+ guarantee the CTH-Δ^* safety.

C. CTH-Δ^* Safety

We first verify Theorem 1 **Claim 1** on CTH-Δ^* safety guarantee, which answers Research Problem 1.

Fig. 5 plots the statistics of sampled time headways (relative to the respective immediate predecessor CAVs, see Def. 1) of all CAVs in all simulation trials (for each CAV simulated, its time headway is sampled every 0.08 s).

Fig. 5(a) (b) (c) show the time headway statistics under zero velocity disturbance ($E = 0$m/s). Such disturbance complies with the prerequisites of Theorem 1. Accordingly, for all the three protocols (our proposed, PerLC+, ConLC+), under all three wireless packet loss rates ($P = 10\%$, 50%, 90%), and all three traffic loads ($n = 10, 30, 50$), the time headways are always no less than 6.0 s (remember $\Delta^* = 6$s, see Section VI-A). This means the CTH-Δ^* safety always holds.[4]

Fig. 5(d) (e) (f) and (g) (h) (i) respectively show the time headway statistics under moderate ($E = 1$m/s) and severe ($E = 2$m/s) velocity disturbances. Such disturbances violate the prerequisites of Theorem 1. However, according to Fig. 5(d) (e) (f), under moderate velocity disturbances, CTH-Δ^* safety can still roughly hold. According to Fig. 5(g) (h) (i), under severe velocity disturbances, CTH-Δ^* safety are significantly violated; but even in the worst case, all CAVs maintain a time headway of at least 2.24 s.

[3]Note given the CTH safety demands $v_{\mathsf{lim}}\Delta^* = 150$ m distance between any two consecutive CAVs on a same lane, our 10,000m lane segment can host at the most $10,000/150 < 67$ CAVs.

[4]Note our computer simulation's time granularity is 0.01 s, hence our minimum time headway value is rounded to one digit after the floating point.

D. Liveness (Automatic Resetting)

Next, we verify Theorem 1 **Claim 2** on liveness (i.e. automatic resetting), which answers Research Problem 2.

The claim regards two states of the holistic system to be stable: **Stable State 1** and **Stable State 2**. **Stable State 1** is the initial state of the holistic system; while **Stable State 2** is the desired end state of the holistic system, where R successfully changes lane into the target lane, and together with all target lane CAVs regain the constant speed of v_{lim}.

Theorem 1 **Claim 2** basically says every time the holistic system becomes unstable (say at time $t_1 \in [t_0, +\infty)$), it will reset itself either back to **Stable State 1** or to **Stable State 2** within finite duration $(t_2 - t_1)$, where $t_2 \in (t_1, t_1 + \Delta_{\mathsf{reset}}]$, and Δ_{reset} is a constant defined by (20). For ease of narration, we call $(t_2 - t_1)$ the "*reset time cost*."

Fig. 6 shows that both our proposed protocol and the PerLC+ protocol comply with Theorem 1 **Claim 2** in all cases of traffic load ($n = 10$, 30, 50) and wireless packet loss rates ($P = 10\%$, 50%, 90%) when there is no velocity disturbances ($E = 0$m/s). The maximum reset time costs observed are always below $\Delta_{\mathsf{reset}} = 89.6795$s as per our simulation configuration (see Section VI-A and (20)). The compliance even holds when under moderate or severe velocity disturbances ($E = 1$m/s, 2 m/s). ConLC+, however, cannot comply with Theorem 1**Claim 2**, even in the most amicable case of $n = 10$, $P = 10\%$, and $E = 0$m/s.

Also note that normally each trial of our simulation lasts 10 minutes (in the simulated world). But in case by the end of the 10th minute, the system is still waiting for a reset to take place, the simulation will go on until the reset takes place (i.e. reaching **Stable State 1** or **Stable State 2**).

E. Lane Change Success Rate and Time Cost

Finally, we evaluate the lane change success rates and time costs, which answers Research Problem 3.

A *lane change success rate* refers to the ratio of simulation trials (under given protocol, n, P, and E) that end with **Stable State 2** (see Section VI-D and Theorem 1 **Claim 2**); while a *lane change time cost* refers to the duration from the start of a simulation trial to the time instance when the holistic system reaches **Stable State 2** in the simulated world.

Fig. 7 shows the lane change *success rates* of all three protocols (our proposed, PerLC+, ConLC+) under different traffic loads ($n = 10$, 30, and 50), wireless packet loss rates ($P = 10\%$, 50%, 90%), and velocity disturbances ($E = 0$ m/s, 1m/s, 2m/s). According to the figure, for any given traffic load n, wireless packet loss rate P, and velocity disturbances E, our proposed protocol's success rates always outperform those of PerLC+ and ConLC+; and the differences become more significant when the traffic load deteriorates. For all the 27 configurations of n, P, and E values, our proposed protocol can improve the success rate by $9.4\% \sim 400.0\%$ (with a median of 66.7%) compared to the PerLC+ protocol; and our proposed protocol can improve the success rate by $5.3\% \sim +\infty\%$ (with a median of 62.2%) compared to the ConLC+ protocol.

For those trials that succeed in the end, Fig. 8 counts the total time costs of lane change. Our proposed protocol's time costs are not always the shortest. This is because both PerLC+ and ConLC+ mainly succeed under benign driving conditions, hence lane change time costs are likely low; while our proposed protocol succeeds in both benign and harsh driving conditions, the latter usually costs more time to handle (e.g. requires more complex driving cooperation). Encouragingly, even under such biased comparisons, the time costs statistics of our proposed protocol are still comparable to those of PerLC+ and ConLC+.

Also note the lane change time cost is tightly related to the choice of the system configuration parameters (see Section VI-A). However, how to optimize the choice of the configuration parameters is not the focus of this paper. Instead, our simulation adopts classic configurations for vehicles and drivings in our simulations (see Section VI-A and [51]).

VII. Conclusion

In this paper, we propose a protocol to realize the safe lane change of automatic driving of CAVs in dedicated highway lanes. We formally prove that the protocol can always guarantee the CTH safety and liveness, even under arbitrary wireless packet losses. These theoretical claims are verified by our simulations, which also show the lane change time costs are satisfactory. Comparing to the other two lane change protocols, our proposed protocol can achieve significantly better success rates, particularly under adverse traffic loads. Furthermore, extensive simulations have demonstrated the robustness of our protocol's safety, liveness, success rates, and total time costs in the presence of random vehicle velocity disturbances.

Acknowledgment

The authors would like to thank the comments by Shiyu Zhang, Zhihao Zhao, Haoyang Song, Yibin Zhang, Zewen Pan, Haolin Zhang, Dylan Haryoto, and anonymous reviewers on improving this paper.

References

[1] H. Chen, F. Wu, K. Hou, and T. Z. Qiu, "Leveraging dynamic right-of-way allocation and tolling policy for CAV dedicated lane management to promote CAV and improve mobility," *IEEE Trans. Intell. Transp. Syst.*, vol. 25, no. 7, pp. 6667–6676, Jul. 2024.

[2] L. Claussmann, M. Revilloud, D. Gruyer, and S. Glaser, "A review of motion planning for highway autonomous driving," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 5, pp 1826–1848, May 2020.

[3] T. Krisher et al., "Michigan plans dedicated road lanes for autonomous vehicles," *ABC News*, Aug. 2020. [Online]. Available: https://abcnews.go.com

[4] W. Liu et al., "A systematic survey of control techniques and applications in connected and automated vehicles," *IEEE Internet Things J.*, vol. 10, no. 24, pp. 21892–21916, Dec. 2023.

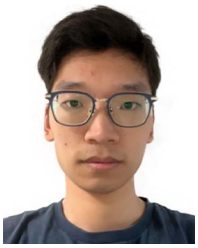
[5] A. Rammohan, "Revolutionizing intelligent transportation systems with cellular vehicle-to-everything (C-V2X) technology: Current trends, use cases, emerging technologies, standardization bodies, industry analytics and future directions," *Veh. Commun.*, vol. 43, Oct. 2023, Art. no. 100638.

[6] N. E. Yunus, S. F. A. Razak, S. Yogarayan, and M. F. A. Abdullah, "Lane changing models: A short review," in *Proc. IEEE 12th Control System Graduate Res. Colloq.*, 2021, pp. 110–115.

[7] R. Molina-Masegosa and J. Gozalvez, "LTE-V for sidelink 5G V2X vehicular communications: A new 5G technology for short-range vehicle-to-everything communications," *IEEE Veh. Technol. Mag.*, vol. 12, no. 4, pp. 30–39, Dec. 2017.

[8] G. Araniti, C. Campolo, M. Condoluci, A. Iera, and A. Molinaro, "LTE for vehicular networking: A survey," *IEEE Commun. Mag.*, vol. 51, no. 5, pp. 148–157, May 2013.
[9] S. Hüsges, M. Meuleners, N. Bateni, and C. Degen, "Simulation and measurement for sidelink communication between cars and bicycles," in *Proc. 16th Eur. Conf. Antennas Propag.*, 2022, pp. 1–5.
[10] L. Lusvarghi, C. A. Grazia, M. Klapez, M. Casoni, and M. L. Merani, "Awareness messages by vulnerable road users and vehicles: Field tests via LTE-V2X," *IEEE Trans. Intell. Veh.*, vol. 8, no. 10, pp. 4418–4433, Oct. 2023.
[11] L. Bréhon-Grataloup, R. Kacimi, and A. -L. Beylot, "Field trial for enhanced V2X multi-rat handover in autonomous vehicle networks," in *Proc. IEEE 48th Conf. Local Comput. Netw.*, 2023, pp. 1–8.
[12] J. Kim, H. Chung, I. Kim, and G. Noh, "Integration of 5G mmWave-enabled V2I and V2V: Experimental evaluation," *IEEE Commun. Mag.*, vol. 62, no. 1, pp. 104–110, Jan. 2024.
[13] S.-W. Choi et al., "V2I and V2V service demonstration of millimeter wave communication in urban road environment," in *Proc. Int. Conf. Inf. Commun. Technol. Convergence*, 2023, pp. 756–759.
[14] M. Mikami, Y. Ishida, K. Serizawa, H. Nishiyori, K. Moto, and H. Yoshino, "Field experimental trial of dynamic mode switching for 5 G NR-V2X sidelink communications towards application to truck platooning," in *Proc. IEEE 91st Veh. Technol. Conf.*, 2020, pp. 1–5.
[15] H. Pirayesh and H. Zeng, "Jamming attacks and anti-jamming strategies in wireless networks: A comprehensive survey," *IEEE Commun. Surveys Tuts.*, vol. 24, no. 2, pp. 767–809, Secondquarter 2022.
[16] F. V. Monteiro and P. Ioannou, "Safe lane change and merging gaps in connected environments," *IFAC-PapersOnLine*, vol. 54, no. 2, pp. 69–74, 2021.
[17] C. Viel, U. Vautier, J. Wan, and L. Jaulin, "Platooning control for heterogeneous sailboats based on constant time headway," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 5, pp. 2078–2089, May 2020.
[18] D. Swaroop and K. R. Rajagopal, "A review of constant time headway policy for automatic vehicle following," in *Proc. IEEE Intell. Transp. Syst.*, 2001, pp. 65–69.
[19] S. Tsugawa, S. Kato, K. Tokuda, T. Matsui, and H. Fujii, "A cooperative driving system with automated vehicles and inter-vehicle communications in Demo 2000," in *Proc. IEEE Intell. Transp. Syst.*, 2001, pp. 918–923.
[20] A. H. Sakr, G. Bansal, V. Vladimerou, and M. Johnson, "Lane change detection using V2V safety messages," in *Proc. 21st IEEE Int. Conf. Intell. Transp. Syst.*, 2018, pp. 3967–3973.
[21] L. Wang, R. F. Iida, and A. M. Wyglinski, "Coordinated lane changing using V2V communications," in *Proc. 88th IEEE Veh. Technol. Conf.*, 2018, pp. 1–5.
[22] F. Tan, Y. Wang, Q. Wang, L. Bu, and N. Suri, "A lease based hybrid design pattern for proper-temporal-embedding of wireless CPS interlocking," *IEEE Trans. Parallel Distrib. Syst.*, vol. 26, no. 10, pp. 2630–2642, Oct. 2015.
[23] C. Gray and D. Cheriton, "Leases: An efficient fault-tolerant mechanism for distributed file cache consistency," in *Proc. the 20th ACM SIGOPS Operating Syst. Rev.*, Dec. 1989, pp. 202–210.
[24] T. Li et al., "A cooperative lane change model for connected and automated vehicles," *IEEE Access*, vol. 8, pp. 54940–54951, 2020.
[25] Y. Luo, G. Yang, M. Xu, Z. Qin, and K. Li, "Cooperative lane-change maneuver for multiple automated vehicles on a highway," *Automot. Innov.*, vol. 2, no. 3, pp. 157–168, 2019.
[26] H. Wang, W. Hao, J. So, Z. Chen, and J. Hu, "A faster cooperative lane change controller enabled by formulating in spatial domain," *IEEE Trans. Intell. Veh.*, vol. 8, no. 12, pp. 4685–4695, Dec. 2023.
[27] K. Sun, X. Zhao, and X. Wu, "A cooperative lane change model for connected and autonomous vehicles on two lanes highway by considering the traffic efficiency on both lanes," *Transp. Res. Interdiscipl. Perspectives*, vol. 9, 2021, Art. no. 100310.
[28] B. Li, Y. Zhang, Y. Feng, Y. Zhang, Y. Ge, and Z. Shao, "Balancing computation speed and quality: A decentralized motion planning method for cooperative lane changes of connected and automated vehicles," *IEEE Trans. Intell. Veh.*, vol. 3, no. 3, pp. 340–350, Sep. 2018.
[29] H. Zhang, L. Du, and J. Shen, "Hybrid MPC system for platoon based cooperative lane change control using machine learning aided distributed optimization," *Transp. Res. Part B, Meth.*, vol. 159, pp 104–142, 2022.
[30] W. Zhou, D. Chen, J. Yan, Z. Li, H. Yin, and W. Ge, "Multi-agent reinforcement learning for cooperative lane changing of connected and autonomous vehicles in mixed traffic," *Auton. Intell. Syst.*, vol. 2, no. 5, pp 1–11, 2022.
[31] A. Bazzi, A. O. Berthet, C. Campolo, B. M. Masini, A. Molinaro, and A. Zanella, "On the design of sidelink for cellular V2X: A literature review and outlook for future," *IEEE Access*, vol. 9, pp. 97953–97980, 2021.
[32] T. S. G. Services and S. Aspects, "Enhancement of 3GPP support for V2X scenarios," 3GPP, Sophia Antipolis, France, Tech. Rep. TS 22.186,3, 2022.
[33] T. S. G. Services and S. Aspects, "Study on enhancement of 3GPP support for 5 G V2X services," 3GPP, Sophia Antipolis, France, Tech. Rep. TR 22.886, 12, 2018.
[34] Cohda Wireless. (2025) MK6C EVK, 2025. [Online]. Available: https://www.cohdawireless.com/solutions/hardware/
[35] Quectel. (2024) AG15 automotive module, 2024. [Online]. Available: http://www.quectel.com/product/c-v2x-ag15
[36] Cohda Wireless. (2025) MK6, 2025. [Online]. Available: https://www.cohdawireless.com/solutions/mk6/
[37] T. S. G. R. A. Network, "Study on nr sidelink relay," 3GPP, Sophia Antipolis, France, Tech. Rep. TR 38.836,3, 2021.
[38] T. S. G. R. A. Network, "Study on integrated access and backhaul," 3GPP, Sophia Antipolis, France, Tech. Rep. TR 38.874,12, 2018.
[39] M. Nouri, M. G. Shayesteh, H. Behroozi, and Z. Ding, "Digital/hybrid beamforming via KLMS algorithm in presence of mutual coupling with experimental evaluation for 5 G and V2V communications," *IEEE Trans. Veh. Technol.*, vol. 73, no. 4, pp. 5229–5242, Apr. 2024.
[40] G. Twardokus and H. Rahbari, "Evaluating V2V security on an SDR testbed," in *Proc. IEEE INFOCOM Workshops*, 2021, pp. 1–3.
[41] S. An and K. Chang, "Enhancing reliability in 5 G NR V2V communications through priority-based groupcasting and IR-HARQ," *IEEE Access*, vol. 11, pp. 72717–72731, 2023.
[42] Y. Pu, F. Wen, Y. Zeng, and S. Zhou, "Comparative study on outage probability of mmWave vehicle-to-vehicle communications," in *Proc. IEEE Glob. Comm. Conf.*, 2023, pp. 5623–5628.
[43] O. Elgarhy, O. Yener, and M. M. Alam, "Performance evaluation of 5G NR sidelink for V2X scenarios," in *Proc. Biennial Baltic Electro. Conf.*, 2022, pp. 1–5.
[44] X. Fan, "Cooperative driving for connected and automated vehicles on dedicated lanes," Ph.D. dissertation, The Hong Kong Polytechnic Univ., Hong Kong SAR, China, Apr. 2022. [Online]. Available: https://theses.lib.polyu.edu.hk/handle/200/12155
[45] M. K. Shalaby, A. Farag, O. M. AbdelAziz, D. M. Mahfouz, O. M. Shehata, and E. I. Morgan, "Design of various dynamical-based trajectory tracking control strategies for multi-vehicle platooning problem," in *Proc. IEEE Intell. Transp. Syst. Conf.*, Oct. 2019, pp. 1631–1637.
[46] M. Wang, "Infrastructure assisted adaptive driving to stabilise heterogeneous vehicle strings," *Transp. Res. Part C*, vol. 91, pp 276–295, Apr. 2018.
[47] S. Lefevre, C. Sun, R. Bajcsy, and C. Laugier, "Comparison of parametric and non-parametric approaches for vehicle speed prediction," in *Proc. Amer. Control Conf.*, Jun. 2014, pp. 3494–3499.
[48] D. J. Chang and E. K. Morlok, "Vehicle speed profiles to minimize work and fuel consumption," *J. Transp. Eng.*, vol. 131, no. 3, pp 173–182, 2005.
[49] J. Axelsson, "Safety in vehicle platooning: A systematic literature review," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp 1033–1045, May 2017.
[50] K. C. Dey et al., "A review of communication, driver characteristics, and controls aspects of cooperative adaptive cruise control (CACC)," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 2, pp. 491–509, Feb. 2016.
[51] R. Rajamani, *Vehicle Dynamics and Control*. Berlin, Germany: Springer, 2011.
[52] R. Alur et al., "Hybrid automata: An algorithmic approach to the specification and verification of hybrid systems," in *Proc. Hybrid Syst.*, 1992, pp. 209–229.

Xueli Fan received the BS degree in automation from Shanxi University, Taiyuan, China, in 2012, the MS degree in control theory and control engineering from Beijing Jiaotong University, Beijing, China, in 2015, and the PhD degree from the Department of Computing, the Hong Kong Polytechnic University, Hong Kong, China, in 2022. Her research interests include cyber-physical-system (CPS) and smart transportation.



Jieming Chen received the BEng degree in electrical engineering from Shanghai Maritime University, Shanghai, China, in 2017, the MS degree in control engineering from the Technical University of Kaiserslautern, Kaiserslautern, Germany, in 2021. He is currently working toward the PhD degree in electrical engineering with Hong Kong Polytechnic University, Hong Kong SAR, China. His research focuses on the operations and control of emerging intelligent vehicles.

Qixin Wang received the BE and ME degrees from the Department of Computer Science and Technology, Tsinghua University, Beijing, China, in 1999 and 2001, respectively, and the PhD degree from the Department of Computer Science, University of Illinois at Urbana-Champaign in 2008. He joined the Department of Computing, Hong Kong Polytechnic University in 2009, as an Assistant professor. He is currently an associate professor.

Edward Chung received the Bachelor of Civil Engineering (Hons.) and PhD degrees from Monash University. He is currently a professor of intelligent transport systems (ITS) with the Department of Electrical Engineering, The Hong Kong Polytechnic University. Prior to joining PolyU in 2017, he was a professor with the Queensland University of Technology (QUT) and the director of the Smart Transport Research Centre, QUT.