Provably Correct Safety Protocol for Cooperative Platooning

Sebastian Mair and Matthias Althoff

***Abstract*— Cooperative platooning is a promising method for improving energy efficiency and traffic throughput on interstates. Ensuring collision avoidance is particularly difficult in platooning due to the small desired inter-vehicle spacing. We propose a safety protocol that can be applied to arbitrary controllers in platooning to prevent collisions in a provably correct manner while still realizing a small distance to the preceding vehicle. Our protocol intervenes as rarely and smoothly as possible, and its safety is ensured even if communication fails. In addition, we propose a safety protocol for consensus techniques where the vehicles of the platoon successively agree on a common braking limit. Our safety protocols are evaluated on various scenarios using the CommonRoad benchmark suite.**

I. INTRODUCTION

Platooning denotes the contactless formation of vehicles driving in a lane – typically performed on interstates. It aims to 1) reduce inter-vehicle distances in order to improve energy efficiency due to diminished aerodynamic drag, and 2) increase traffic throughput [1]. Platooning is often realized via cooperative adaptive cruise control (CACC), which denotes longitudinal vehicle control incorporating information communicated between vehicles. Due to the individual limitations of different CACC concepts [2], our safety concept is agnostic of the underlying CACC. Moreover, it intervenes with the CACC as little as possible in order to maintain its benefits, such as ensuring string stability.

To realize a particularly dense spacing and ensure string stability, recent work proposes consensus techniques to establish common dynamics among heterogeneous platoons [3]–[6]. In particular, a consensus on the allowed acceleration interval helps to avoid large safe distances. This is especially attractive for trucks, where the mass can significantly differ depending on the load, affecting the individual braking capabilities. Our safety concept includes a protocol for consensus techniques to ensure safety 1) before a consensus braking limit is reached, and 2) when the consensus is adjusted in case the composition of the platoon changes.

A. Related Work

Safety in platooning is usually defined as the strict absence of collisions, but also a weaker safety concept based on the potential damage caused by collisions has been proposed [7]. To ensure safe platooning, mainly two types of approaches are followed [8]: correct-by-construction controllers and online verification.

All authors are with the School of Computation, Information and Technology, Technical University of Munich, 85748 Garching, Germany
sebi.mair@tum.de, althoff@tum.de

1) Correct-by-construction Controllers: One group of methods designs controllers for keeping the state of platoons in invariably safe sets [7], [9], [10]. The notion of safety can also include the avoidance of collision propagations between different platoons [7]. Modelling a two-vehicle platoon as a pursuit-evasion game makes it possible to compute the largest possible set of initial states of the following vehicle, for which a safe controller exists [11].

2) Online Verification: By keeping a continuously updated fail-safe trajectory available [12], one can verify autonomous vehicles online. In [13], an online verification concept for adaptive cruise control is presented and evaluated for platooning, which uses fixed braking profiles as fail-safe trajectories. The work in [14] applies a similar approach to CACC, and [15] expands the idea in [13] to customizable fail-safe trajectories and considers cut-in vehicles by solving an optimization problem online. The work in [5] applies [15] to CACC by using less comfortable braking profiles in favor of smaller safe distances; however, this possibly leads to strong safety interventions. In [6], a CACC approach based on model predictive control is extended by safety constraints. However, the concept is not provably collision-free in continuous time. The work also proposes that platoon vehicles communicate their braking limits to the succeeding vehicles to minimize inter-vehicle distances.

Apart from safety mechanisms in normal operation, dangerous situations can be defused incorporating communication between vehicles [16]. A vehicle encountering an abnormal situation, e.g., a mechanical defect, traffic incidents, or hazardous surface conditions, can warn the other vehicles so that they can react accordingly [17], [18].

B. Contributions

We propose a generic safety protocol for cooperative platooning based on online verification. In particular, we contribute the following novelties:

- We prevent causing collisions in a provably correct manner under changing road inclines;
- we enable both rare and soft safety interventions by combining strong fail-safe maneuvers with a fallback controller;
- we guarantee safety for braking limit changes in consensus techniques.

The rest of the paper is organized as follows: After describing the considered system in Sec. II, we provide an in-depth safety specification for platooning in Sec. III, which allows us to formulate our problem statement. Sec. IV presents our overall safety protocol, and Sec. V describes the safety protocol for consensus techniques. We evaluate our concept

in Sec. VI, and conclude our work with a discussion in Sec. VII.

II. PRELIMINARIES

A. System Dynamics

We address platooning on interstates with unidirectional driving and bounded road curvature [19]. Since the vehicles follow lanes with restricted curvature, we only consider longitudinal dynamics. To reflect the vehicle ordering within a specific lane, we number the vehicles in ascending order, i.e., vehicle i follows vehicle $i+1$. We write $\square^{(i)}$ to refer to the variable $\square$ of vehicle i, and denote a vehicle state at time t by $x(t) = [\, s(t) \quad v(t) \,]^\top$, where s is the front position along the road and $v \in [0, v_{\max}]$ the velocity, with $v_{\max}$ being either the physical or legal maximum velocity. The desired acceleration a_d is the input of each vehicle, and we use a for the actual acceleration. We denote the mass of a vehicle as m, the drag coefficient as c, the frontal area as A, and the length as l. Let us also introduce the air density ρ, the headwind velocity v_{wind}, and the gravity g. The road incline angle at position s along the road is $\alpha(s) \in [\underline{\alpha}, \overline{\alpha}]$, where $\alpha(s) > 0$ represents an ascent. Combining the braking capability $\underline{a}$ due to tires and brakes, the drag $a_{\text{drag}}(v) = -\frac{1}{2m}\rho c A (v + v_{\text{wind}})^2$, and the acceleration $a_{\text{incline}}(\alpha) = -g\sin(\alpha)$ caused by an incline, the physical braking limit is [15, Sec. II.A]

$$a_{\text{phys}}(\alpha, v) = \underline{a} + a_{\text{incline}}(\alpha) + a_{\text{drag}}(v). \tag{1}$$

The clearing time Δt_C specifies the time within which a vehicle needs to recapture the safe distance after a cut-in by another vehicle [15], [20]. We use the predicate $\texttt{cut-in}(i)$ to indicate that vehicle i performed a cut-in within the last Δt_C. During Δt_C, we assume that the cut-in vehicle does not brake harder than $a_{\text{cut-in}}$, where this assumption is adjusted as soon as the cut-in vehicle brakes harder. With that, the overall deceleration limit of vehicle i is

$$a_{\min}^{(i)}(\alpha, v) = \begin{cases} \max(a_{\text{cut-in}}, a_{\text{phys}}(\alpha, v)) & \text{if } \texttt{cut-in}(i) \\ a_{\text{phys}}(\alpha, v) & \text{otherwise.} \end{cases} \tag{2}$$

We introduce the disturbance $w \in [\underline{w}, \overline{w}]$, where the disturbance set $[\underline{w}, \overline{w}]$ can be used to compensate model inaccuracies using reachset conformance [21]. Given the maximum possible acceleration $a_{\max}(s, v)$ due to engine characteristics, drag, and incline, the vehicle dynamics can be written as [15, Eq. (1)]

$$\dot{s} = v$$
$$\dot{v} = \begin{cases} 0 & \text{if } (v \le 0 \wedge a_d + w \le 0) \\ & \vee (v \ge v_{\max} \wedge a_d + w \ge 0) \\ a_{\min}(\alpha(s), v) + w & \text{if } a_d < a_{\min}(\alpha(s), v) \\ a_{\max}(\alpha(s), v) + w & \text{if } a_d > a_{\max}(\alpha(s), v) \\ a_d + w & \text{otherwise.} \end{cases} \tag{3}$$

For an initial state x_0, an input trajectory $a_d(\cdot)$, and a disturbance trajectory $w(\cdot)$, the solution of the model in (3) over time t is denoted as $\boldsymbol{\xi}(t; x_0, a_d(\cdot), w(\cdot))$. We write $\boldsymbol{\xi}_s$ to refer to the position of $\boldsymbol{\xi}$. We allow the input $a_d = -\infty$, which results in full braking according to (3).

B. Platoon Vehicle Assumptions

In addition to measuring its state variables, each vehicle in the platoon obtains the relative position and velocity of each preceding vehicle within a sensor range of at least s_{sensor}. To account for measurement uncertainties, we assume that a measurement results in an interval $[\underline{\square}, \overline{\square}]$ enclosing the actual value of variable $\square$ [15]. Measurement intervals are also provided to each platoon vehicle for $\alpha(s)$, ρ, and v_{wind}.

Furthermore, we assume that each platoon vehicle i is equipped with an arbitrary CACC, denoted as the *nominal* CACC of vehicle i. It operates with a planning period of $\Delta t_{\text{p}}^{(i)}$, i.e., it provides a new input a_d at discrete planning times $t_k = k\Delta t_{\text{p}}^{(i)}$ to be applied in $[t_k, t_{k+1})$.

III. SAFETY SPECIFICATION AND PROBLEM STATEMENT

In this section, we introduce a comprehensive notion of safety in platooning, and subsequently formulate our problem statement.

A. Safety Specification

First, we introduce relevant predicates and a function referring to vehicle i at time t. For brevity, we omit the dependence on i and t in the notation.

- $\texttt{emg}$: Vehicle i brakes as strong as possible at time t. To prevent potential collisions within the platoon, the vehicle communicates a possible imminent collision backward within the time interval $[t, t + \Delta t_{\text{p}}^{(i)}]$, containing its predicted rear position at the time of the collision.
- $\texttt{safe}(j)$: Vehicle i is safe w.r.t. vehicle $j > i$ at time t, formally defined by

$$\begin{aligned} \texttt{safe}(j) \iff \exists a^{(i)}(\cdot)\, \forall w^{(i)}(\cdot)\, \forall a^{(j)}(\cdot)\, \forall w^{(j)}(\cdot)\, \forall t' \ge 0 \colon \\ \boldsymbol{\xi}_s^{(i)}(t'; x^{(i)}(t), a^{(i)}(\cdot), w^{(i)}(\cdot)) \\ < \boldsymbol{\xi}_s^{(j)}(t'; x^{(j)}(t), a^{(j)}(\cdot), w^{(j)}(\cdot)) - l^{(j)}. \end{aligned} \tag{4}$$

- $\texttt{stop-before}(s)$: Vehicle i can stop before position s. For a standing vehicle j with rear position s, the predicate is defined by $\texttt{stop-before}(s) \iff \texttt{safe}(j)$.
- $\mathsf{coll}(j)$: The function returns the collision position that vehicle i received by vehicle j as part of a collision alert before time t. If vehicle i did not receive a collision alert or if it was withdrawn by vehicle j, ∞ is returned.

Using first-order logic, we now provide formal specifications to guarantee legal safety in platooning [12]. We use the symbol $\veebar$ to denote the exclusive disjunction operation.

Specification III.1 (Stopping within Sensor Range [15]): Vehicle i is always able to stop within its sensor range:

$$\forall t \colon \texttt{stop-before}(s^{(i)}(t) + s_{\text{sensor}}). \tag{5}$$

Specification III.2 (Collision Avoidance): Vehicle i has to keep a safe distance unless another vehicle performed a cut-in, triggering the emergency procedure:

$$\forall t\, \forall j > i \colon \texttt{safe}(j) \veebar \texttt{emg}. \tag{6}$$

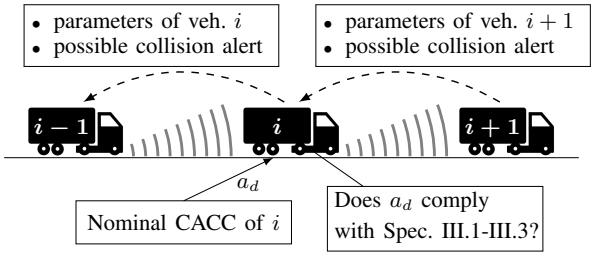

Fig. 1. Overview of our safety protocol without consensus for vehicle i, including communication (dashed).

Specification III.3 (Multiple Collision Avoidance): When receiving a collision alert, vehicle i must stop before the collision or trigger the emergency procedure:

$$\forall t\, \forall j > i\colon \texttt{stop-before}(\mathsf{coll}(j)) \veebar \texttt{emg}. \tag{7}$$

Based on these safety specifications, we formulate the problem statement next.

B. Problem Statement

The objective of this work is to develop a safety protocol applicable to CACC and consensus techniques in platooning, which ensures that each vehicle of the platoon always fulfills the specifications III.1-III.3, while 1) still realizing small inter-vehicle distances, 2) intervening with the nominal CACC as rarely and smoothly as possible, 3) handling communication failures, and 4) ensuring convergence of the consensus techniques.

IV. SAFETY PROTOCOL

This section describes the overall safety protocol; the protocol for consensus techniques is explained later. During platoon formation, adjacent vehicles $i-1$ and i perform a handshake and mutually confirm that they apply the safety protocol. Furthermore, vehicle i communicates its parameters to vehicle $i-1$. We use the predicate $\texttt{coupled}(i-1, i)$ to denote that vehicles $i-1$ and i successfully coupled.

Fig. 1 provides an overview of the protocol: Every vehicle i verifies compliance with the specifications III.1-III.3 in each planning step. The planned input a_d is only applied if it is compliant. Otherwise, a fallback controller is engaged. In some cases, vehicle i must communicate an imminent collision with its succeeding vehicles.

A. Solution Concept

We describe how a vehicle i verifies if its planned input a_d complies with Spec. III.1-III.3. Fig. 2 sketches the approach, which extends the concept in [15] by collision alerts: Vehicle i simulates its own state forward in time for the time $\Delta t_\mathrm{p}^{(i)}$, followed by a full brake as a fail-safe trajectory until standstill at time t_stop. It also simulates each preceding vehicle $j > i$ until t_stop, assuming that vehicle j immediately performs a full brake. Here, vehicle i assumes worst-case parameters for vehicle j. Only for vehicle $i+1$, vehicle i utilizes the parameters received via communication (cf. Fig. 1) to enable a smaller safe distance. The input a_d is classified as safe if vehicle i always stays behind a) its sensor range (cf. Spec. III.1), b) each vehicle j (cf. Spec. III.2), and c) the received collision positions (cf. Spec. III.3). If the verification fails in the next planning cycle, vehicle i can still safely apply full braking from t_{k+1} on. This verification procedure guarantees compliance with Spec. III.1-III.3, which follows by induction over the planning steps assuming that $v^{(i)}(t_0) = 0$ [15, Sec. IV.A)].

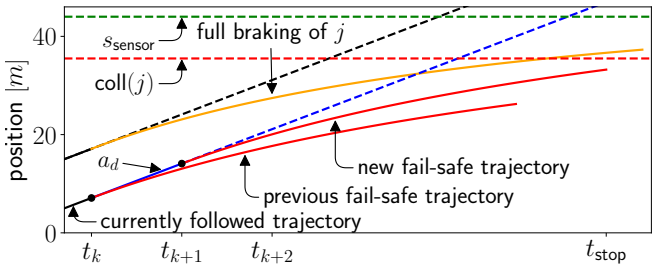

Fig. 2. Vehicle i successfully verifies an input a_d in the k-th planning step w.r.t. a preceding vehicle j. The dashed blue and black lines represent possible future trajectories of vehicle i and j, respectively.

Algorithm 1 VERIFY$(a_d, \mathcal{X})$

Input: Input a_d, preceding vehicles $\mathcal{X} = \{i+1, \dots, N\}$
Output: Boolean indicating if vehicle i can safely apply a_d

1: $a_{d,\mathrm{ego}}(t) = \begin{cases} a_d & \text{if } t \le \Delta t_\mathrm{p}^{(i)} \\ -\infty & \text{otherwise.} \end{cases}$ ▷ simulated traj. for vehicle i
2: $a_{d,\mathrm{prec}}(t) = -\infty$ ▷ simulated traj. for preceding vehicles
3: stop-in-sensor-range $\leftarrow \overline{\mathrm{POS}}^{(i)}(a_{d,\mathrm{ego}}(\cdot), t_\mathrm{stop}) < s_\mathrm{sensor}$
4: no-coll $\leftarrow \forall j \in \mathcal{X}\, \forall r \in \{\, r \in \mathbb{N}_0 \mid (r-1)\Delta t_\mathrm{sim} \le t_\mathrm{stop} \,\}\colon$
$\overline{\mathrm{POS}}^{(i)}(a_{d,\mathrm{ego}}(\cdot), (r+1)\Delta t_\mathrm{sim}) < \underline{\mathrm{POS}}^{(j)}(a_{d,\mathrm{prec}}(\cdot), r\Delta t_\mathrm{sim})$
5: no-multi-coll $\leftarrow \forall j \in \mathcal{X}\colon \overline{\mathrm{POS}}^{(i)}(a_{d,\mathrm{ego}}(\cdot), t_\mathrm{stop}) < \mathsf{coll}(j)$
6: **return** stop-in-sensor-range $\wedge$ no-coll $\wedge$ no-multi-coll

Note that the simulation cannot be computed exactly due to measurement uncertainties, unknown future disturbances, and a missing closed-form solution of $\boldsymbol{\xi}_s$. To still guarantee safety, we compute upper bounds $\overline{\mathrm{POS}}^{(i)}(a_d(\cdot), t)$ for the front position of vehicle i for future time points t, and lower bounds $\underline{\mathrm{POS}}^{(j)}(a_d(\cdot), t)$ for the rear positions of the preceding vehicles, as detailed in Sec. IV-C.

Alg. 1 summarizes the described verification procedure. The time discretization Δt_sim for the forward simulation can be chosen by the user (l. 4). Note that during a time interval $[r\Delta t_\mathrm{sim}, (r+1)\Delta t_\mathrm{sim}]$, we use the left limit $r\Delta t_\mathrm{sim}$ for the preceding vehicles and the right limit $(r+1)\Delta t_\mathrm{sim}$ for vehicle i (l. 4) to guarantee safety over each consecutive time interval.

B. Overall Protocol

The overall safety protocol for a single planning step is summarized in Alg. 2. If vehicles i and $i+1$ are coupled, then vehicle i can safely ignore all vehicles in front of vehicle $i+1$ (l. 1), which holds by induction over the preceding platoon vehicles. We then invoke the protocol for consensus techniques (l. 2, cf. Sec. V). To ensure an appropriate reaction to cut-in vehicles, we integrate an approach similar to [15, Sec. V.B.2)] (l. 3). Both the SAFECONSENSUS and CUT-INHANDLING subroutine return an upper bound for the acceleration. Thus, the desired acceleration is the nominal

Algorithm 2 SAFETYPROTOCOL($\mathcal{X}$)

Input: Preceding vehicles $\mathcal{X} = \{i+1, \dots, N\}$
Output: Input acceleration a_d for vehicle i in $[t_k, t_{k+1})$

1: **if** coupled$(i, i+1)$: $\mathcal{X} \leftarrow \{i+1\}$ ▷ vehicle elimination
2: $a_0 \leftarrow$ SAFECONSENSUS($\mathcal{X}$) ▷ Alg. 3 in Sec. V
3: $a_1 \leftarrow$ CUT-INHANDLING$(i, \mathcal{X})$ ▷ cf. [15, Sec. V.B.2)]
4: $a_2 \leftarrow$ NOMINALCACC($\mathcal{X}$)
5: $a_d \leftarrow \min(\{a_0, a_1, a_2\})$
6: desired-safe $\leftarrow$ VERIFY$(a_d, \mathcal{X})$ ▷ Alg. 1 in Sec. IV-A
7: **if** ¬desired-safe:
8: $(a_d,$ fallback-safe$) \leftarrow$ FALLBACK$(i, \mathcal{X})$
9: **if** ¬fallback-safe: SENDALERT(s_{coll})
10: **if** desired-safe ∨ fallback-safe: WITHDRAWALERT()
11: **return** a_d

input (l. 4) limited by these two values (l. 5). We verify safety of the desired acceleration (l. 6) (cf. Sec. IV-A). To avoid full braking whenever possible, we additionally engage a fallback controller in case the desired acceleration is unsafe, which computes the maximum acceleration that is still safe using binary search (l. 8). This is a simple yet effective approach, as it can be run anytime-like in parallel to the nominal CACC by continuously improving the solution accuracy. In case of a cut-in or a received collision alert, even the previous fail-safe maneuver can become unsafe. The fallback controller then returns $a_d = -\infty$ and fallback-safe = false, and the imminent collision is communicated with the predicted position of collision s_{coll} (l. 9). Otherwise, collision alerts sent previously are withdrawn (l. 10).

C. Computing Bounds on Reachable Positions

We now present the computation of bounds on the reachable positions of a vehicle, denoted by the functions $\underline{\text{POS}}^{(j)}(a_d(\cdot), t)$ and $\overline{\text{POS}}^{(i)}(a_d(\cdot), t)$. We utilize a specific type of monotonicity for this; the dynamics of our vehicle model in (3) is not monotone in a classical sense [22], because the change in velocity is position-dependent due to the road incline. However, we show that under the assumption (8) on $a_d(\cdot)$, monotonicity in the position domain holds.

Theorem 1 (Monotonicity in the Position Domain): For any input trajectories $\underline{a}_d(\cdot)$, $a_d(\cdot)$, and $\overline{a}_d(\cdot)$ fulfilling

$$a_d(\cdot) \text{ is non-increasing over } t, \text{ and} \tag{8}$$

$$\forall t\colon \underline{a}_d(t) \le a_d(t) \le \overline{a}_d(t), \tag{9}$$

it holds that

$$\begin{aligned} \forall t \ge 0\colon \boldsymbol{\xi}_s(t; \underline{x}, \underline{a}_d(\cdot), \underline{w}) \\ \le \boldsymbol{\xi}_s(t; x, a_d(\cdot), w(\cdot)) \\ \le \boldsymbol{\xi}_s(t; \overline{x}, \overline{a}_d(\cdot), \overline{w}). \end{aligned} \tag{10}$$

Proof: We only show the first inequality in (10), as the proof for the second one works analogously. For brevity, we write $\underline{s}(t)$ and $\underline{v}(t)$ for the position and velocity of $\boldsymbol{\xi}(t; \underline{x}, \underline{a}_d(\cdot), \underline{w})$, and $s(t)$ and $v(t)$ analogously for $\boldsymbol{\xi}(t; x, a_d(\cdot), w(\cdot))$. First, we show that for any $t_a \le t_b$ with $\underline{s}(t_b) = s(t_a)$ and $\underline{v}(t_b) = v(t_a)$, it holds that

$$\dot{\underline{v}}(t_b) \le \dot{v}(t_a). \tag{11}$$

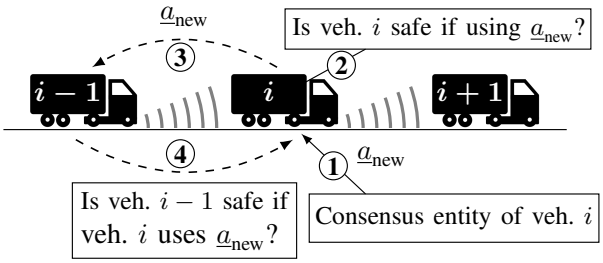

Fig. 3. Sketch of safely updating braking limits from the perspective of vehicle i, including communication (dashed).

From (8) and (9) follows that $\underline{a}_d(t_b) \le a_d(t_a)$, which, together with $\underline{w} \le w(t_a)$, results in (11) according to the dynamics in (3).

Let us assume that the first inequality in (10) does not hold for the sake of contradiction. The continuity of s in t implies that there is a minimum $t' > 0$, such that $\underline{s}(t') = s(t')$ and $\exists \epsilon > 0\ \forall \delta \in (0, \epsilon]\colon \underline{s}(t'+\delta) > s(t'+\delta)$. Thus

$$\underline{v}(t') > v(t'), \tag{12}$$

as $\underline{v}(t') = v(t')$ implies that $\dot{\underline{v}}(t') > \dot{v}(t')$, contradicting (11). From (12), $\underline{x} \le x$, and the continuity of v in s follows that there must be a position s', such that there are t_a, t_b with $t_a \le t_b < t'$ and $\underline{s}(t_b) = s(t_a) = s'$, $\underline{v}(t_b) = v(t_a)$ and $\exists \epsilon > 0\, \forall \delta \in (0, \epsilon]\colon \underline{v}(t_b + \delta) > v(t_a + \delta)$. This implies that $\dot{\underline{v}}(t_b) > \dot{v}(t_a)$, which contradicts (11), thus, the assumption on t' must be wrong, proving the theorem. □

For a time step size $\Delta t_{\text{step}} \ge 0$ and an input trajectory $a_d(\cdot)$ fulfilling (8), we choose $\underline{a}_d(t) = a(t + \Delta t_{\text{step}})$, and $\overline{a}_d(\cdot)$ is the zero-order hold function of $a_d(\cdot)$. Theorem 1 allows us to compute $\underline{\text{POS}}^{(j)}(a_d(\cdot), t)$ and $\overline{\text{POS}}^{(i)}(a_d(\cdot), t)$ using $\underline{a}_d(\cdot)$ and $\overline{a}_d(\cdot)$, respectively, with standard solvers for ordinary differential equations (see Appendix for further details).

V. SAFETY PROTOCOL FOR CONSENSUS TECHNIQUES

We additionally consider the possibility that all vehicles in the platoon use a consensus scheme to establish a common braking limit $\underline{a}_{\text{cons}}$ that is not known a priori; this value obviously changes when vehicles enter or leave the platoon. We assume that the consensus scheme runs black-box entities on all vehicles in the platoon, denoted as *consensus entities*, that keep proposing new braking limits for their vehicle, which gradually converge to $\underline{a}_{\text{cons}}$. Both decentralized and centralized consensus schemes can be represented in this way, as well as schemes that establish a consensus braking limit successively and those that only require a single change.

Our protocol in Fig. 3 for safely changing the braking limit using consensus techniques is embedded in the overall safety protocol as the SAFECONSENSUS procedure (cf. l. 2 in Alg. 2). For each new braking limit $\underline{a}_{\text{new}}$ provided by its consensus entity (step 1 in Fig. 3), vehicle i verifies if safety is still upheld (step 2). If the safety of the succeeding vehicle is affected by the new braking limit, vehicle i requests a safety confirmation by vehicle $i-1$ for $\underline{a}_{\text{new}}$ (steps 3-4). Only if safety is ensured, vehicle i adopts $\underline{a}_{\text{new}}$ as the new braking limit.

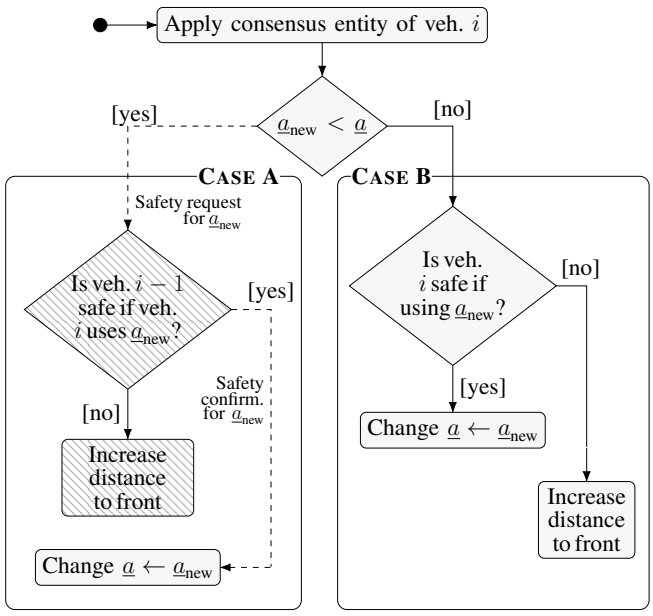

Fig. 4. Activity diagram of the protocol for consensus techniques from the perspective of vehicle i, including communication (dashed) and activities carried out by vehicle $i-1$ (hatched).

A. Overview

We present an overview of the SAFECONSENSUS procedure in Fig. 4 using an activity diagram: The consensus entity of vehicle i provides a new braking limit $\underline{a}_{\text{new}}$ in the k-th planning step. Vehicle i distinguishes two cases:

a) Case A ($\underline{a}_{new} < \underline{a}$): Safety of vehicle i w.r.t. the preceding vehicles would not be changed by $\underline{a}_{\text{new}}$, but safety of the coupled succeeding vehicle $i-1$ due to stronger braking capabilities of the preceding vehicle. Therefore, vehicle i requests a safety confirmation for $\underline{a}_{\text{new}}$ from vehicle $i-1$. Vehicle $i-1$ receives the confirmation request and re-verifies its safety, assuming that vehicle i was using $\underline{a}_{\text{new}}$ instead of $\underline{a}$. If safety is not provided, vehicle $i-1$ starts to increase the distance to vehicle i. As soon as safety is ensured, vehicle $i-1$ sends a confirmation to vehicle i so that vehicle i can safely change its currently used braking limit to $\underline{a}_{\text{new}}$.

b) Case B ($\underline{a}_{new} \geq \underline{a}$): An increased braking limit requires vehicle i to re-verify its safety w.r.t. the preceding vehicles. As soon as the verification is successful, it can change its braking limit to $\underline{a}_{\text{new}}$. Otherwise, it starts to increase the distance to the front, such that $\underline{a}_{\text{new}}$ can be adopted at some point in the future. By this, convergence of the consensus scheme is ensured.

B. Protocol in Detail

Alg. 3 shows the complete SAFECONSENSUS protocol. Let us now examine crucial steps of Alg. 3.

a) Apply consensus scheme: Apart from sending a safety request in case A, vehicle i also needs to send its currently used braking limit $\underline{a}$ to the succeeding vehicle in case B (ll. 9, 11-12). If the braking limit sent to vehicle $i-1$ is larger than in the previous planning step, vehicle i discards

Algorithm 3 SAFECONSENSUS($\mathcal{X}$)

Persistent variables:

- $\underline{a}$: currently used braking limit of vehicle i, initially set to the physical limit.
- $\underline{a}^{(\text{prec})}$: braking limit for vehicle $i+1$, initially chosen based on worst-case assumptions.
- $\underline{a}_{\text{comm}}$: latest communicated braking limit of vehicle i, initially set to $-\infty$.

Input: Received time-labeled data:

- From vehicle $i-1$ (sent in l. 19): $\underline{a}_{\text{conf}}$ (braking limit of vehicle i for which vehicle $i-1$ confirms its safety). If no message has been received: $\underline{a}_{\text{conf}} \leftarrow \infty$.
- From vehicle $i+1$ (sent in l. 12): $\underline{a}_{\text{comm}}^{(\text{prec})}$ (new braking limit that vehicle $i+1$ either requests a safety confirmation for or already adopted safely).

Output: Upper bound a_{trans} on input acceleration

```
1: increase-dist ← false
2: ▷ Apply consensus scheme:
3: a_new ← CONSENSUSENTITY()
4: if a_new ≥ a:                                          ▷ Case B
5:   if VERIFY(−∞, X) using a_new for a:
6:     a ← a_new
7:   else
8:     increase-dist ← true
9:     a_new ← a
10: if a_new > a_comm:  DISCARDPREVIOUSMESSAGES()
11: a_comm ← a_new
12: SENDNEWLIMITTOSUCCEEDINGVEH(a_comm)
13: ▷ Process incoming safety request: Case A of vehicle i+1
14: safe← true
15: if a_comm^(prec) < a^(prec):
16:   safe ← VERIFY(−∞, {i+1}) using a_comm^(prec) for a^(prec)
17: if safe:  a^(prec) ← a_comm^(prec)
18: else: increase-dist ← true
19: SENDCONFIRMATIONTOPRECEDINGVEH(a^(prec))
20: ▷ Process received safety confirmation:
21: if a_conf ≤ a:
22:   a ← max({a_conf, a_comm})
23: return GETINPUTBOUND(increase-dist)
```

all messages sent so far (l. 10). The following proposition, which is used in the proof of Lemma 3, is thus fulfilled:

Proposition 1: If at times $t' < t$, vehicle i sent the braking limits $\underline{a}_{\text{comm}}\langle t', i\rangle < \underline{a}_{\text{comm}}\langle t, i\rangle$, respectively, in line 12, a confirmation for the braking limit $\underline{a}_{\text{comm}}\langle t', i\rangle$ received at time t is discarded.

b) Process incoming safety request: If $\underline{a}_{\text{comm}}^{(\text{prec})} < \underline{a}^{(\text{prec})}$ (l. 15), vehicle i knows that vehicle $i+1$ requests a confirmation for $\underline{a}_{\text{comm}}^{(\text{prec})}$, and re-verifies its safety. If safety is provided, vehicle i updates the stored braking limit $\underline{a}^{(\text{prec})}$ for vehicle $i+1$ (l. 17). It always sends a confirmation for the currently stored braking limit $\underline{a}^{(\text{prec})}$ (l. 19), which is either the currently requested one, or was requested previously. By that, we account for lost confirmations in case communication fails.

c) Process received safety confirmation: As a confirmation by the succeeding vehicle is only required if $\underline{a}_{\text{new}} < \underline{a}$ (cf. case A in Fig. 4), a confirmation for $\underline{a}_{\text{conf}} > \underline{a}$ is obviously outdated and thus ignored (l. 21). The adopted braking limit is the maximum of the confirmed braking limit

and the currently communicated one (l. 22). By that, the last braking limit provided by the consensus entity is not undercut.

d) Increasing distance to front: A vehicle executing a safety verification unsuccessfully must start to increase the distance to the preceding vehicle. This is indicated by the variable increase-dist, which is always disabled at the beginning of Alg. 3 (l. 1), and possibly enabled later (ll. 8 and 18). If increase-dist is true, the function GETINPUTBOUND (l. 23) returns a successively decreasing acceleration starting from the currently applied one $a(t_{k-1})$, which is used as an upper bound for the input acceleration (cf. l. 5 in Alg. 2). If increase-dist is false, GETINPUTBOUND returns ∞.

C. Safety Proof

This section proves that Alg. 3 ensures safe braking limit changes. We explicitly assume that the vehicles of the platoon execute Alg. 3 concurrently, and we take communication failures and delays into account by not assuming that messages always and immediately arrive or arrive in the order sent. By providing each message with a timestamp, the vehicles can filter out obsolete messages as noted in the following Remark:

Remark 1: A vehicle always discards a message received by another vehicle if it received a more recent message by that vehicle before.

Note that an unexpected termination of an execution of Alg. 3 at any point does not impede safety. To achieve this, we never assume in the proofs below that the execution of one line in Alg. 3 implies the execution of a subsequent one. Throughout this section, we write $\square\langle t, i\rangle$ to refer to the value of variable $\square$ at time t during the execution of Alg. 3 by vehicle i. We make use of the following lemmas:

Lemma 1: It always holds that $\underline{a}_{\text{comm}} \leq \underline{a}$.

Proof: We only change $\underline{a}_{\text{comm}}$ in line 11, where the new value is $\underline{a}_{\text{new}}$. In case the *if*-block in line 4 was not executed, it holds that $\underline{a}_{\text{new}} < \underline{a}$. In case it was executed, $\underline{a}_{\text{new}} = \underline{a}$ holds afterwards. $\square$

Lemma 2: $\underline{a}$ is never decreased in line 6 and never increased in line 22.

Proof: The former case is implied by the condition in line 4. The latter case follows from $\underline{a}_{\text{conf}} \leq \underline{a}$ (l. 21) and Lemma 1. $\square$

Lemma 3: At any time t, the braking limit that vehicle $i-1$ assumes for its coupled preceding vehicle i is an underestimation:

$$\underline{a}^{(\text{prec})}\langle t, i-1\rangle \leq \underline{a}\langle t, i\rangle. \tag{13}$$

Proof: see Appendix.

Lemma 4: If a vehicle is safe w.r.t. each preceding vehicle and changes $\underline{a}$, it is also safe afterwards.

Proof: Line 5 implies safety for the change in line 6, and Lemma 2 for the one in line 22. $\square$

Lemma 5: If a vehicle is safe w.r.t. its preceding vehicle applying $\underline{a}^{(\text{prec})}$, and the vehicle changes $\underline{a}^{(\text{prec})}$, it is also safe if the preceding vehicle uses the new value of $\underline{a}^{(\text{prec})}$.

TABLE I
COMMON PARAMETERS

Parameter	Value	Parameter	Value
$[\underline{\rho}, \overline{\rho}]$	$[1.1, 1.3]\text{kg/m}^3$	Δt_{p}	$0.1\,\text{s}$
$[\underline{v}_{\text{wind}}, \overline{v}_{\text{wind}}]$	$[1.4, 4.2]\text{m/s}$	a_{tol}	$0.05\,\text{m/s}^2$
$[\underline{\alpha}, \overline{\alpha}]$	$[-0.06, 0.06]\text{rad}$	$a_{\text{cut-in}}$	$1\,\text{m/s}^2$
$[\underline{w}, \overline{w}]$	$[-0.1, 0.1]\ \text{m/s}^2$ [21]	t_C	$4\,\text{s}$
s_{sensor}	$200\,\text{m}$		

Proof: A vehicle can change $\underline{a}^{(\text{prec})}$ only in line 17, where for the new value $\underline{a}^{(\text{prec})}_{\text{comm}}$ it either holds that $\underline{a}^{(\text{prec})}_{\text{comm}} \geq \underline{a}^{(\text{prec})}$, or safety verification was done in line 16. $\square$

We can now prove the safety of the protocol.

Theorem 2 (Braking Limit Changes are Safe): If vehicle i changes $\underline{a}$ in Alg. 3, safety of 1) vehicle i w.r.t. each vehicle $j > i$, and safety of 2) vehicle $i-1$ w.r.t. vehicle i is maintained.

Proof: Lemma 4 proves part 1). Part 2) holds as the braking limit that vehicle $i-1$ assumes about vehicle i is always underapproximate according to Lemma 3, and a change of $\underline{a}^{(\text{prec})}$ always preserves safety with Lemma 5.

VI. EVALUATION

We evaluate our concept on various scenarios using the CommonRoad platform [23]. We executed all simulations on a machine with an AMD Ryzen 9 5900HX processor with 4.6GHz and 64 GB of DDR4 3.200 MHz memory, and we implemented the safety protocol in Python. Tab. I shows the values of common parameters. We use a PD controller as nominal CACC inspired by [4] that maintains a headway $h = 0.3\,\text{s}$ to the preceding vehicle or controls a certain velocity. Furthermore, we use the consensus scheme proposed in [5], extended by a simple reset mechanism if a vehicle leaves the platoon to enable a decreasing convergence target for the braking limit. For the simulations, we generate measurement intervals and disturbances with a Gaussian distribution within the specified range, employing a 99% confidence interval and truncating values outside this range. We assume a measurement uncertainty range of $\pm[\,0.2\,\text{m} \quad 0.05\,\text{m/s}\,]^\top$ for the state of the ego vehicle, and $\pm[\,0.1\,\text{m} \quad 0.05\,\text{m/s}\,]^\top$ for the relative states of the preceding vehicles [24]. Tab. II shows the vehicle parametrizations used in the following scenarios. The runtime of a single planning step was consistently below $80\,\text{ms}$, highlighting the real-time capability of the approach. For each scenario, we show the occupancies relative to the first vehicle, and the position induced by the safe distances to the direct predecessor (dashed). We also plot the effective accelerations, already including disturbances; we use dotted lines to indicate that the fallback controller was active.

A. Scenario 1: Fallback Controller Evaluation

We consider a scenario with a platoon of two trucks parametrized by p_0 and p_1 (cf. Tab. II), where no consensus technique is used. The platoon is slowly approaching a non-platoon vehicle, which performs a full brake starting at $t = 30\,\text{s}$. The results are shown in Fig. 5. Vehicles 1 and 2 always keep a safe distance, and no collision occurs when vehicle 3 brakes fully. The nominal CACC frequently

TABLE II

VEHICLE PARAMETERS [25]

Parameter	Worst Case	p_0	p_1	p_2	p_3	p_4
$\underline{a}[\mathrm{m/s^2}]$	-12	-5	-6	-10	-5.5	-9
$\overline{a}[\mathrm{m/s^2}]$	—	1	1.5	4	1	3.5
$v_{\max}[\mathrm{m/s}]$	—	25	25	60	25	50
$m[\mathrm{t}]$	0.4	20	15	2.5	20	2
c	2	0.7	0.5	0.25	0.6	0.35
$A[\mathrm{m^2}]$	12.5	7	8	1.7	6	2.4
$l[\mathrm{m}]$	—	16	14	4.9	16	4.2

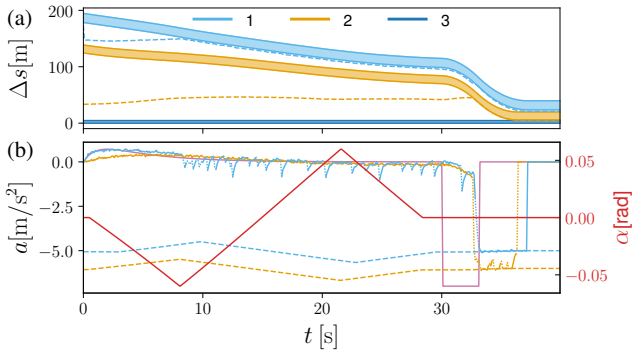

Fig. 5. (a) Occupancies. The safe distance drops after coupling at the very beginning. (b) Effective accelerations and $a_{\min}^{(i)}$ (dashed) for each platoon vehicle i, and road incline angle α (red) from the perspective of 2.

fails to compute a safe input for vehicle 1, so the fallback controller is engaged. The inputs computed by the latter are usually above $-1\,\mathrm{m/s^2}$, confirming that the fallback controller prevents unnecessarily harsh safety interventions.

B. Scenario 2: Consensus Techniques

We consider a scenario with a platoon of three trucks and two cars heterogeneously parametrized by p_0 to p_4 (cf. Tab. II). The scenario is executed with the consensus scheme. The results are shown in Fig. 6. A much denser spacing is achieved after the consensus scheme converges at $t \approx 5\,\mathrm{s}$. Vehicle 1, which has the weakest braking capability, leaves the platoon at $t = 29\,\mathrm{s}$ by a lane change, resulting in a reset of the consensus and a stronger consensus braking limit formed afterward. The full brake performed by the leader at $t = 80\,\mathrm{s}$ does not end in a collision. The safe distances are always kept despite the dense spacing.

VII. CONCLUSION

We propose a provably correct safety protocol for cooperative platooning that can be applied to any existing CACC. Considering changing road inclines, the safety of each nominal input is verified online against every possible acceleration behavior of the preceding vehicles, which is enabled by dynamics that is monotone in the position domain. If the nominal input is identified as unsafe, a fallback controller prevents harsh interventions. Impending collisions caused by vehicles cutting in are communicated backward to allow the succeeding vehicles to react proactively. For consensus techniques establishing a common braking limit among the platoon vehicles, we additionally propose a protocol allowing

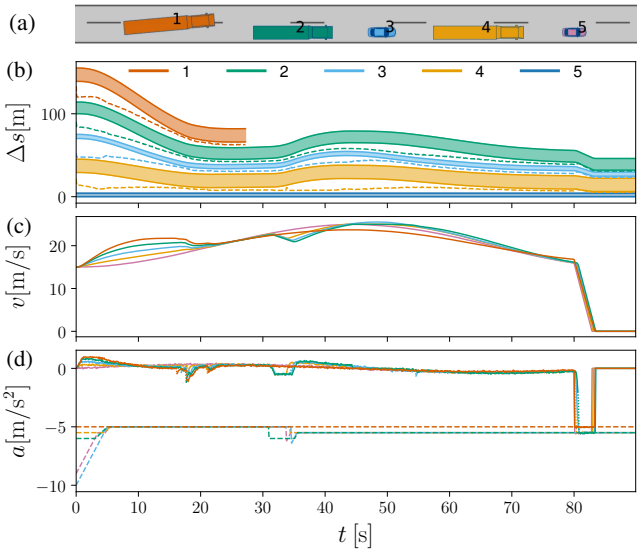

Fig. 6. (a) Illustration of CommonRoad scenario at $t = 26\,\mathrm{s}$. (b) Occupancies (c) Velocities (d) Effective accelerations and $\underline{a}^{(i)}$ (dashed). After vehicle 1 leaves the platoon, the consensus braking limit decreases.

the vehicles to safely change their braking limits. We confirm the benefit of our approach in our experiments.

ACKNOWLEDGMENT

The authors thank Adrian Kulmburg for his help with the proof of Theorem 1. Furthermore, the authors gratefully acknowledge the financial support by the European Commission Project justITSELF under grant number 817629.

REFERENCES

[1] J. Axelsson, "Safety in Vehicle Platooning: A Systematic Literature Review," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1033–1045, 2017.

[2] K. C. Dey, *et al.*, "A Review of Communication, Driver Characteristics, and Controls Aspects of Cooperative Adaptive Cruise Control (CACC)," *IEEE Transactions on Intelligent Transportation Systems*, vol. 17, no. 2, pp. 491–509, 2016.

[3] T. Tao, V. Jain, and S. Baldi, "An Adaptive Approach to Longitudinal Platooning with Heterogeneous Vehicle Saturations," *IFAC-PapersOnLine*, vol. 52, no. 3, pp. 7–12, 2019.

[4] S. Baldi, D. Liu, V. Jain, and W. Yu, "Establishing Platoons of Bidirectional Cooperative Vehicles With Engine Limits and Uncertain Dynamics," *IEEE Transactions on Intelligent Transportation Systems*, vol. 22, no. 5, pp. 2679–2691, 2021.

[5] D. Liu, S. Mair, K. Yang, S. Baldi, P. Frasca, and M. Althoff, "Resilience in Platoons of Cooperative Heterogeneous Vehicles: Self-organization Strategies and Provably-correct Design." [Online]. Available: https://arxiv.org/abs/2305.17443

[6] S. Thormann, A. Schirrer, and S. Jakubek, "Safe and Efficient Cooperative Platooning," *IEEE Transactions on Intelligent Transportation Systems*, vol. 23, no. 2, pp. 1368–1380, 2022.

[7] L. Alvarez and R. Horowitz, "Safe Platooning in Automated Highway Systems Part I: Safety Regions Design," *Vehicle System Dynamics*, vol. 32, no. 1, pp. 23–55, 1999.

[8] N. Mehdipour, M. Althoff, R. D. Tebbens, and C. Belta, "Formal Methods to Comply with Rules of the Road in Autonomous Driving: State of the Art and Grand Challenges," *Automatica*, vol. 152, 2023.

[9] A. Scheuer, O. Simonin, and F. Charpillet, "Safe Longitudinal Platoons of Vehicles without Communication," in *IEEE International Conference on Robotics and Automation*, 2009, pp. 70–75.

[10] A. Khalifa, O. Kermorgant, S. Dominguez, and P. Martinet, "Vehicles Platooning in Urban Environments: Integrated Consensus-based Longitudinal Control with Gap Closure Maneuvering and Collision Avoidance Capabilities," in *18th European Control Conference*, 2019, pp. 1695–1701.

[11] A. Alam, A. Gattami, K. H. Johansson, and C. J. Tomlin, "Guaranteeing Safety for Heavy Duty Vehicle Platooning: Safe Set Computations and Experimental Evaluations," *Control Engineering Practice*, vol. 24, pp. 33–41, 2014.
[12] M. Althoff and J. M. Dolan, "Online Verification of Automated Road Vehicles Using Reachability Analysis," *IEEE Transactions on Robotics*, vol. 30, no. 4, pp. 903–918, 2014.
[13] S. Magdici and M. Althoff, "Adaptive Cruise Control with Safety Guarantees for Autonomous Vehicles," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 5774–5781, 2017.
[14] J. Ligthart, E. Semsar-Kazerooni, J. Ploeg, M. Alirezaei, and H. Nijmeijer, "Controller Design for Cooperative Driving with Guaranteed Safe Behavior," in *IEEE Conference on Control Technology and Applications*, 2018, pp. 1460–1465.
[15] M. Althoff, S. Maierhofer, and C. Pek, "Provably-Correct and Comfortable Adaptive Cruise Control," *IEEE Transactions on Intelligent Vehicles*, vol. 6, no. 1, pp. 159–174, 2021.
[16] T. Willke, P. Tientrakool, and N. Maxemchuk, "A survey of inter-vehicle communication protocols and their applications," *IEEE Communications Surveys & Tutorials*, vol. 11, no. 2, pp. 3–20, 2009.
[17] Xue Yang, Jie Liu, Feng Zhao, and N. Vaidya, "A vehicle-to-vehicle communication protocol for cooperative collision warning," in *The First Annual International Conference on Mobile and Ubiquitous Systems: Networking and Services, 2004. MOBIQUITOUS 2004.* IEEE, 2004, pp. 114–123.
[18] D. Reichardt, M. Miglietta, L. Moretti, P. Morsink, and W. Schulz, "CarTALK 2000: safe and comfortable driving based upon inter-vehicle-communication," in *Intelligent Vehicle Symposium, 2002. IEEE*, vol. 2. IEEE, 2003, pp. 545–550.
[19] M. Irzik, K. Lemke, C. Lippold, L. Nink, M. Rohloff, and W. Wirth, *Guidelines for the Design of Motorways*. Road and Transportation Research Association, Cologne/Germany, 2008.
[20] S. Maierhofer, A.-K. Rettinger, E. C. Mayer, and M. Althoff, "Formalization of Interstate Traffic Rules in Temporal Logic," in *IEEE Intelligent Vehicles Symposium*, 2020, pp. 752–759.
[21] B. Schürmann, D. Heß, J. Eilbrecht, O. Stursberg, F. Koster, and M. Althoff, "Ensuring Drivability of Planned Motions Using Formal Methods," in *IEEE 20th International Conference on Intelligent Transportation Systems*, Yokohama, 2017, pp. 1–8.
[22] D. Angeli and E. Sontag, "Monotone Control Systems," *IEEE Transactions on Automatic Control*, vol. 48, 2003.
[23] M. Althoff, M. Koschi, and S. Manzinger, "CommonRoad: Composable Benchmarks for Motion Planning on Roads," in *IEEE Intelligent Vehicles Symposium*, 2017, pp. 719–726.
[24] F. Engels, P. Heidenreich, M. Wintermantel, L. Stäcker, M. Al Kadi, and A. M. Zoubir, "Automotive Radar Signal Processing: Research Directions and Practical Challenges," *IEEE Journal of Selected Topics in Signal Processing*, vol. 15, no. 4, pp. 865–878, June 2021.
[25] S. Pischinger and U. Seiffert, Eds., *Vieweg Handbuch Kraftfahrzeugtechnik*. Wiesbaden: Springer Fachmedien Wiesbaden, 2021.

APPENDIX

A. Details of Computing Bounds on Reachable Positions

When computing the position bounds $\underline{\text{POS}}^{(j)}(a_d(\cdot),t)$ and $\overline{\text{POS}}^{(i)}(a_d(\cdot),t)$, we need to evaluate $a_{\min}(\alpha,v)$ and $a_{\max}(\alpha,v)$ (cf. (2)) under- and overapproximatively, respectively. With the global acceleration limits

$$a_{\min,\text{glob}} = \underline{a} + a_{\text{incline}}(\overline{\alpha}) + a_{\text{drag}}(v_{\max}) \text{ and}$$
$$a_{\max,\text{glob}} = \overline{a} + a_{\text{incline}}(\underline{\alpha}),$$

we overapproximate the reachable position, velocity and incline intervals during the time Δt_{step} by

$$\mathcal{I}_s = [s, s + v\Delta t_{\text{step}} + 0.5 a_{\max,\text{glob}} \Delta t_{\text{step}}^2],$$
$$\mathcal{I}_v = [v + \Delta t_{\text{step}} a_{\min,\text{glob}}, v + \Delta t_{\text{step}} a_{\max,\text{glob}}], \text{ and}$$
$$\mathcal{I}_\alpha = [\min_{s' \in \mathcal{I}_s} \alpha(s'), \max_{s' \in \mathcal{I}_s} \alpha(s')].$$

We evaluate $a_{\min}(\alpha,v)$ and $a_{\max}(\alpha,v)$ on the right limits of $\mathcal{I}_\alpha$ and $\mathcal{I}_v$ for computing $\underline{\text{POS}}^{(j)}(a_d(\cdot),t)$, and on the left limits for computing $\overline{\text{POS}}^{(i)}(a_d(\cdot),t)$.

B. Proof of Lemma 3

Initially, (13) holds due to the worst-case assumption that vehicle $i-1$ makes about vehicle i. At time t, (13) can only change if vehicle i has changed $\underline{a}$ (l. 6 or 22), or vehicle $i-1$ has changed $\underline{a}^{(\text{prec})}$ (l. 17 in Alg. 3). Let us examine these three cases:

Vehicle i has changed $\underline{a}$ in line 6: This does not change (13) according to Lemma 2.

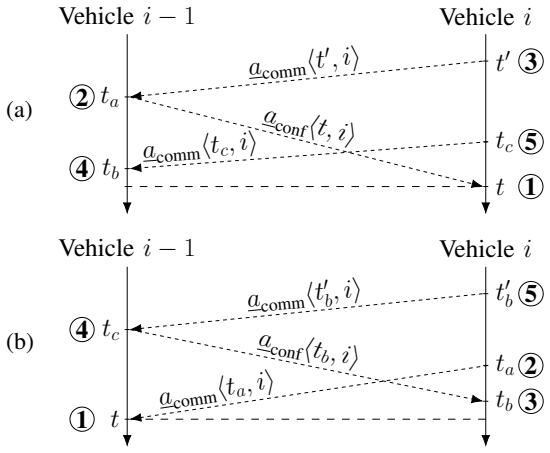

Fig. 7. Communication processes illustrating the proofs of Lemma 3 (a) Case 2, and (b) Case 3.

Vehicle i has changed $\underline{a}$ in line 22 (cf. ① in Fig. 7 (a)): Let $t_a \le t$ be the time at which vehicle $i-1$ sent the confirmation for $\underline{a}_{\text{conf}}\langle t, i\rangle$ (cf. ② in Fig. 7 (a)), and $t' \le t_a$ the last time that vehicle i communicated $\underline{a}_{\text{comm}}\langle t', i\rangle = \underline{a}_{\text{conf}}\langle t, i\rangle$ (cf. ③ in Fig. 7 (a)). Then, vehicle $i-1$ stores $\underline{a}_{\text{conf}}\langle t, i\rangle$ in its variable $\underline{a}^{(\text{prec})}$. As $\underline{a}_{\text{conf}}\langle t, i\rangle \le \underline{a}\langle t, i\rangle$ (cf. l. 22), (13) holds at time t, unless vehicle $i-1$ increased $\underline{a}^{(\text{prec})}$ in line 17 at some time $t_b \in [t_a, t]$, which we assume for the sake of contradiction (cf. ④ in Fig. 7 (a)). Then, vehicle i must have sent a braking limit at some time $t_c \in [t', t_b]$ with $\underline{a}_{\text{comm}}\langle t_c, i\rangle > \underline{a}_{\text{comm}}\langle t', i\rangle$ (cf. ⑤ in Fig. 7 (a)). However, Prop. 1 gives that vehicle i would not have accepted the confirmation received at time t then, thus our assumption was wrong.

Vehicle $i-1$ has changed $\underline{a}^{(\text{prec})}$ in line 17 (cf. ① in Fig. 7 (b)): Then vehicle i sent $\underline{a}_{\text{comm}}\langle t_a, i\rangle = \underline{a}^{(\text{prec})}\langle t, i-1\rangle$ in line 12 at some time $t_a \le t$ (cf. ② in Fig. 7 (b)). As $\underline{a}_{\text{comm}}\langle t_a, i\rangle \le \underline{a}\langle t_a, i\rangle$ due to Lemma 1, (13) holds at time t, unless vehicle i decreased $\underline{a}$ to a value below $\underline{a}_{\text{comm}}\langle t_a, i\rangle$ at some time $t_b \in [t_a, t]$ in line 22, which we assume for the sake of contradiction (cf. ③ in Fig. 7 (b)), i.e., $\underline{a}\langle t_b, i\rangle < \underline{a}_{\text{comm}}\langle t_a, i\rangle$. Let t_c the time that vehicle $i-1$ sent the confirmation for $\underline{a}_{\text{conf}}\langle t_b, i\rangle \le \underline{a}\langle t_b, i\rangle$ (cf. ④ in Fig. 7 (b)), and $t'_b \le t_c$ the last time that vehicle i sent $\underline{a}_{\text{comm}}\langle t'_b, i\rangle = \underline{a}_{\text{conf}}\langle t_b, i\rangle$ to vehicle $i-1$ (cf. ⑤ in Fig. 7 (b)). It holds that $t'_b < t_a$ with Remark 1. Now we have that vehicle i sent $\underline{a}_{\text{comm}}\langle t'_b, i\rangle < \underline{a}_{\text{comm}}\langle t_a, i\rangle$ at times $t'_b < t_a$, respectively. However, Prop. 1 gives that vehicle i would not have accepted the confirmation received at time t_b then, thus our assumption was wrong. □