Improving Urban Traffic Throughput With Vehicle Platooning: Theory and Experiments

STANLEY W. SMITH [1], (Graduate Student Member, IEEE), **YEOJUN KIM** [2,3], **JACOPO GUANETTI** [3],
RUOLIN LI [2], **ROYA FIROOZI** [2], (Graduate Student Member, IEEE), **BRUCE WOOTTON** [3],
ALEXANDER A. KURZHANSKIY [4], **FRANCESCO BORRELLI** [2,3],
ROBERTO HOROWITZ [2], (Senior Member, IEEE), AND **MURAT ARCAK** [1,2], (Fellow, IEEE)

[1] Department of Electrical Engineering and Computer Sciences, University of California at Berkeley, Berkeley, CA 94720, USA
[2] Department of Mechanical Engineering, University of California at Berkeley, Berkeley, CA 94720, USA
[3] Widesense
[4] California Partners for Advanced Transportation Technology, University of California at Berkeley, Berkeley, CA 94720, USA

Corresponding author: Stanley W. Smith (swsmth@eecs.berkeley.edu)

ABSTRACT In this paper we present a model-predictive control (MPC) based approach for vehicle platooning in an urban traffic setting. Our primary goal is to demonstrate that vehicle platooning has the potential to significantly increase throughput at intersections, which can create bottlenecks in the traffic flow. To do so, our approach relies on vehicle connectivity: vehicle-to-vehicle (V2V) and vehicle-to-infrastructure (V2I) communication. In particular, we introduce a customized V2V message set which features a velocity forecast, i.e. a prediction on the future velocity trajectory, which enables platooning vehicles to accurately maintain short following distances, thereby increasing throughput. Furthermore, V2I communication allows platoons to react immediately to changes in the state of nearby traffic lights, e.g. when the traffic phase becomes green, enabling additional gains in traffic efficiency. We present our design of the vehicle platooning system, and then evaluate performance by estimating the potential gains in terms of throughput using our results from simulation, as well as experiments conducted with real test vehicles on a closed track. Lastly, we briefly overview our demonstration of vehicle platooning on public roadways in Arcadia, CA.

INDEX TERMS Vehicle platooning, traffic throughput, model predictive control.

I. INTRODUCTION

Vehicle connectivity and autonomy are important areas of research, both of which have made a notable impact on the automotive industry [1]. For example, advanced driver assist systems (ADAS) which automate the longitudinal and lateral motion of the vehicle, such as the Tesla Autopilot and Cadillac Super Cruise systems, are being offered as an option in an increasing number of production vehicles. Furthermore, V2V communication technology is now included as a standard feature in Cadillac CTS sedans [2].

The advent of connected automated vehicles has also paved the way towards significant improvements in transportation broadly [3], including increased safety (by allowing, for

The associate editor coordinating the review of this manuscript and approving it for publication was Dong Shen.

example, the detection of vehicles occluded from sight) and reduced reliance on traffic lights [4]. V2V communication allows for nearby vehicles to coordinate their motion accurately and to form *vehicle platoons*: strings of vehicles driving at the same speed and at short distance. There are two primary benefits of vehicle platooning: an improvement in traffic efficiency due to increased roadway capacity, and an increase in fuel efficiency due to reduced aerodynamic drag forces acting on the platooning vehicles, especially for heavy-duty vehicles such as semi-trucks. Regarding the first point, there is demonstrated potential for platooning to increase the capacity of both highways and urban roadways. For example, a microscopic simulation study in [5] predicts that increasing the penetration of vehicles capable of cooperative adaptive cruise control (CACC) will result in an increase in highway capacity, since it enables the driver to select smaller time

FIGURE 1. Test vehicles at the Hyundai-KIA Motors California Proving Grounds, California City, CA.

headways. In [6] the authors predict that the throughput of urban roadways could potentially be doubled by forming platoons of vehicles, particularly by increasing the capacity of intersections, which they confirm with a subsequent simulation study. For the second point, experiments presented in [7] confirm that small spacings between two heavy-duty trucks results in reduced fuel consumption.

Previous demonstrations have showcased the technical feasibility of vehicle platooning. For example, vehicle platooning was demonstrated in 1994 and 1997 by the California PATH team on the I-18 highway in San Diego, CA [8]. Other experimental evaluations conducted on highways include [9], where the authors develop a platooning system architecture for heavy-duty vehicles. The system is evaluated in terms of controller tracking performance and fuel consumption over varying levels of road grade. In [10], [11] the authors present the design of a CACC system and tested it on a fleet of test vehicles. A primary controller performance metric in these works is *string stability* [12], meaning that the preceding vehicles are able to attenuate disturbances in traffic downstream (for example, changes in velocity). In 2011 the first Grand Cooperative Driving Challenge was held in the Netherlands [13], with the goal of accelerating the deployment of cooperative driving technologies. The competition focused on CACC and included both an urban and highway driving challenge [14]. For the urban driving challenge one criterion used to judge the participating teams was throughput improvement at the traffic light. This scenario is similar to the one we considered in our previous work [15], where we focused on the trade-off between traffic throughput gains and safety.

In addition to maintaining a platoon formation, the related tasks of forming, merging, and splitting platoons require structured coordination between vehicles, i.e. interaction protocols, which can be achieved in principle with V2V communication. For example, in [16] state machines are provided which describe the sequence of events, coordinated via V2V communication, that must occur during merge, split, and change lane maneuvers. Furthermore, low level control laws for the leader vehicle to execute these maneuvers have been developed [17]. In [18] an extended message set is proposed for the purpose of enabling connected vehicles to coordinate more complex maneuvers in merging, intersection, and emergency vehicle scenarios for a follow-up Grand Cooperative Driving Challenge which was held in 2016 [19]. Other works studying communication include [20], where the authors present a strategy for maintaining string stability in a vehicle platoon while using significantly fewer communication resources.

Unlike the aforementioned studies, in this work we focus on advancing vehicle platooning to a public urban environment where increased intersection throughput can result in significant improvements in overall traffic efficiency. Enabling platooning in an urban environment involves addressing various challenges, such as forming and disbanding platoons in moving traffic, decision-making (e.g. whether or not to proceed through an upcoming intersection), and ensuring safety when a lead vehicle is present. These challenges are especially important on a public roadway, where the future behavior of vehicles ahead of the platoon and the phase of upcoming traffic lights are uncertain. We present a design for the urban platooning system, and then analyze performance by estimating throughput using data obtained from simulations and experiments conducted on a closed track. We also introduce a state machine for managing the participating platooning vehicles, and propose strategies for the platoon to ensure safety when it encounters an intersection and / or a leading vehicle, utilizing predictions of their future behavior.

The closest comparable effort that we are aware of is the MAVEN project, which has laid out the various technologies that are needed to develop and deploy urban platooning, and reported on test results with two automated vehicles [21], where technologies such as a green light optimal speed advisory system and a collective perception message were utilized. Unlike [21], our focus in this paper is on improving throughput by maintaining short (constant) distances between the vehicles as the platoon accelerates from rest to a nominal speed. In particular, we achieve such accurate tracking by transmitting velocity forecasts between platooning vehicles and using them as disturbance previews in our MPC problems.

The remainder of the paper is organized as follows. We outline our design for the urban vehicle platooning system in Sections II - IV, including a platoon model and management system, MPC formulation, and strategy for the leader to ensure safety. In Section V we present results from our simulation tool, and analyze the performance of the platooning system by estimating the potential gains in intersection throughput. Next, in Section VI we discuss the experimental setup and present results from conducting tests on a closed track and on public roadways in Arcadia, CA, including our estimates of throughput. We end with concluding remarks in Section VII and discuss some of the challenges we encountered during the tests, as well as potential solutions. We note that parts of Sections II - IV are adapted from our previous work [15], but the remaining content in the paper is completely new and advances platooning to an urban setting.
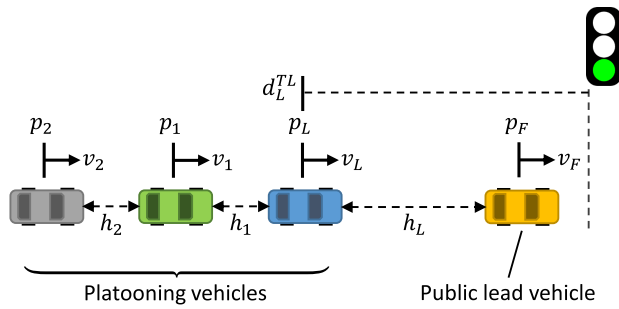
FIGURE 2. Depiction of the states for a platoon of size $N = 3$ and public lead vehicle approaching an upcoming traffic light.

II. PLATOON MODEL AND MANAGEMENT

In this section we introduce the model of the platoon and various systems that enable management of its behavior (beyond the control algorithms themselves), including state estimation via on-board sensors, V2X communication, and a finite-state machine (FSM) system which ensures that the platoon acts in a coordinated manner, that is, vehicles start moving as a single platoon at the same time and break the platoon at the same time as needed. In particular, we discuss how vehicle-to-vehicle communication enables the follower vehicles to do accurate distance tracking of the leader, and how vehicle-to-infrastructure communication enables the leader to decide whether or not to proceed through an upcoming intersection.

A. VEHICLE MODELS

The longitudinal dynamics of the leader vehicle [22] are modelled as

$$\dot{p}_L(t) = v_L(t), \tag{1a}$$

$$\dot{h}_L(t) = v_F(t) - v_L(t), \tag{1b}$$

$$\dot{d}_L^{TL}(t) = -v_L(t), \tag{1c}$$

$$\dot{v}_L(t) = \frac{1}{M}\left(\frac{T_L^a(t) - T_L^b(t)}{R_w} - F_f(t)\right), \tag{1d}$$

$$\dot{T}_L^a(t) = \frac{1}{\tau}\left(T_L^{a,ref}(t) - T_L^a(t)\right), \tag{1e}$$

where the states are as follows: $p_L(t)$ is the position, $h_L(t)$ is the distance to the public vehicle ahead (specifically, the distance from the front bumper of the leader vehicle to the rear bumper of the front vehicle), $d_L^{TL}(t)$ is the distance to the nearest upcoming intersection stop bar, $v_L(t)$ is the ego vehicle velocity, and $T_L^a(t) \in \mathbb{R}_{\geq 0}$ is the accelerating wheel torque. The inputs $T_L^{a,ref}(t) \in \mathbb{R}_{\geq 0}$ and $T_L^b(t) \in \mathbb{R}_{\geq 0}$ are the accelerating wheel torque command and the braking wheel torque. Lastly, $v_F(t)$ is the velocity of the public vehicle ahead, henceforth referred to as the front vehicle. The parameters M, R_w, and τ are the vehicle mass, wheel radius, and actuation time constant for acceleration, respectively. We note that (1e) models actuation delay while the vehicle is accelerating, which has been empirically estimated by collecting wheel torque measurements from the test vehicle. During these experiments we observed no delay while braking, and

TABLE 1. Model parameters.

M	vehicle mass	kg	2044
R_w	wheel radius	m	0.3074
β	frictional force modelling parameter	-	339.1329
γ	(same as above)	-	0.77
τ	accelerating torque actuation time constant	s	0.7868
Δt	sampling time	s	0.1

therefore the model does not include actuation delay while braking. Lastly, $F_L^f(t)$ is a longitudinal force acting on the leader vehicle, given by

$$F_L^f(t) = Mg\left((\sin(\theta) + r\cos(\theta)) + \frac{1}{2}\rho Ac_x v_L(t)^2\right) \tag{2}$$

where g is the gravitational constant, θ is road grade, A is the area of the vehicle, r is a rolling coefficient of the vehicle, ρ is air density, and c_x is an air drag coefficient. We assume road grade is negligible, and thus $\theta = 0$ for $t \geq 0$. For simplicity, we represent (2) as

$$F_L^f(t) = \beta + \gamma v_L(t)^2 \tag{3}$$

where the parameters $\beta, \gamma \in \mathbb{R}_{\geq 0}$ were identified by collecting driving data at a testing area near UC Berkeley, and then fitting predictions from (3) to the data (see Table 1). We write the leader vehicle dynamics (1) concisely as

$$\dot{x}_L(t) = f_L(x_L(t), u_L(t), w_L(t)) \tag{4}$$

where $x_L(t) := [p_L(t); h_L(t); d_L^{TL}(t); v_L(t); T_L^a(t)]$, $u_L(t) := [T_L^{a,ref}(t); T_L^b(t)]$, and $w_L(t) := v_F(t)$. Note that the velocity of the front vehicle $v_F(t)$ appears as a disturbance here. Since we cannot accurately predict the behavior of non-platooning vehicles, we make the conservative assumption that the front vehicle will decelerate from its current speed until coming to a stop. This assumed trajectory of the front vehicle is used for planning, to be discussed further in Section III-A.

We model the longitudinal dynamics of each of the $N - 1$ follower vehicles in the platoon as

$$\dot{p}_i(t) = v_i(t), \tag{5a}$$

$$\dot{h}_i(t) = v_{i-1}(t) - v_i(t), \tag{5b}$$

$$\dot{s}_i(t) = v_L(t) - v_i(t), \tag{5c}$$

$$\dot{v}_i(t) = \frac{1}{M}\left(\frac{T_i^a(t) - T_i^b(t)}{R_w} - F_f(t)\right), \tag{5d}$$

$$\dot{T}_i^a(t) = \frac{1}{\tau}\left(T_i^{a,ref}(t) - T_i^a(t)\right), \quad i = 1, \ldots, N, \tag{5e}$$

where $s_i(t)$, used for distance tracking relative to the leader vehicle, is defined as follows:

$$s_i(t) = \sum_{k=1}^{i} h_k(t). \tag{6}$$

We refer to $s_i(t)$ as the distance from follower i to the leader (note that (6) implies $s_1(t) = h_1(t)$). Furthermore, we let

$v_0(t) = v_L(t)$ so that (5c) is valid for follower $i = 1$. We write (5) compactly as

$$\dot{x}_i(t) = f_i(x_i(t), u_i(t), w_i(t)), \quad i = 1, \dots, N-1, \quad (7)$$

where $x_i(t) := [p_i(t); \; h_i(t); \; d_i^{TL}(t); \; v_i(t); \; T_i^a(t)]$, $u_i(t) := [T_i^{a,ref}(t); \; T_i^b(t)]$, and $w_i(t) := [v_L(t); \; v_{i-1}(t)]$. We note that the velocity of the leader and front vehicle $v_L(t)$ and $v_{i-1}(t)$ appear as disturbances here - since these are both platooning vehicles in this case, we can receive a forecast of their future behavior via V2V communication. In Section II-C we discuss the information transmitted between platooning vehicles which includes a velocity forecast, to be used as a disturbance preview in our MPC formulation.

For planning, our goal is to obtain linear, discrete time models from (4) and (7). We use the procedure outlined in [15] for doing so: we first linearize the leader and follower vehicle dynamics about the nominal velocities v_L^0 and v_i^0, respectively, and then discretize the resulting linear models each with time step $\Delta t = 0.1$s, resulting in

$$x_L(k+1) = A_L x_L(k) + B_L u_L(k) + E_L w_L(k),$$
$$x_i(k+1) \quad = A_i x_i(k) + B_i u_i(k) + E_i w_i(k), \quad (8)$$

where the matrices $A_L \in \mathbb{R}^{5 \times 5}$ and $B_L \in \mathbb{R}^{5 \times 2}$ are functions of the velocity v_L^0, and $A_i \in \mathbb{R}^{6 \times 6}$ and $B_i \in \mathbb{R}^{6 \times 2}$ are functions of the velocity v_i^0. At each time step, the current ego vehicle velocity is substituted into these expressions to obtain the appropriate dynamics matrices to be used for MPC.

B. STATE ESTIMATION

To localize the leader and follower vehicle positions $p_L(t)$ and $p_i(t)$ we use a differential GPS measurement which has lane-level accuracy. Furthermore, with GPS and information received from nearby traffic lights we can also estimate the distances $d_L^{TL}(t)$ and $d_i^{TL}(t)$ from each vehicle to the nearest upcoming traffic light. The forward-looking radar on each vehicle measures the headways $h_L(t)$ and $h_i(t)$, and standard on-board sensors provide the current velocity estimates $v_L(t)$ and $v_i(t)$, as well as estimates of the accelerating wheel torques $T_L^a(t)$ and $T_i^a(t)$.

An important sensing challenge for each follower i is to estimate the distance to the leader as defined in (6). We have tested two methods for doing so: 1) estimating $s_i(t)$ using GPS, and 2) estimating $s_i(t)$ directly using the radar measurements $h_i(t)$, which can be transmitted via V2V communication. For the first method, we use GPS to measure the distance $d_i^L(t)$ from the center of vehicle i to the center of the leader vehicle and use the estimate

$$\hat{s}_i(t) = \hat{d}_i^L(t) - i \cdot L_{\text{veh}} \quad (9)$$

where $\hat{d}_i^L(t)$ is an estimate of $d_i^L(t)$ from GPS. The main drawback to this approach is GPS measurement noise - we observed up to 3 meters of error when estimating $s_i(t)$ using GPS. Because of this, we also used a Kalman filter, where the idea is to use the current velocity of the leader (received

via V2V communication) and the ego vehicle velocity to improve our estimate of $s_i(t)$. For the second method, we use the estimate

$$\hat{s}_i(t) = \sum_{k=1}^{i} \hat{h}_k(t), \quad (10)$$

where $\hat{h}_k(t)$ is an estimate of $h_k(t)$ from radar. Since measurements from the forward-looking radar are generally very reliable, we observed smaller measurement errors using the second method. The main drawback to the second approach, however, is that it will require more vehicles in the platoon to communicate with one another (discussed further in the next section). For the experiments discussed in Section VI-B we used GPS to estimate $s_i(t)$, and for the experiments in Section VI-C we used radar measurements to estimate $s_i(t)$.

C. VEHICLE-TO-VEHICLE COMMUNICATION

We assume each platooning vehicle is capable of V2V communication. An important piece of information transmitted within the platoon is a forecast of the future velocity trajectory for each vehicle, given by

$$v_L^{\text{forecast}} = [v_L(t|t); \; v_L(t+1|t); \; \dots; \; v_L(t+N_p|t)],$$
$$v_i^{\text{forecast}} = [v_i(t|t); \; v_i(t+1|t); \; \dots; \; v_i(t+N_p|t)], \quad (11)$$

for the leader vehicle and follower vehicle i, respectively. Here, $v_L(k|t)$ is the planned velocity of the leader vehicle at time step k, obtained by solving an MPC problem at the current time step t (the notation is the same for the follower vehicles), and N_p is the MPC horizon in time steps. Each follower vehicle receives a velocity forecast from the front vehicle and the leader vehicle, corresponding to the flow of information depicted in Figure 3a. The front vehicle forecast is used to ensure safety, and the leader vehicle forecast is used to do distance tracking of the leader.

In addition to the velocity forecast, each experimental vehicle transmits its radar measurement, current GPS coordinates, and plan status signal. A secondary reason for transmitting GPS coordinates, beyond estimating $s_i(t)$, is so that the leader vehicle can estimate the distance $d_L^{N-1}(t)$ from itself to the rear platooning vehicle. The transmission of GPS coordinates from follower $N-1$ to the leader is shown in Figure 3b. This lets the leader check whether the entire platoon has enough time to pass through an upcoming intersection, as discussed in the next section. As mentioned in the previous section, for some of our experiments we used radar measurements, transmitted via V2V communication, to estimate $s_i(t)$. In Figure 3c we depict the flow of information in this case, for $N = 4$. We note that each vehicle, upon receiving an incoming message, checks the ID of the vehicle that transmitted it (indicating the vehicle's position in the platoon, e.g. leader vehicle, rear vehicle, etc.) to determine which information fields to extract, if any.
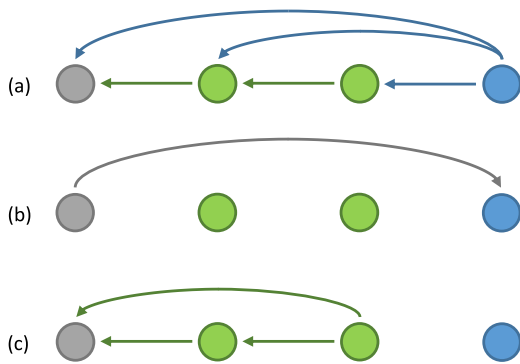
FIGURE 3. Flow of V2V messages for a platoon of size $N = 4$, where the blue node represents the leader vehicle and the grey node represents the rear vehicle. Figure 3a shows the transmission of velocity forecasts and 3b shows the transmission of GPS coordinates from the rear vehicle (used by the leader to determine if the platoon can make it through the intersection, see Section II-D). Figure 3c shows how we share radar measurements when we use the second method for estimating $s_i(t)$ as in (10).

D. VEHICLE-TO-INFRASTRUCTURE COMMUNICATION

In addition to V2V messages, we assume the platooning vehicles also receive SPaT (signal, phase, and timing) messages from nearby traffic lights via V2I communication. In this way, each vehicle obtains the following prediction on the nearest upcoming traffic light state:

$$\hat{x}_{TL}(t) = [p_{up}(t); \; c_r(t)] \tag{12}$$

where $p_{up}(t) \in \{red, yellow, green\}$ is the current phase of the nearest upcoming traffic light and $c_r(t) \in \mathbb{R}_{\geq 0}$ is a prediction on the time remaining in the current phase. We note that it is necessary to predict $c_r(t)$ here since in our experiments the traffic signals are actuated.

In the remainder of this section, we discuss how the leader decides whether or not the platoon should stop at an upcoming traffic light. This decision is handled by the leader only - the follower vehicles simply track the leader, and therefore we do not allow platoon separation. Suppose the platoon is approaching a traffic light during its green phase, with $c_r(t)$ seconds remaining in the phase. In this scenario, the leader checks if the following condition holds

$$c_r(t) \cdot v_L(t) \geq d_L^{N-1}(t) + d_L^{TL}(t) + L_{int} \tag{13}$$

to determine whether a stop is necessary (specifically, if (13) is false the platoon should stop), where L_{int} is the intersection length. Condition (13) provides a quick and simple way to check whether the rear platooning vehicle, travelling at the current leader velocity $v_L(t)$, will pass through the intersection during the green phase. We use $v_L(t)$ in (13) since the leader effectively sets the speed for all platooning vehicles behind it, and also to avoid having to transmit $v_{N-1}(t)$ to the leader. We note that when N is large, $d_L^{N-1}(t)$ is large and thus (13) is easily violated. This means the platoon may begin braking during a green light, which can be unexpected for nearby drivers. To avoid this, for large N allowing platoon separation may become necessary.
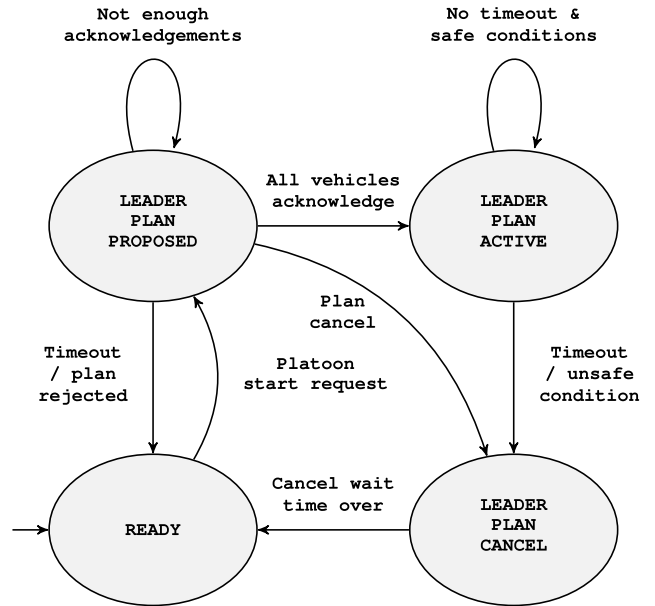

FIGURE 4. A diagram of the transitions in our finite state machine, shown here for the leader vehicle for simplicity.

At low velocity (13) is not easily satisfied and will be overly restrictive, for example if the light just turned green and the platoon is stopped. For this reason, if $v_L(t) \leq v_{low}$ the leader simply checks if the following condition holds

$$c_r(t) \geq t_{min} \tag{14}$$

where the threshold t_{min} is a tuning parameter. If so, it is considered safe to proceed. By checking (13) and (14) to determine whether to stop, we try to ensure the platoon will not be crossing the intersection when the phase becomes yellow. However, since the traffic signal is actuated and can change randomly due to uncertain traffic conditions, we cannot formally guarantee that this will never occur.

Suppose the leader determines it should stop while the phase is green, or that the phase is yellow, in which case the leader should stop if it can do so safely. Then, we also check if the leader is capable of stopping before the intersection stop line with a margin of d_{min}, that is

$$\frac{v_L(t)^2}{2a_{min,brake}} \leq d_L^{TL}(t) - d_{min} \tag{15}$$

where $-a_{min,brake} \in \mathbb{R}_{<0}$ is an upper bound on (1d) while the maximum braking force is applied. If (15) does not hold, then it is deemed safer for the leader to proceed through the intersection (in this scenario, for large N a platoon separation may also be necessary). For a red phase, however, we require the platoon to stop in any case.

E. FINITE STATE MACHINE (FSM)

We have designed a FSM (see Figure 4) which acts as a mechanism for safely forming and maintaining a platoon. There are four primary states in our FSM: 'Ready', 'Plan Proposed', 'Plan Active', and 'Plan Cancel'. Each platooning vehicle is initialized in the 'Ready' state and communicates

its state at all times. The platoon formation process is initiated when the leader moves to the 'Plan Proposed' state by proposing to the follower vehicles the 'plan', including a plan ID, ordering of the vehicles in the platoon, desired gap / speed, etc. Note that the ordering of vehicles in the platoon refers to the list of vehicle IDs ordered from the leader to the last follower. As soon as the 'plan' is received by the followers, the states of the followers transition to the 'Plan Proposed' state. In the 'Plan Proposed' state, each vehicle acknowledges that the 'plan' is valid by checking the on-board sensor data and communicated GPS data. For example, each vehicle can confirm that the driver agrees to join the platoon and that the proposed 'Plan' is safe to follow. We also note that the leader can manually cancel the plan while in the 'Plan Proposed' state, forcing a transition to the 'Plan Cancel' state.

When the leader receives an acknowledgement from every vehicle in the 'Plan', it moves to the 'Plan Active' state while also informing the followers so that all vehicles move to the 'Plan Active' state together. To ensure safety, while in the 'Plan Active' state every vehicle in the platoon continuously monitors the surrounding conditions to decide if the 'Plan' must stop. In our experiments, the conditions that cancel the plan include: 1) incorrect ordering of the vehicles, 2) message timeout, 3) any driver taps the gas / brake pedal, 4) front vehicle out of range (radar measurement too high), and 5) velocity upper / lower bound violated. Here, message timeout refers to when a particular message has not been received for a period of time longer than a specified threshold. When one of these conditions is detected by one vehicle, it informs the other vehicles in the platoon and they move together to the 'Plan Cancel' state. After some threshold time, each vehicle transitions from the 'Plan Cancel' state to the 'Ready' state and the platoon can be restarted as needed.

In Figure 5 we display some data collected while forming a platoon during testing in Arcadia, CA (see Section VI-C). The procedure for forming a platoon was to manually drive the test vehicles to get them close together and moving at similar speeds, at which point the leader vehicle would propose a 'plan' via the state machine and engage the platooning controllers simultaneously. This enabled platoon formation even while the vehicles are moving.

III. MPC FORMULATION

In this section we present our MPC problem formulation for the platoon. The leader vehicle has a separate MPC problem which allows it to react to changing traffic conditions and set the desired velocity for the following vehicles. For example, if a stop at an intersection is necessary, the leader computes a velocity trajectory in order to stop safely and comfortably at the intersection stop bar. Furthermore, the leader maintains a safe following distance when a vehicle is present ahead of it. The follower vehicles simply do distance tracking relative to the leader, as we do not allow platoon separation.
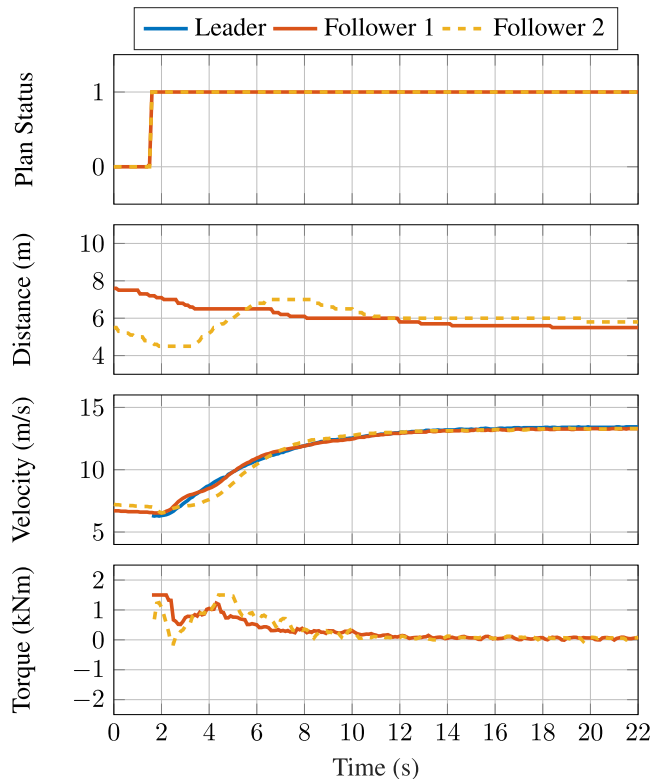

FIGURE 5. Experimental data collected in Arcadia, CA during the platoon formation process. The vehicles begin at a low speed and unequal spacing. At around the 2s mark, the platoon leader proposes a 'plan' which is accepted by the following vehicles, and the plan status signal (plotted above) switches from 0 to 1. This engages all platooning controllers simultaneously, and the vehicles quickly converge to the desired speed and distance.

A. LEADER VEHICLE MPC

The goal for the leader is to track a desired velocity when it is safe to do so. When necessary, it must yield to a slower-moving front vehicle or stop at the intersection stop bar. The MPC problem for the leader is

$$\min_{u_i(\cdot|t)} J_L = \sum_{k=t}^{t+N_p+1} (v_L(k|t) - v_L^{des})^2 \tag{16a}$$

$$+ \sum_{k=t}^{t+N_p} u_L(k|t)^T R u_L(k|t) \tag{16b}$$

$$+ \alpha \sum_{k=t}^{t+N_p-1} \|u_L(k+1|t) - u_L(k|t)\|^2 \tag{16c}$$

$$\text{s.t. } x_L(k+1|t) = \tag{16d}$$
$$A_L x_L(k|t) + B_L u_L(k|t) + E_L \hat{w}_L(k),$$

$$v_{\min} \le v_L(k|t) \le v_{\max}, \tag{16e}$$

$$d_{\min} + t_h v_L(k|t) \le d_L^*(k|t), \tag{16f}$$

$$0 \le T_L^a(k|t) \le T_{\max}^a, \tag{16g}$$

$$0 \le T_L^{a,ref}(k|t) \le T_{\max}^a, \tag{16h}$$

$$0 \le T_L^b(k|t) \le T_{\max}^b, \tag{16i}$$

$$x_L(t|t) = \hat{x}_L(t), \tag{16j}$$

$$\forall k = t, \ldots, t + N_p,$$

$$\begin{bmatrix} d_L^*(t + N_p|t) \\ v_L(t + N_p|t) \end{bmatrix} \in C(\hat{x}_L(t), \hat{v}_F(t), \hat{v}_F(t + N_p)), \quad (16k)$$

where N_p is the MPC horizon in time steps, and $x_L(k|t)$ and $u_L(k|t)$ are the planned state and input of the leader vehicle at time step k, computed at time step t, respectively (the notation for the other states is the same). Furthermore, $d_L^*(k|t)$ is the distance from the leader vehicle to either the front vehicle or the upcoming intersection stop bar - whichever is a higher priority obstacle (the method for determining this is outlined in Section IV). Lastly, $\hat{x}_L(t)$, $\hat{v}_F(t)$ are estimates of the leader vehicle and front vehicle state, based on measurements from the on-board sensors, and $\hat{v}_F(t + N_p)$ is an estimate of the front vehicle velocity at the end of the MPC planning horizon. Indeed, since $\hat{w}_L(k) := \hat{v}_F(k)$ appears as a disturbance in (16d), we must predict the future velocity trajectory of the front vehicle. To ensure safety, we assume worst-case behavior, i.e. the front vehicle will decelerate from its current speed at the rate $a_{\max,\text{brake}} \in \mathbb{R}_{>0}$ until coming to a complete stop as follows

$$\hat{w}_L(k) := \hat{v}_F(k)$$

$$= \begin{cases} \tilde{v}^0, & k = t, \\ \max(0, \; \hat{v}_F(k-1) & \quad (17) \\ -k \cdot a_{\max,\text{brake}} \cdot \Delta t), & k = t+1, \ldots, t+N_p, \end{cases}$$

where $\tilde{v}^0$ is an under-approximation of the front vehicle's current velocity v^0, to be discussed further in Section IV. Here, $-a_{\max,\text{brake}} \in \mathbb{R}_{<0}$ is a lower bound for (1d) and (5d) while the maximum braking force is applied.

The leader vehicle cost function J_L penalizes deviations from the desired velocity v_L^{des} (16a), nonzero control inputs (16b), and nonzero control input rates (16c), effectively penalizing vehicle jerk. The scalar $\alpha \in \mathbb{R}_{>0}$ and matrix $R \in \mathbb{R}^{2\times2}$ are design parameters which allow one to tune controller performance. Increasing α, for example, smooths the acceleration and deceleration profiles of the vehicle, but reduces the controller's agility. Furthermore, we set

$$R = \begin{bmatrix} R_a & R_0 \\ R_0 & R_b \end{bmatrix} \quad (18)$$

where the diagonal entries $R_a, R_b \in \mathbb{R}$ can be increased to encourage the controller to use smaller actuation torques $T_{a,ref}^L(t)$ and $T_a^L(t)$, respectively, and the off-diagonal entries $R_0 \in \mathbb{R}$ are made sufficiently large in order to prevent the accelerating and braking control inputs from being active simultaneously.

The leader MPC problem is subject to the following constraints: vehicle dynamics (16d), lower and upper bounds on velocity (16e), distance constraint (16f), torque and reference torque constraints (16g) - (16i), and initial condition (16j). The terminal constraint (16k) ensures the leader maintains a safe distance to any obstacle ahead (namely, a front vehicle or intersection requiring a stop), and will be discussed further in Section IV. The parameters $d_{\min}$ and t_h are tuned to increase

TABLE 2. MPC parameters.

d^{des}	desired distance	m	6
$d_{\min}$	minimum distance (front vehicle)	m	6
$d_{\min}$	minimum distance (stop bar)	m	5
t_h	time headway	s	1.6
v_L^{des}	desired velocity (leader)	m/s	15
$v_{\min}$	minimum velocity	m/s	0
$v_{\max}$	maximum velocity	m/s	20
$T_{\max}^a$	maximum accelerating torque	Nm	1500
$T_{\max}^b$	maximum braking torque	Nm	2000
N_p	MPC horizon	-	20

passenger comfort. For example, if t_h is too small it may feel as if the vehicle is braking late when approaching slow-moving traffic or a stop bar, and if t_h is too large the vehicle will brake harshly in response to cut-in vehicles. The values of all MPC parameters are given in Table 2.

At each time step, the leader vehicle solves its MPC problem and obtains an optimal control input sequence and velocity trajectory:

$$u_L(t|t), \; u_L(t+1|t), \; \ldots, \; u_L(t+N_p|t), \quad (19)$$

$$v_L(t|t), \; v_L(t+1|t), \; \ldots, \; v_L(t+N_p+1|t). \quad (20)$$

The first control input $u_L(t|t)$ of the sequence (19) is then implemented on the vehicle, and the MPC problem is solved again at the next time step. Furthermore, the computed velocity trajectory in (20) is sent to the other platooning vehicles at each time step via V2V communication as a velocity forecast, as discussed in Section II-C.

B. FOLLOWER VEHICLE MPC

The goal of each follower vehicle is to maintain a desired distance s_i^{des} to the leader vehicle, while also maintaining a minimum safety distance d_{min} to the front vehicle at all times. The MPC problem to be solved is defined as follows

$$\min_{u_i(\cdot|t)} J_i = \sum_{k=t}^{t+N_p+1} (s_i(k|t) - s_i^{des})^2 \quad (21a)$$

$$+ \sum_{k=t}^{t+N_p} u_i(k|t)^T R u_i(k|t) \quad (21b)$$

$$+ \alpha \sum_{k=t}^{t+N_p-1} \|u_i(k+1|t) - u_i(k|t)\|^2 \quad (21c)$$

$$\text{s.t. } x_i(k+1|t) = \quad (21d)$$

$$A_i x_i(k|t) + B_i u_i(k|t) + E_i \hat{w}_i(k),$$

$$v_{\min} \leq v_i(k|t) \leq v_{\max}, \quad (21e)$$

$$d_{\min} \leq h_i(k|t), \quad (21f)$$

$$0 \leq T_i^a(k|t) \leq T_{\max}^a, \quad (21g)$$

$$0 \leq T_i^{a,ref}(k|t) \leq T_{\max}^a, \quad (21h)$$

$$0 \leq T_i^b(k|t) \leq T_{\max}^b, \quad (21i)$$

$$x_i(t|t) = \hat{x}_i(t), \quad (21j)$$

$$\forall k = t, \ldots, t + N_p,$$

$$\begin{bmatrix} h_i(t + N_p|t) \\ v_i(t + N_p|t) \end{bmatrix} \in \mathcal{C}_F(\hat{v}_{i-1}(t + N_p)), \qquad (21k)$$

where the notation used is the same as in (16). The follower vehicle objective function J_i penalizes deviations from the desired distance to the leader vehicle, given by

$$s_i^{\text{des}} := d^{\text{des}} \cdot i, \qquad (22)$$

where d^{des} is a design parameter. Furthermore, we also include penalties on input (21b) and jerk (21c). Similar to the leader, these penalties have to be adjusted carefully to balance performance and passenger comfort. Furthermore, we note that constraints (21e) and (21k) are imposed with respect to the front (platooning) vehicle only, since safety tasks regarding an upcoming intersection are handled by the platoon leader (the terminal constraint (21k) will be discussed further in the next section).

Similar to the leader, at each time step the follower vehicle solves its MPC problem and obtains an optimal control input sequence and velocity trajectory. We apply the first control input of the sequence, and the computed velocity trajectory is broadcast to the platoon via V2V communication. Hence, since velocity forecasts (11) are received by all follower vehicles via V2V communication, we use the following disturbance preview for MPC:

$$\begin{aligned} \hat{w}_i(k) &:= [\hat{v}_L(k); \ \hat{v}_{i-1}(k)] \\ &= [v_L(k|t); \ v_{i-1}(k|t)], \quad k = t, \ldots, t + N_p, \quad (23) \end{aligned}$$

where the planned velocity trajectories $v_L(k|t)$ and $v_{i-1}(k|t)$ were computed by the leader and front vehicle when they solved their respective MPC problems.

Remark 1: Since we use the *full* velocity forecast as a disturbance preview in (23), a natural question that arises is whether or not these predictions are reliable. To address this question, in [15] we defined the *trust horizon F*, which allows us to adjust how much of the velocity forecasts are used. For a trust horizon of F, time steps t through $t + F$ of all velocity forecasts are used. After time step $t + F$ the front vehicle is assumed to decelerate at the maximum rate until coming to a stop, and therefore the terminal constraint (21k) is imposed at time step $t + F$. This is in contrast to the approach in this paper, where we assume the front (platooning) vehicle will fully realize the trajectory in its velocity forecast as in (23), corresponding to $F = N_p$. Doing so introduces some risk to the follower vehicles; however, this is necessary to achieve a reasonable increase in traffic throughput with vehicle platooning, as shown in our previous work [15].

IV. SAFETY CONSTRAINTS AND MPC SOLUTION

We now discuss how we formally ensure safety in an urban traffic setting. First, in Section IV-A we describe the set of safe states for a vehicle in relation to the two primary obstacles it can encounter in an urban setting: another vehicle ahead of it, and an upcoming intersection. Furthermore, we show that at each time instant the vehicle needs to consider only one of these obstacles, which we refer to as the *priority*

obstacle, thereby simplifying the task of ensuring safety. Next, in Section IV-B we discuss how we use the safe sets from Section IV-A in our MPC problems, as well as how we efficiently solve the MPC problems at runtime.

A. SAFE STATES AND PRIORITY OBSTACLE

Consider an ego vehicle (representing either a platoon leader or follower here), a front vehicle ahead of it, and an upcoming intersection. Throughout the section, we let $a(t)$ and $a_F(t)$ be the accelerations of the ego and front vehicles, respectively, so that the vehicle dynamics become

$$\begin{aligned} \dot{h}(t) &= v_F(t) - v(t), \\ \dot{d}^{TL}(t) &= -v(t), \\ \dot{v}_F(t) &= a_F(t), \\ \dot{v}(t) &= a(t), \qquad (24) \end{aligned}$$

where $h(t)$ is the headway of the ego vehicle, $d^{TL}(t)$ is the distance from the ego vehicle to the upcoming traffic light stop bar, and $v_F(t)$ and $v(t)$ are the velocities of the front and ego vehicles, respectively. Since we observed no actuation delay while braking during experimentation, it is sufficient to use (24) in place of (1) for the analysis here.

We first assume that only a front vehicle is present, and define safety for the ego vehicle with respect to the front vehicle as

$$h(t) \geq d_{\min}, \quad t \geq 0. \qquad (25)$$

To enforce (25), the ego vehicle must ensure it can maintain a minimum safety distance $d_{\min}$ if the front vehicle applies the maximum braking force until coming to a stop. We formalize this requirement in the following Proposition:

Proposition 1: Consider the vehicle dynamics given in (24). Let $a_{\min,\text{brake}}$, $a_{\max,\text{brake}} \in \mathbb{R}_{>0}$, and $a_{\min,\text{brake}} \leq a_{\max,\text{brake}}$. Suppose the accelerations $a_F(t)$ and $a(t)$ satisfy

$$a_F(t) = \begin{cases} -a_{\max,\text{brake}}, & t \in [0, t_F^s], \\ 0, & t > t_F^s, \end{cases} \qquad (26)$$

$$a(t) = \begin{cases} -a_{\min,\text{brake}}, & t \in [0, t^s], \\ 0, & t > t^s, \end{cases} \qquad (27)$$
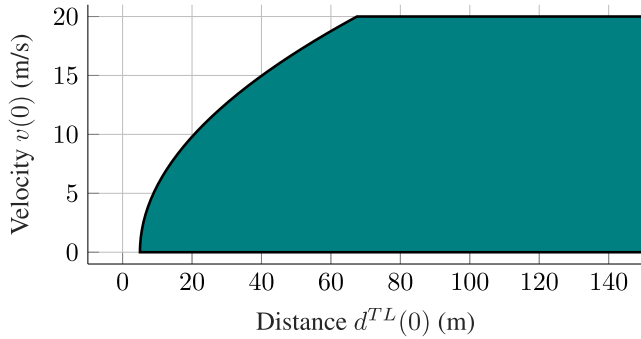
where $t_F^s := v_F(0)/a_{\max,\text{brake}}$ and $t^s := v_L(0)/a_{\min,\text{brake}}$ are the first time instants in seconds such that $v_F(t_F^s) = 0$ and $v(t^s) = 0$, respectively. Then, (25) will hold if $[h(0); v(0)] \in \mathcal{C}_F(v_F(0))$, where

$$\mathcal{C}_F(v_F(0))$$
$$:= \left\{ \begin{bmatrix} h(0) \\ v(0) \end{bmatrix} : \begin{array}{l} h(0) \geq \dfrac{v(0)^2}{2\,a_{\min,\text{brake}}} - \dfrac{v_F(0)^2}{2\,a_{\max,\text{brake}}} + d_{\min}, \\ h(0) \geq d_{\min}, \quad v(0) \geq 0 \end{array} \right\}$$
$$(28)$$

for $v_F(0) \in \mathbb{R}_{\geq 0}$. For a proof we refer to [23], Lemma 1 (see also [17], [24]). We note that in addition to $v_F(0)$, the set
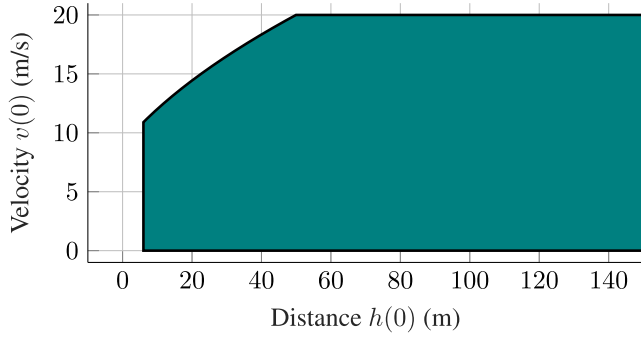
(a) $\mathcal{C}_F(v_F(0))$ for $v_F(0) = 14$m/s and $d_{\min} = 6$m.

(b) $\mathcal{C}_{TL}$ for $d_{\min} = 5$m.

FIGURE 6. In 6a and 6b we plot the terminal sets (28) and (30) for the front vehicle and upcoming intersection, respectively. For computing the sets, we use $a_{\min,\text{brake}} = 3.2$ m/s^2 and $a_{\max,\text{brake}} = 5.0912$ m/s^2.

$\mathcal{C}_F(v_F(0))$ also depends on $a_{\min,\text{brake}}, a_{\max,\text{brake}}, d_{\min} \in \mathbb{R}_{>0}$. A plot of $\mathcal{C}_F$ is given in Figure 6a.

Next, we suppose that only an upcoming intersection requiring a stop is present. In this case, the ego vehicle must ensure it can make a complete stop and leave a distance of $d_{\min}$ to the intersection stop bar. Formally, we require that if the ego vehicle decelerates until coming to a stop as in (27), then the following will hold

$$d^{TL}(t) \geq d_{\min}, \quad t \geq 0. \tag{29}$$

We note that when the light cycles to green, this constraints is relaxed and the platoon is allowed to proceed. Similar to Proposition 1, we can show that (29) holds if the ego vehicle decelerates as in (27) and $[d^{TL}(0); \, v(0)] \in \mathcal{C}_{TL}$, where

$$\mathcal{C}_{TL} := \left\{ \begin{bmatrix} d^{TL}(0) \\ v(0) \end{bmatrix} : \begin{array}{l} d^{TL}(0) \geq \dfrac{v(0)^2}{2\,a_{\min,\text{brake}}} + d_{\min}, \\[2mm] v(0) \geq 0 \end{array} \right\}. \tag{30}$$

A plot of $\mathcal{C}_{TL}$ is given in Figure 6b.

Now, we suppose that both a front vehicle and an upcoming intersection requiring a stop are present simultaneously. In this scenario, we require that if the front and ego vehicle (representing the platoon leader here) decelerate until coming to a stop as in (26) and (27), then *both* (25) and (29) will hold.

(a) Truck has priority.

(b) Intersection has priority.

FIGURE 7. View from the middle platooning vehicle as it approaches an intersection during our demonstration in Arcadia, CA. In Figure 7a there is a slow-moving truck attempting to turn right ahead of the leader vehicle. Since the truck takes priority over the intersection at this point, the platoon is forced to slow down. In Figure 7b the truck completes the right turn and priority shifts to the intersection.

To determine which obstacle is prioritized, the ego vehicle can check if the following condition holds:

$$h(0) + \frac{v_F(0)^2}{2a_{\max,\text{brake}}} \leq d^{TL}(0). \tag{31}$$

If (31) holds then the front vehicle is capable of stopping in front of the intersection stop line, and therefore must be prioritized. If (31) does not hold then the upcoming intersection is prioritized (see Figure 7 for an illustration). We summarize this idea in the following Proposition, which follows directly from the definitions of $\mathcal{C}_F$ and $\mathcal{C}_{TL}$.

Proposition 2: Assume $h(0) \geq d_{\min}$. If (31) does not hold, then $[d^{TL}(0); \, v(0)] \in \mathcal{C}_{TL}$ implies $[h(0); \, v(0)] \in \mathcal{C}_F(v_F(0))$. Otherwise, if (31) holds, then $[h(0); \, v(0)] \in \mathcal{C}_F(v_F(0))$ implies $[d^{TL}(0); \, v(0)] \in \mathcal{C}_{TL}$.

Based on Proposition 2, we conclude that for the leader vehicle MPC problem discussed in the previous section, it is sufficient to impose a terminal constraint with respect to only the priority obstacle. This is beneficial for efficiently solving the MPC problems at runtime, as discussed further in the next section.

Remark 2: In the above discussion we assumed $d_{\min}$ is the same for both the front vehicle and the intersection, whereas

in our experiments we used slightly different values of $d_{\min}$ for each. Although this is beneficial for passenger comfort, there is one drawback to this adjustment: in corner cases where priority between the two obstacles can easily switch, we may only satisfy (25) and (29) for the minimum of these two values, i.e. for $d_{\min} := \min\{d_{\min,F},\ d_{\min,TL}\}$, where $d_{\min,F}$ and $d_{\min,TL}$ are the unique minimum distance values used for the front vehicle and intersection, respectively. We ensured, however, that this minimum safety margin is still sufficient for testing purposes. Furthermore, in normal traffic conditions the priority between obstacles is clear (usually, the front vehicle is clearly stopping at the intersection, or clearly passing through it).

Remark 3: If an upcoming intersection is not present (or does not require a stop), then the front vehicle is prioritized if one is present. This allows, for example, the platoon to pass through a green light if it is safe to do so. Similarly, if only a front vehicle is present then it is prioritized. If neither obstacle is present, then no obstacle-related constraints are imposed on the leader.

B. TERMINAL CONSTRAINTS AND MPC SOLUTION

We now connect the discussion in the previous section to terminal constraints (16k) and (21k). For the follower vehicles, the primary safety task is to maintain a minimum distance to the front (platooning) vehicle. Therefore, the terminal constraint (21k) is imposed with respect to the front vehicle only. For the leader vehicle, the primary safety tasks are to stop at an upcoming intersection when necessary, and to maintain a minimum distance to the front (non-platooning) vehicle. Based on the discussion in Section IV-A, this is accomplished by imposing the terminal constraint (16k) with respect to the priority obstacle. To this end, we define

$$
d_L^*(t+k|t)
$$
$$
:= \begin{cases} h_L(t+k|t), & \text{if } \hat{x}_L(t) \text{ and } \hat{v}_F(t) \text{ satisfy (31)}, \\ d_L^{TL}(t+k|t), & \text{otherwise}, \end{cases} \quad (32)
$$

as the planned distance from the leader to the priority obstacle at time step k, computed at time step t, and

$$
\mathcal{C}(\hat{x}_L(t), \hat{v}_F(t), \hat{v}_F(t+N_p))
$$
$$
:= \begin{cases} \mathcal{C}_F(\hat{v}_F(t+N_p)), & \hat{x}_L(t) \text{ and } \hat{v}_F(t) \text{ satisfy (31)}, \\ \mathcal{C}_{TL}, & \text{otherwise}, \end{cases} \quad (33)
$$

as the terminal set with respect to the priority obstacle. We note that the priority obstacle will be the same throughout the MPC planning horizon, since (31) checks whether the front vehicle can stop before the intersection stop bar if it decelerates at the rate $a_{\max,\text{brake}}$, which is its assumed behavior in the leader MPC problem in (17).

To solve the leader and follower vehicle MPC problems at runtime we use the tool CVXGEN [25], which allows one to generate C code for solving a custom quadratic program (QP) reliably and efficiently. Since CVXGEN can only be used for moderately-sized QPs, it is beneficial to impose terminal

constraint (16k) with respect to only the priority obstacle, as imposing a terminal constraint with respect to both obstacles would create additional (redundant) constraints. Furthermore, since our MPC problems must be represented as QPs with linear constraints, the sets $\mathcal{C}_F$ and $\mathcal{C}_{TL}$ discussed in the previous section cannot be directly encoded into our MPC problems. Instead, we use a procedure from [26] to compute polyhedral constraint sets to be used in place of $\mathcal{C}_F$ and $\mathcal{C}_{TL}$. In particular, we compute a collection of sets $\mathcal{C}_F(v_F(0))$ to be used for $v_F(0) \in [v_{\min}, v_{\max}]$. This collection of sets is computed offline, and the proper set is selected during runtime to be used for MPC (for more details, we refer the reader to [15]).

Since it is important to avoid infeasibility of the MPC problems during experimentation, all constraints in each problem (except for the vehicle dynamics constraints) are converted to soft constraints. This means that for a hard constraint such as $Gx \leq h$, where $x \in \mathbb{R}^n$, $G \in \mathbb{R}^{m \times n}$, and $h \in \mathbb{R}^m$, we instead add the term $\lambda \Bbbk^T (Gx - h)_+$ to the objective function, where $\lambda \in \mathbb{R}_{>0}$, $\Bbbk \in \mathbb{R}^m$ is the vector of all 1's, and y_+ for $y \in \mathbb{R}^m$ indicates that we are thresholding each element of y so that $y_+ \in \mathbb{R}_{\geq 0}^m$ (see [27]).

V. SIMULATION RESULTS

We now present results from our simulation tool developed in MATLAB, which enabled us to validate the platooning software prior to conducting real-world experiments. In particular, the tool is useful to confirm that the platoon preserves safety even when it encounters traffic lights and other non-platooning vehicles, using the approach in Sections II-D and IV. Furthermore, we are able to estimate the potential gains in traffic throughput at intersections, using a metric from [15].

A. URBAN STOP AND GO SCENARIO

We use our tool to simulate the vehicle platoon travelling along an arterial roadway with moderate traffic. In particular, our goal here is to imitate the conditions we will encounter during our field experiments in Arcadia, CA (see Section VI-C). To simulate public vehicles in traffic, we create velocity trajectories in simulation and then replay them so that simulations are repeatable. Taking into account the positions / velocities of the platoon leader and a public vehicle ahead of it, we can send radar signals as an input to the leader controller and observe how the platoon responds. Furthermore, we can also create signalized intersections with the following attributes: position (m), V2I communication range (m), cycle offset (s), red / yellow / green time (s), and cycle length (s). We placed intersections along the simulated arterial road so that the distances between traffic lights are similar to the Arcadia corridor discussed in Section VI-C. All the individual intersections are composed to create a traffic network object which can be queried to determine the nearest upcoming traffic light relative to the platoon leader. As the platoon leader approaches the intersection, we send V2I messages from that traffic light as
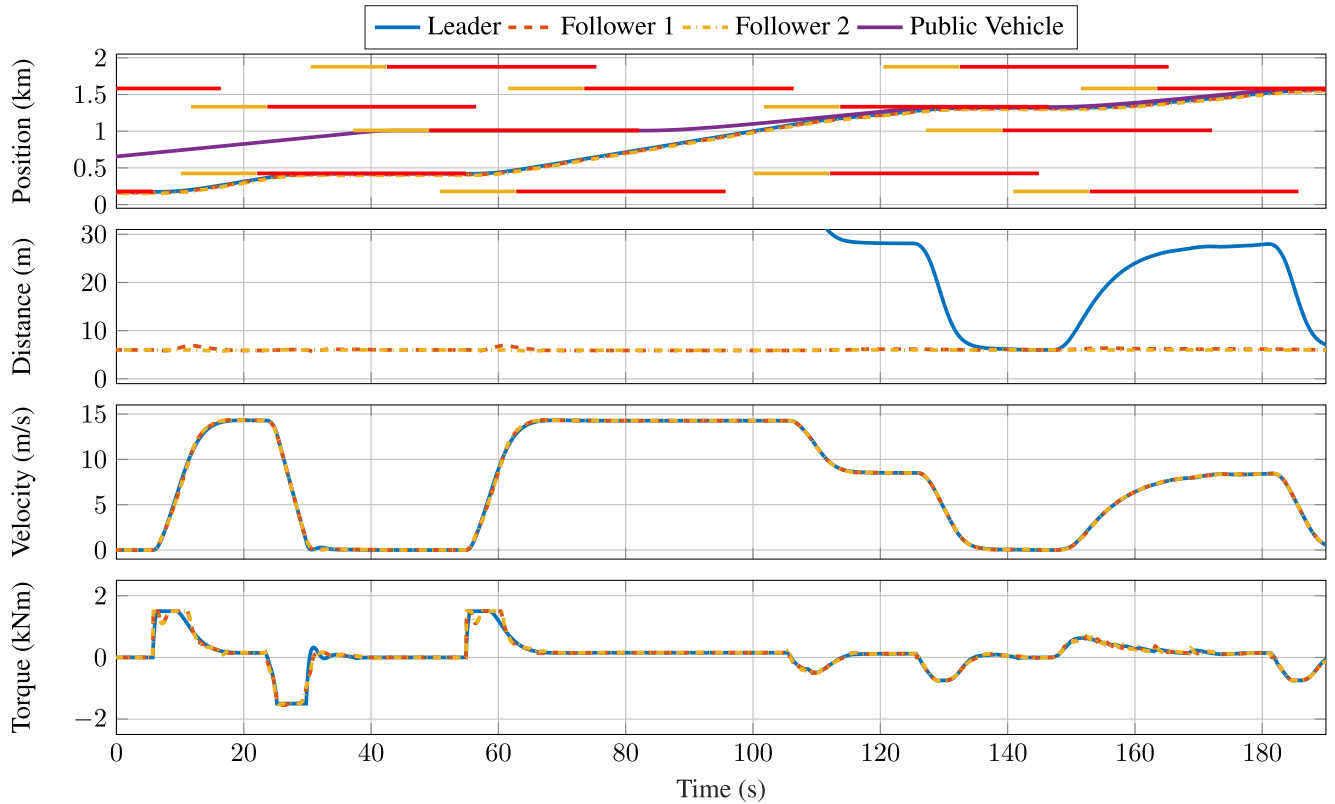
FIGURE 8. Simulation results for an urban traffic scenario with a non-platooning lead vehicle and multiple signalized intersections. In the top plot, we show the position of all simulated vehicles (including the public vehicle which is not platooning), as well as the position of each intersection which has either a yellow or red phase. In the bottom three plots, we show the inter-vehicle distances (including the distance from the leader to the public vehicle), velocities, and torque commands for the platooning vehicles.

an input to the leader controller and simulate the platoon response.

The simulation results are shown in Figure 8. In particular, we note that the horizontal yellow and red lines in the top plot represent intersections which have a yellow and red phase at that time instant, respectively. Furthermore, the purple line represents the position of a public vehicle which is not platooning. In the beginning of the simulation, the platoon encounters red lights at the first few intersections, stopping at each. Near the end of the simulation the platoon approaches a (non-platooning) public vehicle which is travelling much more slowly, and the platoon is forced to reduce its speed for the remainder of the simulation. We note that near the end of the simulation, the public vehicle comes to a complete stop at an intersection and as a result the platoon leader also stops, leaving a distance of 6m as desired. ThreeAs mentioned previously, one of the primary goals of the simulation tool is to verify that the platoon responds appropriately when it encounters other non-platooning vehicles and signalized intersections. Observing the simulation results, we can see that the platoon stops at each intersection when necessary, and that the distance from the leader to the public vehicle stays above 6m at all times as desired. Lastly, we remark that for simulation we did not use the same controller parameters that we did for experimentation, where the parameters were mainly selected to improve passenger comfort.

B. ESTIMATING THROUGHPUT

We now analyze the performance of the vehicle platooning system by estimating intersection throughput. To do so, we recall a performance metric defined in [15]. At time $t = 0$ let the platoon be stopped at the (current) intersection stop bar with no vehicles ahead

$$[p_L(0); \ v_L(0)] = [-d_{\min}; \ 0],$$
$$[p_i(0); \ v_i(0)] = [-d_{\min} - (L_{\text{veh}} + d^{\text{des}}) \cdot i; \ 0],$$
$$i = 1, \dots, N-1,$$

where L_{veh} is the vehicle length (assumed to be uniformly 4.5 meters for all vehicles), and the intersection stop bar is assumed to be positioned at 0 meters. Suppose at time $t = 0$ the traffic light cycles from red to green, and the platoon immediately starts moving through the intersection. Let $\ell \in \mathbb{R}_{>0}$ be the length of the intersection in meters, and define t_L and t_{N-1} to be the smallest time instants in seconds such that $p_L(t_L) \geq \ell$ and $p_{N-1}(t_{N-1}) \geq \ell$, respectively. We then estimate intersection throughput in vehicles per hour as

$$\text{throughput (vph)} \approx 3600 \cdot \frac{N-1}{t_{N-1} - t_L}. \tag{34}$$

Thus, performance is maximized when the platoon 1) accelerates to a high velocity while crossing the intersection, and 2) accurately maintains the desired inter-vehicle gaps while

TABLE 3. Improved throughput.

Simulation Intersection 1	4,336.4 vph
Simulation Intersection 2	4,336.4 vph
Simulation Intersection 4	2,477.8 vph
Test Track Intersection (Virtual)	4,463.4 vph

TABLE 4. Baseline throughput.

Simulation Intersection 1	2149.8 vph
Simulation Intersection 2	2156.9 vph
Simulation Intersection 4	1710.5 vph
Test Track Intersection (Virtual)	2730.7 vph

FIGURE 9. Depiction of the on-board hardware setup for the test vehicles. The local CAN bus (in red) connects the computational devices (Matrix embedded PC and dSPACE MicroAutoBox) to the Cohda OBU for DSRC communication. The HCU (CAN gateway) provides an interface between the local CAN bus and the production systems of the test vehicle. Using the local CAN bus and the gateway functionality of the HCU, we can send commands and access measurements to and from the production systems without needing access to proprietary vehicle data.

accelerating. We note that for the estimate (34) to be accurate, we must consider the length of each vehicle, as opposed to treating each as a point mass.

Throughput analysis of simulation results (as well as the test-track experiments discussed in Section VI-B) is shown in Tables 3 and 4, where all estimates are obtained via (34). In particular, throughput is estimated at the 1st, 2nd, and 4th intersection, located at approximately 0.18 km, 0.43 km, and 1.33 km in the simulation, respectively. In Table 3 we show improved levels of throughput achieved using our vehicle platooning system, which are estimated from the simulation run shown in Figure 8. In Table 4 we show baseline levels of throughput, which are estimated by running the same simulation with the *trust horizon* (discussed in Remark 1) set to $F = 0$. We note that throughput is much lower at the 4th intersection, due to the presence of a slower-moving public vehicle ahead of the platoon. Indeed, in situations like this, the benefit of vehicle platooning in terms of traffic throughput may not be fully realized. We note also that our predictions here are in line with predictions from previous works which utilized simulation. For example, in [6] the authors predict that vehicle platooning could enable a saturation flow rate of 4800 vph per intersection movement.

VI. EXPERIMENTAL RESULTS
In this section we present the experimental results and evaluate the performance of our platooning controller via the throughput metric from Section V-B. We discuss the experimental setup in Section VI-A, and in Section VI-B we present results from preliminary tests on a closed track at the Hyundai-KIA Motors California Proving Grounds in California City, CA. Next, we give an overview of a final platooning demonstration on public roadways in Arcadia, CA in Section VI-C. Links to drone videos of each series of tests are also provided.

A. TEST VEHICLES
We use the three test vehicles shown in Figure 1, each of which is equipped with a production forward-looking radar and camera that estimate the front vehicle distance, velocity, and acceleration. To enable V2V and V2I communication,
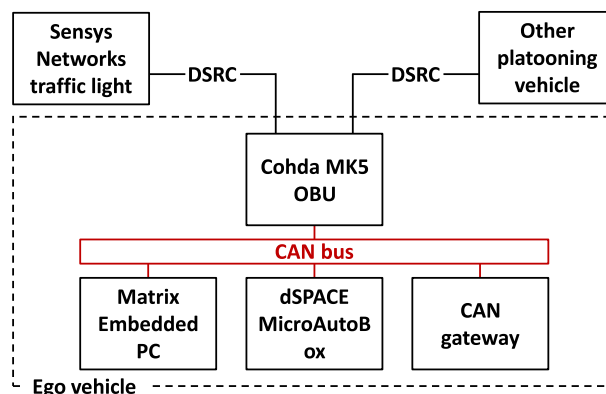
we use a Cohda Wireless MK5 V2X on-board unit (OBU), which also has an integrated GPS. The Cohda OBU allows the vehicles to exchange BSMs and custom V2V messages, which include a velocity forecast and other information. This transmitted information allows the third vehicle in the platoon, for instance, to estimate its current distance to the leader vehicle. The Cohda also allows each vehicle to communicate with any nearby traffic lights which are instrumented to broadcast SPaT and MAP messages. Lastly, the controller for each vehicle is implemented on a dSpace MicroAutoBox, and a Matrix embedded PC exchanges information between the Cohda, MicroAutoBox, and the ego vehicle controller area network (CAN bus). The Matrix also runs a state machine which manages the role of each vehicle in the platoon, and is discussed further in Section II-E. A diagram of the hardware setup is shown in Figure 9.

An important hardware consideration for platooning is that of communication latencies. In [15] we discussed how including a time stamp in transmitted messages enables each vehicle to account for V2V communication delays. The idea is to use the time stamp to estimate the delay d in time-steps (with sampling time $\Delta t = 0.1$s), and then to shift the velocity forecast used for MPC by d steps, where we assume the transmitting vehicle will maintain a constant velocity beyond its planned trajectory. For the experimental work presented in this paper, however, we assume there are no communication delays between vehicles, which is done for two reasons. The first reason is that we have observed that communication latencies are typically small enough to be ignored for our application. The second reason is that estimating d accurately is challenging in practice. Since the clocks on the test vehicle computers are not synchronized, one must estimate the clock skew between vehicles, which could potentially be time-varying, in order to accurately estimate delays.
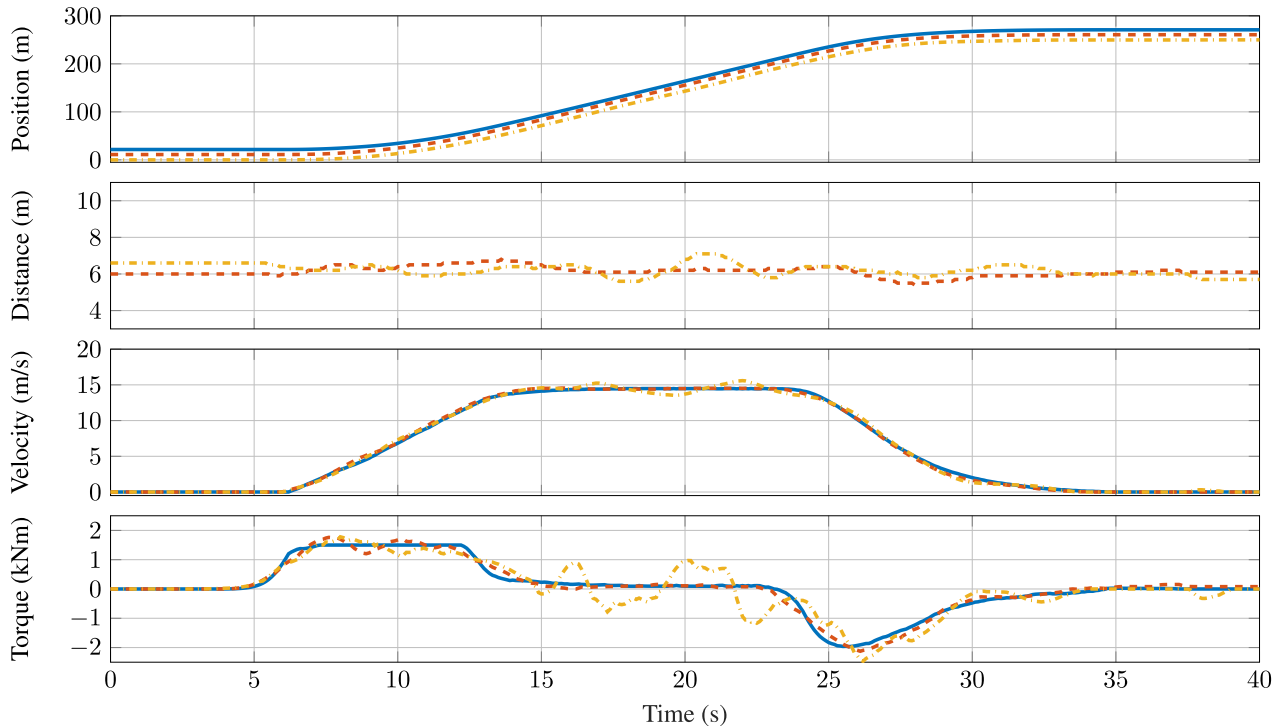
FIGURE 10. Experimental results from the Hyundai-KIA Motors California Proving Grounds with the test vehicles shown in Figure 1. Here, we had the platoon track a reference trajectory which was generated via our simulation tool. The position, inter-vehicle distance, velocity, and MPC torque command for each vehicle are shown in each subplot, respectively. The desired distance between vehicles was 6 meters.

B. CLOSED TRACK EXPERIMENTS

Preliminary vehicle platooning experiments were conducted on a closed test track at the Hyundai-KIA Motors California Proving Grounds in California City, CA (see Figure 1). For all of the tests the leader vehicle does velocity tracking of a predetermined velocity trajectory (meaning v_L^{des} in (16a) becomes time-dependent), and the follower vehicles do distance tracking relative to the leader vehicle. The predetermined velocity trajectories used for tracking were either from real velocity data collected during previous experiments, or artificial velocity data generated by our simulation tool. In Figure 10 we show experimental results from a test using artificial velocity data which has a step function-like trajectory. For these experiments we used a larger admissible range of the wheel torque for the follower vehicles, as seen in the bottom plot of Figure 10. In particular, we note that the torque plotted is the *desired* torque, i.e. the output of the MPC algorithm, as opposed to the *measured* torque (estimated by the vehicle). However, the inter-vehicle distances and vehicle velocities are both from on-board measurements (the position data is then obtained offline by integrating the velocity data). We can see that as the platoon accelerates and decelerates, the followers accurately track the desired distance of 6m to the front vehicle - all tracking errors stay below about 1m throughout the experiment. We note, however, that there is slightly larger tracking error (as well as larger variation of the wheel torque command) for the second follower in this experiment. We can mainly attribute this to state estimation error since GPS was used to estimate the distance

$s_i(t)$ for all experiments at the California Proving Grounds, as discussed in Section II-B.

A video of the testing is available online at `https://youtu.be/U-O9iUZElR8`, which includes several test runs with varying levels of the *trust horizon F* (discussed in Remark 1). We note that in test runs with a small trust horizon, for example $F = 10$ (half of the velocity forecast is trusted) or $F = 0$ (none of the velocity forecast is trusted, meaning the vehicles effectively do not use V2V communication), large gaps appear between the platooning vehicles while they are accelerating. This behavior is expected, since using the full velocity forecast relaxes the constraints on following distance so that the follower vehicles can get closer to the vehicle ahead. In the test run shown in Figure 10 we used $F = 15$, demonstrating that we are able to get accurate tracking performance when using a large portion of the velocity forecast (elsewhere in the paper we use $F = N_p = 20$). Similar to Section V-B, we estimate throughput for the test run shown in Figure 10 by treating the platoon as if it begins stopped at an intersection - our estimate is shown in Table 3. Furthermore, in Table 4 we show a baseline level of throughput computed using data from a test run with $F = 0$. As expected, significantly higher throughput is achieved by utilizing the velocity forecast.

C. PUBLIC ROAD DEMONSTRATION

To demonstrate vehicle platooning in an urban environment with a moderate level of traffic, we conducted further experiments in Arcadia, CA. Our testing area is a 2.45 km long
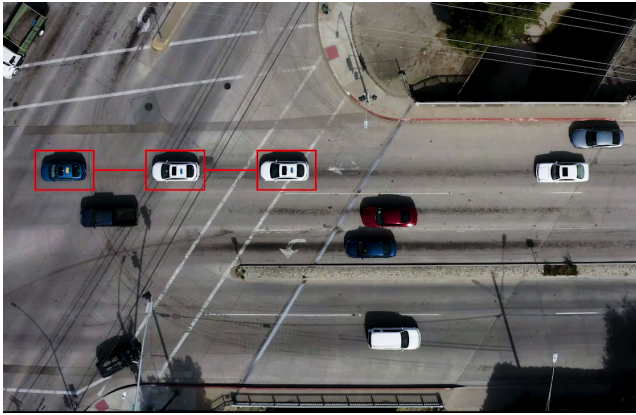
FIGURE 11. Overhead view of the platoon crossing an intersection in Arcadia, CA.

stretch of roadway on Live Oak Ave between S Santa Anita Ave and Peck Rd, and has eight consecutive intersections which are instrumented to send out SPaT and MAP messages for our vehicle platoon to receive. All tests in Arcadia were completed with a 3-vehicle platoon using the same MPC parameters as shown in Table 1, with the exception that $v_L^{des} = 14$ m/s was used here. Footage of our testing is available online: `https://youtu.be/xPYR_xP3FuY`. It captures a few instances where the platoon stops at the stop bar for a red light with no vehicles queued ahead of it. When the light turns green the platoon reacts immediately and moves through the intersection more quickly and compactly than the human-driven vehicles near it, further demonstrating the potential for throughput improvement (see Figure 11).

VII. CONCLUSION AND FUTURE WORK

In this paper we presented the design and evaluation of a vehicle platooning system that can operate in an urban corridor with intersections and other traffic participants. The primary motivation for advancing platooning to an urban setting is to improve traffic efficiency by increasing throughput at intersections, which create bottlenecks for traffic flow. We evaluated the performance of our vehicle platooning architecture by estimating the level of throughput that would be achieved at the intersection, calculated by measuring the time instants at which each vehicle crosses the intersection.

An important challenge we encountered while testing on public roads is that of safely disengaging the platooning system and passing control back to the safety driver when necessary. To do so, we designed our system so that if any driver taps the brake pedal when the platoon is active, the controller for every platooning vehicle disengages immediately (via the finite state machine) and all drivers are notified immediately via a sound. We note, however, that this design can be problematic in certain scenarios. For example, suppose the platoon is approaching an intersection and begins braking when the driver in the leader vehicle, out of caution, disengages the platoon. This requires the drivers in the follower vehicles to react immediately, as their vehicles will suddenly stop braking when the controllers disengage. In the future we hope

to address this issue by creating a safety system that ensures the vehicles start transitioning to a safe state immediately when the 'plan' is cancelled, providing the safety driver more time to react. One potential approach, for example, is to have the platooning system transition to an ACC state of operation immediately after disengagement. The ACC system would then remain active and maintain a safe distance to the front vehicle until the driver takes over.

Another future research direction relates to the procedure for setting cost weights in the MPC objective functions, which were manually tuned here. Indeed, in order to converge on acceptable values for the tuning parameters affecting vehicle drivability, such as the time headway constraint or penalty on vehicle jerk, multiple trial runs on a closed test track were necessary. To reduce development time, it would be interesting to see how a learning-based approach could potentially expedite this procedure. Furthermore, we note that the final tuning values we obtained are only valid for the class of test vehicles in this paper - other classes of vehicles, such as semi-trucks, have different performance characteristics and would therefore need separate tuning values. Learning-based performance tuning would also be beneficial for deploying a platoon with a large number of vehicles, since separate tuning values were used for each vehicle within the platoon in this paper. Learning could also accelerate the deployment of autonomous vehicles more broadly, by enabling companies to more easily meet customer's driving preferences.

REFERENCES

[1] J. Guanetti, Y. Kim, and F. Borrelli, "Control of connected and automated vehicles: State of the art and future challenges," *Annu. Rev. Control*, vol. 45, pp. 18–40, Feb. 2018.

[2] *V2V Safety Technology Now Standard on Cadillac CTS Sedans.* Accessed: Apr. 1, 2020. [Online]. Available: https://media.cadillac.com/media/us/en/cadillac/news.detail.html/content/Pages/news/us/en/2017/mar/0309-v2v.html

[3] E. Uhlemann, "Platooning: Connected vehicles for safety and efficiency [Connected Vehicles]," *IEEE Veh. Technol. Mag.*, vol. 11, no. 3, pp. 13–18, Sep. 2016.

[4] *Red Light, Green Light—No Light: Tomorrow's Communicative Cars Could Take Turns at Intersections.* accessed: Oct. 4, 2018. [Online]. Available: https://doi.org/10.1109/MSPEC.2018.8482420

[5] S. E. Shladover, D. Su, and X.-Y. Lu, "Impacts of cooperative adaptive cruise control on freeway traffic flow," *Transp. Res. Rec., J. Transp. Res. Board*, vol. 2324, no. 1, pp. 63–70, Jan. 2012.

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

[7] C. Bonnet and H. Fritz, "Fuel consumption reduction in a platoon: Experimental results with two electronically coupled trucks at close spacing," SAE Warrendale, PA, USA, Tech. Rep. 2000-01-3056, 2000.

[8] S. E. Shladover, "PATH at 20—History and major milestones," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 4, pp. 584–592, Oct. 2007.

[9] A. Alam, B. Besselink, V. Turri, J. Mårtensson, and K. H. Johansson, "Heavy-duty vehicle platooning for sustainable freight transportation: A cooperative method to enhance safety and efficiency," *IEEE Control Syst. Mag.*, vol. 35, no. 6, pp. 34–56, Dec. 2015.

[10] J. Ploeg, B. T. M. Scheepers, E. van Nunen, N. van de Wouw, and H. Nijmeijer, "Design and experimental evaluation of cooperative adaptive cruise control," in *Proc. 14th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2011, pp. 260–265.

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

[12] D. Swaroop, "String stability of interconnected systems: An application to platooning in automated highway systems," Ph.D. dissertation, Dept. Mech. Eng., UC Berkeley, Berkeley, CA, USA, 1997.

[13] J. Ploeg, S. Shladover, H. Nijmeijer, and N. van de Wouw, "Introduction to the special issue on the 2011 grand cooperative driving challenge," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 989–993, Sep. 2012.

[14] E. van Nunen, M. R. J. A. E. Kwakkernaat, J. Ploeg, and B. D. Netten, "Cooperative competition for future mobility," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 3, pp. 1018–1025, Sep. 2012.

[15] S. W. Smith, Y. Kim, J. Guanetti, A. A. Kurzhanskiy, M. Arcak, and F. Borrelli, "Balancing safety and traffic throughput in cooperative vehicle platooning," in *Proc. 18th Eur. Control Conf. (ECC)*, Jun. 2019, pp. 2197–2202.

[16] A. Hsu, F. Eskafi, S. Sachs, and P. Varaiya, "Design of platoon maneuver protocols for IVHS," California Partners Adv. Transit Highways, Berkeley, CA, USA, Tech. Rep. UCB-ITS-PRR-91-6, 1991.

[17] P. Li, L. Alvarez, and R. Horowitz, "AHS safe control laws for platoon leaders," *IEEE Trans. Control Syst. Technol.*, vol. 5, no. 6, pp. 614–628, 1997.

[18] J. van de Sluis, O. Baijer, L. Chen, H. Bengtsson, L. Garcia-Sol, and P. Balaguer, "D3.2 proposal for extended message set for supervised automated driving," Eur. Commission, DG Res. Innov., Brussels, Belgium, Tech. Rep. 4092490, 2015.

[19] J. Ploeg, C. Englund, H. Nijmeijer, E. Semsar-Kazerooni, S. E. Shladover, A. Voronov, and N. van de Wouw, "Guest editorial introduction to the special issue on the 2016 grand cooperative driving challenge," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 4, pp. 1208–1212, Apr. 2018.

[20] V. S. Dolk, J. Ploeg, and W. P. M. H. Heemels, "Event-triggered control for string-stable vehicle platooning," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 12, pp. 3486–3500, Dec. 2017.

[21] J. Schindler, R. Blokpoel, M. Rondinone, T. Walter, O. Pribyl, H. Saul, A. Leich, D. Wesemeyer, and R. Dariani, "Maven deliverable 6.4: Integration final report," Eur. Commission, DG Res. Innov., Tech. Rep. 3322570, 2020.

[22] L. Guzzella, *Vehicle Propulsion Systems*, vol. 1. Springer, 2007, pp. 13–46.

[23] V. Turri, "Fuel-efficient and safe heavy-duty vehicle platooning through look-ahead control," Ph.D. dissertation, Dept. Autom. Control, KTH Royal Inst. Technol., Stockholm, Sweden, 2015.

[24] S. Shalev-Shwartz, S. Shammah, and A. Shashua, "On a formal model of safe and scalable self-driving cars," 2017, *arXiv:1708.06374*. [Online]. Available: http://arxiv.org/abs/1708.06374

[25] J. Mattingley and S. Boyd, "CVXGEN: A code generator for embedded convex optimization," *Optim. Eng.*, vol. 13, no. 1, pp. 1–27, Mar. 2012.

[26] S. Lefevre, A. Carvalho, and F. Borrelli, "A learning-based framework for velocity control in autonomous driving," *IEEE Trans. Autom. Sci. Eng.*, vol. 13, no. 1, pp. 32–42, Jan. 2016.

[27] *CVXGEN: Code Generation for Convex Optimization—Handling Infeasibility*. Accessed: Apr. 13, 2020. [Online]. Available: https://cvxgen.com/docs/infeasibility.html

STANLEY W. SMITH (Graduate Student Member, IEEE) received the B.S. degree in electrical engineering from the University of Michigan, Ann Arbor, in 2016. He is currently pursuing the Ph.D. degree in electrical engineering and computer sciences with the University of California at Berkeley, Berkeley. His research interests include the control of multi-agent systems using formal methods and optimization-based approaches, with a focus on applications in transportation. He was a recipient of the National Defense Science and Engineering Graduate Research Fellowship.

YEOJUN KIM received the B.S. degree from The University of Texas at Austin, Austin, TX, USA, in 2015, and the M.S. degree in mechanical engineering from the University of California at Berkeley, Berkeley, CA, USA, in 2019, where he is currently pursuing the Ph.D. degree in mechanical engineering. His current research interests include eco-friendly adaptive cruise control, the optimal control for the decentralized platoon systems, and the robust control and predictive control.

JACOPO GUANETTI received the B.Sc. and M.Sc. degrees in automation engineering and the Ph.D. degree in systems and control engineering from the Politecnico di Milano, in 2009, 2012, and 2015, respectively. In 2012, he was a Visiting Scholar at The Ohio State University (Center for Automotive Research), Columbus, OH, USA. From 2016 to 2019, he was a Postdoctoral Researcher at the University of California at Berkeley, Berkeley, CA, USA. Since 2019, he has been at WideSense, a spinoff company focused on mobility contextual intelligence. His research interests include optimal and predictive control and their applications to connected and automated vehicles, chassis and powertrain controls, and energy systems.

RUOLIN LI received the M.S. degree in mechanical engineering from the University of California at Berkeley, Berkeley, in 2018, where she is currently pursuing the Ph.D. degree majoring in controls with the Department of Mechanical Engineering. Her current research interests include controls, game theory, and intelligent transportation systems.

ROYA FIROOZI (Graduate Student Member, IEEE) received the bachelor's degree in mechanical engineering from the University of California at Berkeley, Berkeley, in 2014, where she is currently pursuing the Ph.D. degree in controls with the Mechanical Engineering Department. Her research interests include predictive control, optimization, and robotics.

BRUCE WOOTTON received the B.S. degree in electrical engineering from Cornell University and the M.S. degree in computer engineering from Santa Clara University. For the past 20 years, he has helped launch multiple new technologies and companies in the networking, security, and IoT space. He has been the Technical Director of Software at five software startups, including IP Dynamics, Lookout, and Vontu. He has also served as a Distinguished Engineer at Symantec Inc. Prior to co-founding AV-Connect Inc., he built and led the Software Development T.eam, Brightbox Technologies, where he led the architecture and design and managed the implementation of a cloud-based, optimized, supervisory control service for large-scale HVAC systems. The company was acquired by NEXTracker, the largest supplier of single-axis trackers for utility-scale solar power plants. He led the development team that applied the Brightbox technology to the autonomous control of solar trackers optimized to increase yield. TrueCapture, the resulting product has been a huge commercial success increasing power production by 2%–4%. It has been deployed to hundreds of thousands of individually controlled solar trackers. He is currently the Chief Technology Officer of Widesense. He has been awarded 13 U.S. patents in the network, security, and controls space.

ALEXANDER A. KURZHANSKIY received the Ph.D. degree in control and communication systems from the Department of Electrical Engineering and Computer Science, University of California at Berkeley (UC Berkeley), in 2007. Then, he joined California Partners for Advanced Transportation Technology (PATH), UC Berkeley, as a Postdoctoral Researcher, where he has been a Research Engineer, since 2011. He specializes in large-scale transportation data analysis, modeling and simulation of traffic flows and individual vehicles, traffic control, vehicle automation, and intersection safety.

FRANCESCO BORRELLI received the Laurea degree in computer science engineering from the University of Naples "Federico II," Italy, in 1998, and the Ph.D. degree from the Automatic Control Laboratory, ETH Zurich, Switzerland, in 2002.

He is currently a Professor at the Department of Mechanical Engineering, University of California at Berkeley, USA. Since 2004, he has been serving as a Consultant for major international corporations. He was the Founder and the CTO of Bright-Box Technologies Inc., a company focused on cloud-computing optimization for autonomous systems. He is the Co-Director of the Hyundai Center of Excellence in Integrated Vehicle Safety Systems and Control, UC Berkeley. He has authored more than 150 publications in the field of predictive control. He has authored the book *Predictive Control* (Cambridge University Press). His research interests include model-predictive control and its application to automated driving and energy systems. He was a recipient of the 2009 NSF CAREER Award and the 2012 IEEE Control System Technology Award. In 2017, he received the Industrial Achievement Award from the International Federation of Automatic Control (IFAC) Council.

ROBERTO HOROWITZ (Senior Member, IEEE) received the B.S. degree (Hons.) and the Ph.D. degree in mechanical engineering from the University of California at Berkeley (UC Berkeley), in 1978 and 1983, respectively. He became a Faculty Member of the Mechanical Engineering Department, UC Berkeley, in 1982. He is currently a Professor with the Department of Mechanical Engineering, UC Berkeley, and also holds the James Fife Endowed Chair at the College of Engineering. He teaches and conducts research in the areas of adaptive, learning, nonlinear, and optimal control, with applications to micro-electromechanical systems (MEMS), computer disk file systems, robotics, mechatronics, and intelligent vehicle and highway systems (IVHS). He is also the Chair of the Department of Mechanical Engineering and a former Co-Director of the Partners for Advanced Transportation Technology (PATH) Research Center, UC Berkeley. He is a member of ASME. He was a recipient of the 2010 ASME Dynamic Systems and Control Division (DSCD) Henry M. Paynter Outstanding Investigator Award and the ASME 2018 Rufus Oldenburger Medal in recognition of his pioneering and impactful contributions to control applications in mechatronics, magnetic data storage, and traffic systems.

MURAT ARCAK (Fellow, IEEE) received the B.S. degree in electrical engineering from Bogazici University, Istanbul, Turkey, in 1996, and the M.S. and Ph.D. degrees from the University of California at Santa Barbara, Santa Barbara, in 1997 and 2000, respectively. He is currently a Professor at the Electrical Engineering and Computer Sciences Department, University of California at Berkeley. His research interests include dynamical systems and control theory with applications to synthetic biology, multi-agent systems, and transportation. He is a Fellow of the IFAC. He received the CAREER Award from the National Science Foundation, in 2003, the Donald P. Eckman Award from the American Automatic Control Council, in 2006, the Control and Systems Theory Prize from the Society for Industrial and Applied Mathematics (SIAM), in 2007, and the Antonio Ruberti Young Researcher Prize from the IEEE Control Systems Society, in 2014.

• • •