Energy-Efficient Reactive and Predictive Connected Cruise Control

Minghao Shen , *Graduate Student Member, IEEE*, Robert Austin Dollar , *Member, IEEE*,
Tamas G. Molnar , *Member, IEEE*, Chaozhe R. He , *Member, IEEE*, Ardalan Vahidi , *Senior Member, IEEE*,
and Gábor Orosz , *Senior Member, IEEE*

Abstract—**Connected and automated vehicles (CAVs) have shown great potential in improving the energy efficiency of road transportation. Energy savings, however, greatly depends on driving behavior. Therefore, the controllers of CAVs must be carefully designed to fully leverage the benefits of connectivity and automation, especially if CAVs travel amongst other non-connected and human-driven vehicles. With this as motivation, we introduce a framework for the longitudinal control of CAVs traveling in mixed traffic including connected and non-connected human-driven vehicles. Reactive and predictive connected cruise control strategies are proposed. Reactive controllers are given by explicit feedback control laws. Predictive controllers, on the other hand, optimize the control input in a receding-horizon fashion, by predicting the motions of preceding vehicles. Beyond-line-of-sight information obtained via vehicle-to-vehicle (V2V) communication is leveraged by the proposed reactive and predictive controllers. Simulations utilizing real traffic data show that connectivity can bring up to 30% energy savings in certain scenarios.**

Index Terms—**Connected automated vehicles, V2X connectivity, MPC, traffic flow models.**

I. INTRODUCTION

ENERGY efficiency of vehicles is an everlasting topic in the auto industry, since improving energy efficiency can bring great financial and societal benefits [1]. Driving profiles play an important role in the energy consumption: with the same vehicle traveling on the same route, different drivers may have different driving profiles, which results in great difference in the energy consumption. This shows great potential for improving energy efficiency by optimizing driving profiles.

While human drivers have large variations in their driving behavior [2], which may undermine the energy efficiency, vehicle automation eliminates such variation and provides a more accurate and consistent way to improve energy efficiency. SAE categorizes automated vehicles into 6 levels (0-5) [3]. Since energy consumption is mainly related to longitudinal motion, level 1 or 2 automation can already provide significant energy savings. On one hand, automated vehicles (AVs) may optimize their speed profiles over a horizon, taking into consideration the engine and transmission dynamics, and the road elevation [4]. On the other hand, extensive research has focused on optimizing the control input (pedal, brake and gear shift) to follow optimal driving cycles [5]. However, these two methods do not take traffic into consideration. In real traffic, vehicles may not be able to follow pre-defined ideal trajectories.

With level 1 or 2 automation, AVs rely on *adaptive cruise control (ACC)* algorithms to react to the motion of the preceding vehicle in traffic. The controller design usually falls into one of two categories: *reactive controller* or *predictive controller*. Reactive ACC (RACC) has explicit feedback control laws that are usually parameterized, so that the controller parameters can be optimized for energy efficiency while ensuring other specifications such as stability. On the other hand, predictive ACC (PACC) can directly optimize the future trajectory based on the predicted future motions of neighboring vehicles. While predictions may significantly improve energy efficiency, it is hard to achieve high prediction accuracy, since the motions of neighboring vehicles can be highly correlated or completely stochastic.

Vehicle-to-vehicle (V2V) communication can potentially resolve this problem. Peer-to-peer communication enables connected vehicles to share information for prediction and control, and facilitates cooperation among vehicles in traffic. SAE categorized cooperative driving automation (CDA) into status-sharing, intent-sharing, agreement-seeking, and prescriptive cooperation [6]. Many of the existing research works assume high penetration of connectivity in the traffic which may allow automated vehicles to cooperate with each other. Cooperative adaptive cruise control (CACC) is a typical example.

Similar to ACC, CACC designs can also be categorized into reactive and predictive control [7], [8]. Reactive control tries to synchronize the speed of the platoon, guaranteeing string stability and maintaining desirable headway [9], [10], [11]. On the other hand, predictive controllers have access to the future

motion plans of leading vehicles, therefore coordinated and even global optimization becomes possible [12], [13], [14], [15], [16]. To make these systems more scalable, distributed control protocol has also been studied [17]. Research has shown that CACC and platooning bring significant energy benefits under various traffic scales, traffic compositions, and demand patterns [18], [19], [20], [21], [22].

However, current V2V technology is far from being widely deployed. The assumption of high penetration of connectivity and high level of cooperation is hard to realize in practice in the near future. In order to boost the deployment of connectivity, its benefits first need to be demonstrated at low penetration rates and with low level of cooperation. Hence, the near future of transportation is more likely to evolve into mixed traffic, where potentially four kinds of vehicles may participate: human-driven vehicles (HV), connected human-driven vehicles (CHV), automated vehicles (AV) and connected and automated vehicles (CAV).

While AVs without connectivity may execute adaptive cruise control, connectivity enables CAVs to utilize controllers with higher performance, even with low level of cooperation such as status-sharing protocol. CAV controllers that operate in mixed traffic consisting of connected and non-connected vehicles are referred to as *connected cruise control (CCC)*. In CCC, CAVs have access to beyond-line-of-sight information of CHVs and CAVs in the distance, which is incorporated into the controller design. Only low-level cooperation such as status-sharing is assumed and centralized control is not possible.

Similar to ACC and CACC, CCC can also be categorized into reactive and predictive control. Reactive CCC (RCCC) takes the V2V information from leading vehicles as reference signals, and its objective is to synchronize the speed in the traffic for string stability and smooth driving [23], [24]. Meanwhile, predictive CCC (PCCC) can incorporate the information of preceding vehicles to make predictions on the motion of the vehicle immediately in the front [25], [26]. This may significantly improve predictions, and enable optimized planning of motions in advance, which may reduce speed variations and save energy. In Fig. 1 the concepts of RACC, PACC, RCCC, and PCCC are summarized by illustrations for mixed traffic scenarios containing HVs, AVs, CHVs, and CAVs.

With all these distinctions made, there is a need to integrate these methods into a unified framework. This article thus presents the following contributions:

- We establish a control design framework for the reactive and predictive longitudinal control of connected automated vehicles (CAVs) driving in mixed traffic including connected and non-connected vehicles.
- We propose a predictive connected cruise control strategy, in which we introduce algorithms to estimate the number of hidden non-connected vehicles, and predict the future motion of the vehicles preceding the CAV. These estimation and prediction algorithms play a key role in enhancing the performance of the proposed predictive controller.
- We show significant energy benefits provided by V2V connectivity for both the reactive and predictive controllers and explain these energy savings by comparing simulated trajectories.
- We conduct extensive numerical simulations, in which the motion of the CAV is simulated while the trajectories of the preceding human-driven vehicles are given by
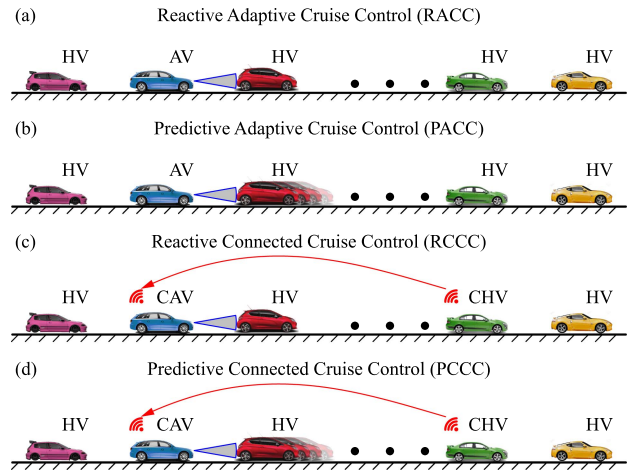

Fig. 1. Illustration of longitudinal control strategies for automated vehicles (AVs) and connected automated vehicles (CAVs) traveling in mixed traffic that includes human-driven vehicles (HVs) and connected human-driven vehicles (CHVs). Predictive controllers rely on the predictions of the future motions of preceding vehicles, as is shown in shadowed vehicles.
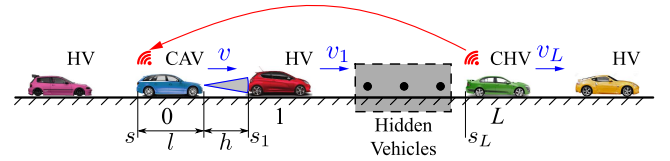
Fig. 2. Connected cruise control in mixed traffic consisting of connected and non-connected vehicles.

experimental data. We compare the performance of the reactive and predictive controllers in three typical traffic scenarios and show the benefit of predictive controllers in terms of energy efficiency.

The remainder of this article is organized as follows. Section II introduces the problem setting and describes the longitudinal dynamics of vehicles. Section III discusses the design of energy-efficient reactive controllers including RACC and RCCC. Section IV discusses predictive controller designs including PACC and PCCC. Section V shows the energy benefits of different controller designs with low market penetration of connected vehicles. Section VI concludes this article and points out future research directions.

II. VEHICLE DYNAMICS

In this section, we introduce the problem setup, and derive the state space model for longitudinal controller design. Consider the connected cruise control scenario in Fig. 2, in which a connected and automated vehicle (CAV), called ego vehicle, is driving on a flat road without elevation change, with the intention to follow human-driven traffic. In this article, we focus on designing a longitudinal controller for the ego vehicle. Therefore, we describe the car-following scenario from the perspective of the ego vehicle: we number the ego vehicle as vehicle 0, and the preceding vehicles are numbered sequentially as $1, 2, \ldots L$, i.e., with the index increasing in the direction of motion.

Fig. 3. Nonlinear functions in the vehicle dynamics and optimal velocity model (OVM). (a) Saturation function (4). (b) Acceleration limits (5). (c) Range policy (11). (d) Speed policy (12).

The longitudinal dynamics of the CAV with respect to its position s and velocity v can be modeled as in [27]:

$$\dot{s} = v,$$
$$\dot{v} = -\frac{1}{m_{\text{eff}}}\left(mg\xi + kv^2\right) + \frac{T_w}{m_{\text{eff}}R}. \quad (1)$$

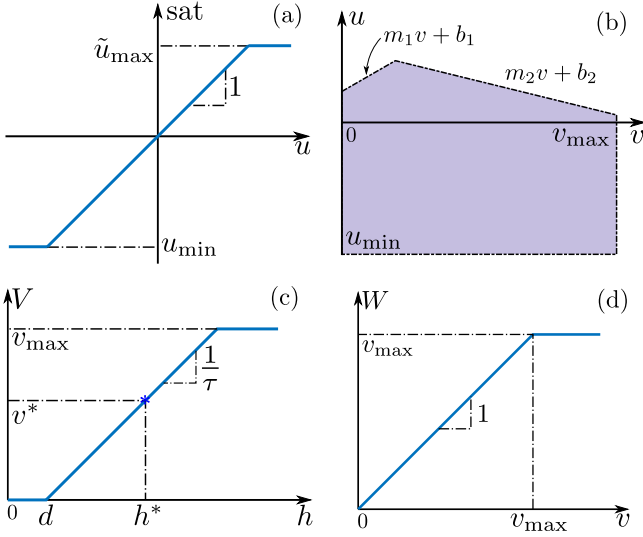Here the effective mass $m_{\text{eff}} = m + I/R^2$ incorporates the mass m, mass moment of inertia I, and the radius R of the wheels. Moreover, g is the gravitational constant, ξ denotes the rolling resistance coefficient and k denotes the air resistance coefficient. We can control the vehicle speed by applying the torque T_w on the wheels using the engine/electric motors and the brakes. To highlight how control actions influence the system, we consider the commanded acceleration as control input u and rewrite (1) as

$$\dot{s}(t) = v(t),$$
$$\dot{v}(t) = -f\left(v(t)\right) + \text{sat}\left(u(t - \sigma)\right), \quad (2)$$

where

$$f(v) = -\frac{1}{m_{\text{eff}}}\left(mg\xi + kv^2\right), \quad \text{sat}\left(u(t - \sigma)\right) = \frac{T_w(t)}{m_{\text{eff}}R}. \quad (3)$$

The model incorporates the delay σ in the powertrain system, and the saturation $\text{sat}(\cdot)$ arising from limitations of engine/motor power, engine/motor torque, and braking capability. More specifically, the saturation is modeled as

$$\text{sat}(u) = \min\left\{\tilde{u}_{\max}, \max\{u_{\min}, u\}\right\}, \quad (4)$$
$$\tilde{u}_{\max} = \min\left\{u_{\max}, m_1 v + b_1, m_2 v + b_2\right\}, \quad (5)$$

as is shown in Fig. 3(a) and (b). Here $u_{\min}$ is the minimum acceleration (maximum deceleration) due to the braking capability, and m_1, m_2, b_1, b_2 are determined by engine torque limit and power limit.

In order to follow the desired acceleration, a_d, the control action

$$u(t) = \tilde{f}\left(v(t)\right) + a_d(t), \quad (6)$$

TABLE I
COMPARISON OF CRUISE CONTROL ALGORITHMS

	Explicit Law	Connectivity	Prediction
RACC	Yes	No	No
RCCC	Yes	Yes	No
PACC	No	No	Constant Speed
PCCC	No	Yes	Car-following Model

is applied, where the term $\tilde{f}$ tries to compensate the nonlinear physical effects f in (3). To further simplify the dynamics and focus on the choice of desired acceleration a_d, we assume that perfect compensation is achieved by low-level controllers. This simplifies the dynamics (2) to

$$\dot{s}(t) = v(t),$$
$$\dot{v}(t) = \text{sat}\left(a_d(t - \sigma)\right). \quad (7)$$

To characterize the performance of longitudinal control, energy consumption is the main interest in this article. It is evaluated with energy consumption per unit mass

$$w = \int_{t_0}^{t_f} v(t) g\left(\dot{v}(t) + f(v(t))\right) \mathrm{d}t, \quad (8)$$

where $g(x) = \max\{x, 0\}$ implies that braking does not consume or recover energy. We remark that the effects of energy recovering systems can be included by choosing different g functions, but this is beyond the scope of this article.

In what follows, we investigate the energy efficiency of four types of controllers: RACC, RCCC, PACC, and PCCC; as summarized by Table I. These four controllers are detailed in the next two sections and in Algorithms 1–4.

III. REACTIVE CONTROLLERS

In this section, we design control algorithms for reactive adaptive cruise control (RACC) and reactive connected cruise control (RCCC). We start with the simple RACC case, where an automated vehicle is controlled and there is no connected vehicle in the traffic, as is shown in Fig. 1(a). With on-board sensors such as camera, lidar, or radar, the ego vehicle can respond to the vehicle immediately in the front. RACC determines the desired acceleration a_d as a function of the distance headway h, its speed v, as well as the speed v_1 of the vehicle immediately in the front:

$$a_d = F(h, v, v_1), \quad (9)$$

where $h = s_1 - s - l$ is related to the positions s and s_1 of the vehicles and the length l of the ego vehicle, as is shown in Fig. 2. Longitudinal controllers are usually constructed analogously to car-following models. For example, the optimal velocity model (OVM) yields the control algorithm [28].

$$F^{\text{OVM}}(h, v, v_1) = \alpha\left(V(h) - v\right) + \beta\left(W(v_1) - v\right), \quad (10)$$

where the range policy $V(h)$ determines the desired velocity as a function of the distance headway h. A common choice of range policy is given in [23] as

$$V(h) = \min\left\{v_{\max}, \max\left\{0, (h - d)/\tau\right\}\right\}. \quad (11)$$

As is shown in Fig. 3(c), when the distance headway is less than the stopping distance d, the ego vehicle tends to stay still, while when the distance headway is larger than $d + \tau v_{\max}$, the

ego vehicle intends to travel with maximum speed $v_{\max}$ without being influenced by the preceding vehicle. The desired velocity grows with constant gradient $1/\tau$ where τ is referred to as *time headway*. Note that in the context of traffic flow theory, time headway is usually defined as the time between two vehicles passing the same location. In the context of car-following, here the time headway τ is interpreted as the time it takes the ego vehicle to move to the current position of vehicle 1 with current speed. So τ specifies the trade-off between safety and traffic efficiency. Moreover, the speed policy

$$W(v_1) = \min \{v_{\max}, v_1\} \tag{12}$$

is used to prevent the ego vehicle from speeding once the preceding vehicle goes faster than $v_{\max}$; see Fig. 3(d).

Another widely-used car-following model is the intelligent driver model (IDM) [29], [30]:

$$F^{\mathrm{IDM}}(h, v, v_1) = a_0 \left(1 - \left(\frac{v}{v_{\max}} \right)^\delta - \left(\frac{H(v, v_1)}{h} \right)^2 \right) \tag{13}$$

where the desired distance headway is calculated using the range policy

$$H(v, v_1) = d + \max \left\{ 0, \tau v - \frac{v(v_1 - v)}{\sqrt{a_0 b_0}} \right\}. \tag{14}$$

Here a_0 corresponds to the maximum acceleration, b_0 is the deceleration coefficient, and τ is the desired time headway. Note that at steady state ($v_1 = v$), the last term is eliminated and we obtain the simplified range policy

$$H(v) = d + \tau v, \tag{15}$$

which is the inverse of $V(h)$ in (11) when $0 < v < v_{\max}$. The stopping distance d and maximum speed $v_{\max}$ have the same meaning as those in (11). The OVM-based controller has been applied on real automated vehicles and tested extensively in various scenarios [31], [32].

When V2V connectivity is available, connected and automated vehicles may rely on information from connected human-driven vehicles and execute *reactive connected cruise control (RCCC)*; see Fig. 1(c). In this case, the ego vehicle not only reacts to the vehicle immediately in the front, but also to the vehicles beyond its line of sight:

$$a_{\mathrm{d}} = F\left(h, v, \{s_i\}_{i\in\mathcal{I}}.\{v_i\}_{i\in\mathcal{I}}\right), \tag{16}$$

Here $\mathcal{I}$ denotes the set of all the vehicles that are connected to or sensed by the ego vehicle, so $1 \in \mathcal{I}$ because vehicle 1 can be sensed by the ego vehicle using onboard sensors even if it is not connected. For example, one can extend OVM (10) to the RCCC controller

$$F(h, v, \{v_i\}_{i\in\mathcal{I}}) = \alpha \left(V(h) - v\right) + \sum_{i\in\mathcal{I}} \beta_i \left(W(v_i) - v\right). \tag{17}$$

Notice that the signals $\{v_i\}_{i\in\mathcal{I}}$ are reference signals. The objective of our control design is to minimize speed variation while maintaining a reasonable distance headway. The controller does not necessarily need to respond to every reference signal immediately. Instead, the controller may wait before responding to the signals from vehicles in the distance [33]. Thus, (17) can

---

Algorithm 1: Reactive Adaptive Cruise Control (RACC).

1   **for** $t = 0$ **to** $T_{\max}$ **do**
2   Observe and store $s_1(t)$, $v_1(t)$;
3   Calculate desired acceleration a_{d} using (10);
4   Apply control command $u = \tilde{f}(v) + a_{\mathrm{d}}$;
5   **end**

Algorithm 2: Reactive Connected Cruise Control (RCCC).

1   **for** $t = 0$ **to** $T_{\max}$ **do**
2   Observe and store $s_1(t)$, $v_i(t)$, $i \in \mathcal{I}$;
3   Calculate desired acceleration a_{d} using (18);
4   Apply control command $u = \tilde{f}(v) + a_{\mathrm{d}}$;
5   **end**

---

be generalized to

$$F(t) = \alpha \left(V(h(t)) - v(t)\right) + \sum_{i\in\mathcal{I}} \beta_i \left(W(v_i(t - \sigma_i)) - v(t)\right), \tag{18}$$

where the delays σ_i are additional design parameters, as opposed to σ in (7) which is a result of powertrain dynamics. In this article, we refer to this as *reactive connected cruise control (RCCC)*. Notice that RACC is essentially a special case of RCCC, where $\beta_i = 0$ for all $i \in \mathcal{I} \setminus \{1\}$. The RACC and RCCC algorithms are summarized in Algorithms 1 and 2, respectively. There has been extensive research on the choice of energy-optimal controller parameters of RACC and RCCC; we refer to [33] for more details.

We remark that safety is often specified as maintaining larger distance headway than a velocity-dependent minimum distance headway, such as $H_{\min}(v) = d_{\min} + \tau_{\min} v$, cf. (15). In order to ensure that a reactive controller is safe, one may use so-called control barrier functions to modify the inputs of RACC and RCCC and make them safe [34]. The implementation of control barrier functions on real trucks is described in [32]. Such controller modifications are beyond the scope of this article and are not discussed here in detail.

IV. PREDICTIVE CONTROLLERS

Apart from constructing explicit reactive control laws, one may also formulate control synthesis as an optimization problem in which safety, energy efficiency, and traffic efficiency are considered simultaneously in the objective functions and constraints. Model predictive control (MPC) is a prevailing choice, which relies on predicting the motion of the vehicle in front of the ego vehicle and choosing optimal action based on the prediction. An accurate prediction is crucial, for both predictive adaptive cruise control (PACC) illustrated in Fig. 1(b) and predictive connected cruise control (PCCC) in Fig. 1(d). The choice of predictor significantly influences the energy efficiency and safety of predictive controllers. With V2V communication, the CAV gets access to information from vehicles ahead in the distance, which enables better prediction of the vehicle immediately in the front. Despite the change of connectivity structure, the general optimization formulation remains the same for both PACC and PCCC, only the estimated future position $\hat{s}_1$ and velocity $\hat{v}_1$ of the preceding vehicle differs.

Let $x = [s, v]^\top$ represent the state of the ego vehicle and $x_1 = [s_1, v_1]^\top$ represent the state of the vehicle immediately in the front. At time t, the controller can be formulated as the continuous-time optimization problem

$$\min \int_t^{t+T} \ell\left(x\left(\tilde{t}|t\right), \hat{x}_1\left(\tilde{t}|t\right), a\left(\tilde{t}|t\right)\right) \mathrm{d}\tilde{t},$$

$$\text{s.t. } G_{\text{dynamics}}\left(x\left(\tilde{t}|t\right), a\left(\tilde{t}|t\right)\right) = 0,$$

$$G_{\text{safety}}\left(x\left(\tilde{t}|t\right), \hat{x}_1\left(\tilde{t}|t\right)\right) \leq 0,$$

$$G_{\text{saturation}}\left(x\left(\tilde{t}|t\right), a\left(\tilde{t}|t\right)\right) \leq 0,$$

$$x(t|t) = x(t),$$

$$a\left(\tilde{t}|t\right) = a_{\text{d}}\left(\tilde{t} - \sigma\right), \quad \forall \tilde{t} \in [t, t+\sigma]. \quad (19)$$

That is, under safety constraints G_{safety} and saturation $G_{\text{saturation}}$, we aim to minimize the cumulative cost function ℓ during the time interval $[t, t+T]$ based on our knowledge of the system behavior $x(\tilde{t}|t)$, and our prediction on the future motion of preceding vehicle $\hat{x}_1(\tilde{t}|t)$. Due to the powertrain delay, when $\tilde{t} \leq t+\sigma$, the acceleration a at time $\tilde{t}$ is determined by the desired acceleration a_{d} at $\tilde{t} - \sigma$.

The objective function of MPC includes penalties on the distance headway and the control input. Similar to the range policies (11), (14), (15) used in the reactive controllers, the predictive controller aims to keep a desirable distance headway as a function of velocity. For example, in this article, we aim to keep a constant time headway τ and thus we utilize (15). As shown below, the MPC controller typically applies a quadratic penalty on the deviation of distance headway from desirable values. In addition, we will also penalize the magnitude of the control input as given below.

While the optimization problem (19) is defined in continuous time, for efficient implementation, we usually need to convert it to discrete-time optimization. We first transform the dynamics (7) into discrete time using the time step Δt. To make the final MPC a convex quadratic programming (QP) problem, we drop the nonlinear terms and move the saturation function to inequality constraints. Thus, the equality constraints are given by the linear dynamics

$$s(k+1) = s(k) + \Delta t\, v(k) + \frac{1}{2}\Delta t^2 a(k),$$

$$v(k+1) = v(k) + \Delta t\, a(k), \quad (20)$$

while the saturation function (4) is transformed to the inequality constraints

$$a(k) \geq u_{\min}, \quad a(k) \leq m_1 v(k) + b_1, \quad a(k) \leq m_2 v(k) + b_2. \quad (21)$$

Due to powertrain delay σ, the acceleration at the current time is determined by the control input in the past, that is,

$$a(k) = a_{\text{d}}(k - q), \quad (22)$$

where $\sigma = q\Delta t$. In addition, we define a minimum distance headway to guarantee safety

$$H_{\min}(v) = d_{\min} + \tau_{\min} v. \quad (23)$$

In order to compensate for the prediction uncertainty, we impose additional safety margin $d_{\text{margin}}(k)$ at each time k in the prediction horizon,

$$\hat{h}(k) - H_{\min}\left(v(k)\right) - d_{\text{margin}}(k) \geq 0, \quad (24)$$

where $\hat{h}(k) = \hat{s}_1(k) - s(k) - l$ is the estimated distance headway. The safety margin is elaborated in Appendix A.

In summary, the MPC controller is formulated as follows

$$\min_{\substack{a(0|t), \\ \vdots \\ a(T-1+q|t),}} q_{\text{g}} \sum_{k=0}^{T}\left(\hat{h}(k|t) - H\left(v(k|t)\right)\right)^2 + q_{\text{a}} \sum_{k=0}^{T-1} a^2(k|t) + q_\epsilon \epsilon,$$

$$\text{s.t. } s(i+1|t) = s(i|t) + \Delta t v(i|t) + \frac{1}{2}\Delta t^2 a(i|t),$$

$$v(i+1|t) = v(i|t) + \Delta t a(i|t),$$

$$\forall i = 0, \ldots, T-1,$$

$$\hat{h}(j|t) = \hat{s}_1(j|t) - s(j|t) - l,$$

$$\hat{h}(j|t) - H_{\min}\left(v(j|t)\right) - d_{\text{margin}}(j) \geq -\epsilon,$$

$$0 \leq v(j|t) \leq v_{\max},$$

$$u_{\min} \leq a(j|t),$$

$$\forall j = 0, \ldots, T,$$

$$a(k+q|t) \leq m_1 v(k+q|t) + b_1,$$

$$a(k+q|t) \leq m_2 v(k+q|t) + b_2,$$

$$\forall k = 0, \ldots, T-1,$$

$$s_1(0|t) = s_1(t), \, s(0|t) = s(t), \, v(0|t) = v(t),$$

$$a(l|t) = a_{\text{d}}(t+l-q), \qquad \forall l = 0, \ldots, q-1, \quad (25)$$

where, by abuse of notation, t and T represent discrete time and we apply soft constraint with $\epsilon \geq 0$ in the safety inequality (24) to ensure feasibility. We highlight that we included the powertrain delay in the controller design, which although rarely considered in the MPC literature, has a significant impact on the performance. We remark that predictive controllers are naturally equipped with delay compensation, which requires a more complex design than the described reactive controllers [35], [36].

A. Predictive Adaptive Cruise Control (PACC)

Accurate prediction of the motion of preceding vehicles is the key to the success of predictive controllers. In PACC, no extra information on the preceding vehicle is available from V2V connectivity. It is a common choice to assume that the preceding vehicle maintains its current speed in the future [37]:

$$\hat{s}_1\left(\tilde{t}|t\right) = s_1(t) + \left(\tilde{t} - t\right) v(t), \quad \tilde{t} \geq t,$$

$$\hat{v}_1\left(\tilde{t}|t\right) = v_1(t), \qquad \tilde{t} \geq t. \quad (26)$$

The corresponding PACC algorithm, which is in discrete time, is shown in Algorithm 3.

Algorithm 3: Predictive Adaptive Cruise Control (PACC).

1 **for** $t = 0$ **to** $T_{\max}$ **do**
2   Observe and store $s_1(t)$, $v_1(t)$;
3   Apply constant speed prediction on leading vehicle 1

$$\hat{s}_1(t+k|t) = s_1(t) + k\Delta t \, v_1(t) \, , \quad k = 1, \ldots, T \, ,$$

$$\hat{v}_1(t+k|t) = v_1(t) \, , \qquad\qquad k = 1, \ldots, T \, ,$$

4   Solve the optimization problem (25) using $\hat{s}_1(t : t + T|t)$;
5   Output desired acceleration $a_{\mathrm{d}}(t)$ to powertrain and braking systems;
6   Update with discrete time dynamics (20), (22);
7 **end**

Algorithm 4: Predictive Connected Cruise Control (PCCC).

1 **for** $t = 0$ **to** $T_{\max}$ **do**
2   Observe and store $s_1(t)$, $v_1(t)$, $s_L(t)$, $v_L(t)$;
3   Estimate $\hat{n}_{\mathrm{h}}$;
4   Apply constant speed prediction on leading vehicle L

$$\hat{s}_L(t+k|t) = s_L(t) + k\Delta t \, v_L(t) \, , \quad k = 1, \ldots, T \, ,$$

$$\hat{v}_L(t+k|t) = v_L(t) \, , \qquad\qquad k = 1, \ldots, T \, ,$$

5   **if** $\hat{n}_{\mathrm{h}} == 0$ **then**
6   Simulate $\hat{s}_1(t : t + T|t)$ following the vehicle numbered L;
7   **else**
8   **for** $i = 0$ **to** $\hat{n}_{\mathrm{h}}$ **do**
9   Simulate $\hat{s}_{L-i-1}(t : t + T|t)$ following vehicle $L - i$ using uniform flow initialization;
10   **end**
11   Simulate $\hat{s}_1(t : t + T|t)$ following vehicle 2 using real data initialization;
12   **end**
13   Solve the optimization problem (25);
14   Output desired acceleration $a_{\mathrm{d}}(t)$ to the powertrain and braking systems;
15   Update with discrete time dynamics (20), (22);
16 **end**

B. Predictive Connected Cruise Control (PCCC)

In this section, we introduce predictive connected cruise control which utilizes V2V connectivity when available. We consider the scenario in Fig. 1(d) where a CAV executes PCCC. We assume lean penetration of connectivity where only a single lead CHV (vehicle L) is connected to the CAV while the preceding vehicle (vehicle 1) is sensed by on-board sensors.

We propose the PCCC control framework detailed in Algorithm 4. Compared to the PACC in Algorithm 3, the only change is in the way we predict $\hat{s}_1(t + k|t)$. In PACC, without additional information, we assume a constant speed. While in PCCC, with the additional information from V2V communication, we can

potentially make more accurate predictions on the future motion of preceding vehicles. In this article, we assume a constant speed of the connected vehicle in the distance, then we simulate the motion of subsequent vehicles, until reaching the vehicle immediately ahead. In traffic with lean penetration of connected vehicles, the number $\hat{n}_{\mathrm{h}}$ of hidden vehicles driving between the vehicle immediately ahead (vehicle 1) and the connected vehicle in the distance (vehicle L) is unknown; see Fig. 2. In the next subsection, we introduce an algorithm to estimate the number of hidden vehicles. We summarize the four controllers discussed in Algorithm 1 to 4 in Table I.

C. Hidden Vehicle Estimation

In mixed traffic with lean penetration of connected vehicles, the number n_{h} of the hidden vehicles is not directly available. Existing literature applies range policy-based algorithms to identify n_{h} [38], [39]. In this paper, we propose an estimation algorithm based on historical data of preceding vehicles: the history of s_1, v_1, s_L, and v_L, which are recorded during driving. We denote the resulting estimation by $\hat{n}_{\mathrm{h}}$ that approximates the unknown number of hidden vehicles $n_{\mathrm{h}} = L - 2$. Note that, in this paper, we focus on the scenario where there is only one connected vehicle in the downstream traffic. When connecting to multiple vehicles, we can still apply Algorithm 5 to estimate the number of hidden vehicles.

The detailed algorithm design is shown in Algorithm 5. Our estimation algorithm conducts a brute-force search for every possible number of hidden vehicles n_{h}. At the time t for a given n_{h}, we consecutively simulate the motion of hidden vehicles over the past $[t - T_{\mathrm{h}}, t]$ using the IDM model (13), (14), to obtain an estimation $\hat{s}_1^{(n_{\mathrm{h}})}$ of the position s_1 of the preceding vehicle. Then we compare the simulated $\hat{s}_1^{(n_{\mathrm{h}})}$ with the recorded measurements of s_1. The n_{h} corresponding to the minimum error is chosen.

There are a few details worth mentioning. First, we initialize the state for simulation such that at time $t - T_{\mathrm{h}}$, the hidden vehicles are equally spaced with distances

$$\hat{s}_{i+1}^{(n_{\mathrm{h}})}(t - T_{\mathrm{h}}) - \hat{s}_i^{(n_{\mathrm{h}})}(t - T_{\mathrm{h}}) = \frac{s_L(t - T_{\mathrm{h}}) - s_1(t - T_{\mathrm{h}})}{n_{\mathrm{h}} + 1},$$
$$i = 2, \ldots, n_{\mathrm{h}} + 1, \tag{27}$$

and uniform velocity is used:

$$\hat{v}_i(t - T_{\mathrm{h}}) = \frac{v_1(t - T_{\mathrm{h}}) + v_L(t - T_{\mathrm{h}})}{2}, \quad i = 2, \ldots, n_{\mathrm{h}} + 1. \tag{28}$$

On the other hand, vehicle 1 is initialized with stored observation data $\hat{s}_1^{(n_{\mathrm{h}})}(t - T_{\mathrm{h}}) = s_1(t - T_{\mathrm{h}})$ and $\hat{v}_1^{(n_{\mathrm{h}})}(t - T_{\mathrm{h}}) = v_1(t - T_{\mathrm{h}})$, just as vehicle $L = n_{\mathrm{h}} + 2$, $\hat{s}_{n_{\mathrm{h}}+2}^{(n_{\mathrm{h}})}(t - T_{\mathrm{h}} : t|t) = s_L(t - T_{\mathrm{h}} : t)$ and $\hat{v}_{n_{\mathrm{h}}+2}^{(n_{\mathrm{h}})}(t - T_{\mathrm{h}} : t|t) = v_L(t - T_{\mathrm{h}} : t)$ are for the V2V data s_L, v_L.

Second, at the beginning of the simulation, there is not enough observation data. Thus, we assume that the vehicles are equally spaced around desired time headway, $\hat{h} = H(v_1)$, and thus the initial estimation of the number of hidden vehicles is

$$\hat{n}_{\mathrm{h}}^{\mathrm{init}} = \left\lceil \frac{s_L(t) - s_1(t)}{H(v_1(t))} \right\rceil - 1. \tag{29}$$

Algorithm 5: Estimation of Number of Hidden Vehicles.

Input : History of trajectories $s_1(t - T_h : t)$,
$s_L(t - T_h : t)$, $v_1(t - T_h : t)$, $v_L(t - T_h : t)$;
Current speed $v(t)$;
Previous estimation $\hat{n}_h^{\text{prev}}$;

Output: Estimated number of hidden vehicles $\hat{n}_h$;

1   $n_{h,\max} = \min\left\{\hat{n}_h^{\text{prev}} + 1,\ \left\lceil \frac{s_L(t) - s_1(t)}{h_{\min} + \tau_{\min} v(t)} \right\rceil - 1\right\}$;

2   $n_{h,\min} = \max\left\{\hat{n}_h^{\text{prev}} - 1,\ 0\right\}$;

3   **for** $n_h = n_{h,\min}$ **to** $n_{h,\max}$ **do**

4      $\hat{s}_{n_h+2}^{(n_h)}(t - T_h : t|t) = s_L(t - T_h : t)$ and
$\hat{v}_{n_h+2}^{(n_h)}(t - T_h : t|t) = v_L(t - T_h : t)$;

5   **if** $n_h \neq 0$ **then**

6   **for** $i = 0$ **to** $n_h - 1$ **do**

7   Simulate $\hat{s}_{n_h - i + 1}^{(n_h)}(t - T_h : t|t)$ and
$\hat{v}_{n_h - i + 1}^{(n_h)}(t - T_h : t|t)$ with IDM
model (13), (14) following vehicle
$n_h - i + 2$ using uniform flow
initialization;

8   **end**

9   **end**

10   Simulate $\hat{s}_1^{(n_h)}(t - T_h : t|t)$ with IDM model (13),
(14) following vehicle 2 with real data
initialization;

11    $J(n_h) =$
$c(n_h) \sum_{k=0}^{T_s} \left(s_1(t - k) - \hat{s}_1^{(n_h)}(t - k|t) \right)^2$, where

$$c(n_h) = \begin{cases} 1, & \text{if } n_h = \hat{n}_h^{\text{prev}}, \\ 1.5, & \text{if } n_h \neq \hat{n}_h^{\text{prev}}, \end{cases}$$

12 **end**

13 $\hat{n}_h = \text{argmin}_{n_h \in [n_{h,\min}, n_{h,\max}]} J(n_h)$.

The estimation is adjusted using Algorithm 5 as more data is collected. In the beginning, the amount of data is limited, so we use all available data for estimation. After $t \geq T_h$, we have abundant data from the leading vehicle, so we only use the nearest T_h data. In this paper, we choose $T_h = \min\{t, 23\}$. Moreover, when comparing the trajectory $\hat{s}_1^{(n_h)}$ generated from different n_h, the cost function $J(n_h)$ only compares it to the recorded trajectory s_1 within the nearest T_s time steps. We choose $T_s = \min\{t, 5\}$.

Third, we discourage frequent jumps in the value of n_h. On one hand, we assume that the number of hidden vehicles should not change over time by more than 1, that is only one vehicle can merge in or out of the ego lane at the same time. On the other hand, we put more penalty on values that are different from the previous estimate $\hat{n}_h^{\text{prev}}$ by $c(n_h)$, so $\hat{n}^{\text{prev}}$ is more likely to have smaller cost function and be chosen in the current step.

V. NUMERICAL RESULTS

In this section, we conduct numerical simulations to compare the four kinds of control algorithms. We demonstrate the benefit of connectivity, prediction, and the generalizability of our algorithms.
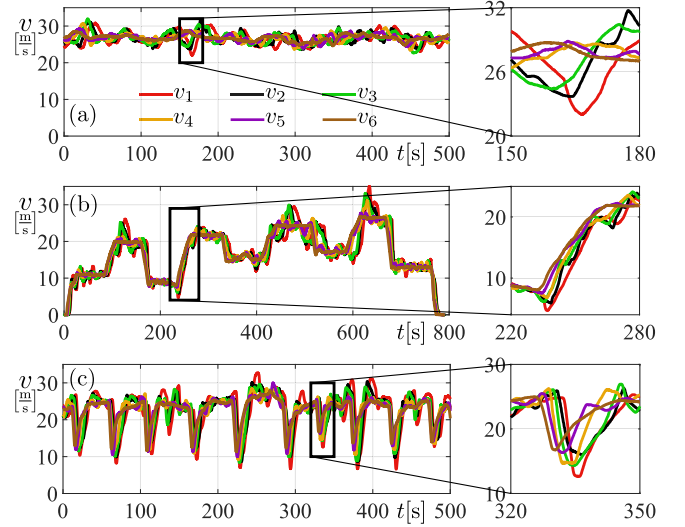

Fig. 4. Three qualitatively different speed trajectories from experimental data [31]. (a) Free flow profile. (b) Step profile. (c) Congested profile.

A. Simulation Setup

First, we introduce the basic setup of our simulation. As is shown in Fig. 1, the ego vehicle follows a chain of human-driven vehicles. In this paper, we assign real human driving data to the speed trajectories of preceding vehicles. The real data were collected in an experiment where all vehicles were connected. The details of the experiment are described in [31].

In this paper, we consider three kinds of qualitatively different datasets: free-flow, step, and congested, as shown in Fig. 4. We choose these three scenarios to represent typical driving profiles. The step and congested scenarios are especially critical for energy consumption. In the free-flow profile, the drivers are driving close to the speed limit with little speed variations. In the step profile, the preceding vehicles accelerate from a halt. After reaching a steady-state speed, the vehicles maintain the speed for some time, and then transition to another steady-state speed. In the congested trajectory, the leading vehicle brakes frequently, resulting in consecutive braking by the following vehicles.

Although the simplified dynamics (7) is utilized for controller design, we consider the nonlinear dynamics (2) for closed-loop simulation. To validate the fidelity of the simulations, we also compare those with experimental data generated by implementing RACC and RCCC controllers given by (10) and (17) on a real CAV [31]. The comparison is shown in Fig. 5. The measured velocity data of the ego vehicle are plotted by dashed black curves. In these experiments, we used the controller parameters $\beta = 0.5\,[1/\text{s}]$ in RACC, and $\beta_1 = 0.2\,[1/\text{s}]$, $\beta_L = 0.3\,[1/\text{s}]$ in RCCC. We plot the simulation trajectories on top of the experimental measurements. The simulated trajectories match the experiments very well. In a few cases, the simulation trajectories slightly deviate from experimental ones, for example around $t = 100\,[\text{s}]$ for RACC and $t = 350\,[\text{s}]$ for RCCC, due to the unmodeled transmission and brake dynamics. The energy consumption w (8) is also well captured by simulations. The energy consumption calculated from the simulated trajectory is only 1% more than the measurement in RACC and 6% in RCCC.

Having validated the model used for simulation, we will conduct more simulations of the four control strategies. In RACC
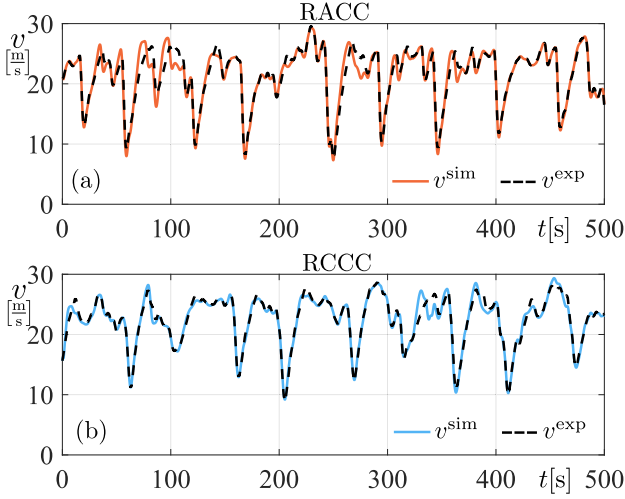
Fig. 5. Comparison between experimental measurements and simulations. Simulation trajectories are plotted as solid curves, and experimental measurements are plotted as black dashed curves. (a) RACC. (b) RCCC with $L = 3, n_{\mathrm{h}} = 1$.

TABLE II
CROSS EVALUATION OF ENERGY CONSUMPTION w. $L = 6$, $n_{\mathrm{h}} = 4$

Parameter \ Simulation [kJ/kg]	Free Flow	Step	Congestion
Free Flow	4.70	4.13	5.40
Step	**4.66**	3.98	4.99
Congestion	4.67	**3.97**	**4.65**

and PACC, the ego vehicle only responds to the vehicle 1. In RCCC and PCCC, connectivity allows the ego vehicle to respond also to the lead vehicle that is chosen to be ranging from $L = 2$ to 6. In simulations, the nonlinear physical term defined in (3) is set to

$$f(v) = 0.0147 + 2.75 \times 10^{-4} v^2. \qquad (30)$$

The acceleration limits (4), (5), parameters of range policy and speed policy are shown in Tables III, IV, and V in Appendix B.

In RCCC, we fix $\alpha = 0.4\,[1/\mathrm{s}]$, and the control parameters β_1, β_L, σ_L are optimized using the method introduced in [33] in each of the three dataset types. The optimal parameters are shown in Table V. The IDM parameters for PCCC are listed in Table VI, and the corresponding MPC parameters are listed in Table VII. The IDM parameter identification process is described in Appendix C. In order to fully demonstrate the potential of predictive methods, we choose the prediction horizon to be 16 seconds [40]. We note that in practice, there is a trade-off between the length of the prediction horizon and the computational efficiency.

B. Sensitivity Analysis

Predictive controllers are known for being more efficient than reactive controllers due to their predicting capabilities. However, their performance is sensitive to prediction accuracy. In what follows, we analyze this sensitivity for PCCC. In Appendix C we listed the human-driver parameters identified from the three datasets: free flow, step, and congested. To evaluate the sensitivity of the predictive controller to the IDM parameters, we
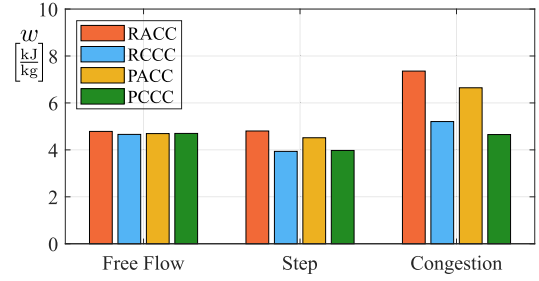

Fig. 6. Comparison of energy consumption of RACC, RCCC, PACC, PCCC. The ego vehicle is connected to vehicle 6 ($L = 6, n_{\mathrm{h}} = 4$).

simulate with all three sets of parameters in the three scenarios, for $L = 2$ to 6 (i.e., $n_{\mathrm{h}} = 0$ to 4), and compare the corresponding energy consumption w defined in (8). We show the results for $L = 2, n_{\mathrm{h}} = 4$ in Table II, and the remaining scenarios are shown in Appendix D. The optimization problem (25) is formulated in MATLAB 2022a with Yalmip [41] and the quadratic programming problems are solved with Gurobi [42]. On our computer with Intel Core i7-9700, the estimation of the hidden vehicles and the calculation of the optimization is 0.1 seconds in total for the predictive controller in each step, while the reactive controller only takes 0.002 seconds in each step.

In Table II, the minimum energy consumption for each dataset is highlighted in bold font. In the free flow simulation, the energy consumption is not sensitive to the choice of parameters in the IDM used for prediction. There is only a 0.8% difference between the highest and the lowest energy consumption. However, the step dataset and the congestion dataset are more sensitive to parameter selection. The difference in energy consumption is 17%. For $L = 2, 3, 4, 5$, we can draw similar conclusions, see Appendix D.

On the other hand, the parameters that obtained the best energy efficiency are not necessarily trained from the corresponding dataset. In the rest of the paper, we choose the IDM parameters trained from the congestion dataset for simulation.

C. Benefits of Connectivity

In this section, we show that connectivity brings great energy benefits. We compare the energy consumption with and without connectivity and provide an explanation for the observed energy savings by comparing simulated trajectories as well as prediction results.

In three traffic scenarios, we compare the energy consumption of the four controllers we introduced above: RACC, RCCC, PACC, and PCCC, as shown in Fig. 6. RACC and PACC serve as the baseline controller compared to RCCC and PCCC to demonstrate the benefits of connectivity. We first investigate the scenario where vehicle 6 is connected, i.e., $L = 6, n_{\mathrm{h}} = 4$. In the free flow scenario, the speed variations of preceding vehicles are small, so there is little difference in the energy efficiency of the four controllers. Contrarily, in the step and congested scenarios, controllers with connectivity (RCCC and PCCC) save a significant amount of energy compared to those without connectivity (RACC and PACC). Compared to RACC, RCCC saves 18.1% energy in the step scenario and 29.2% energy in the congested scenario. Compared to PACC, PCCC saves 12.0% energy in the step scenario and 30.0% energy in the congested scenario. It is also worth noting that RCCC and PCCC achieve
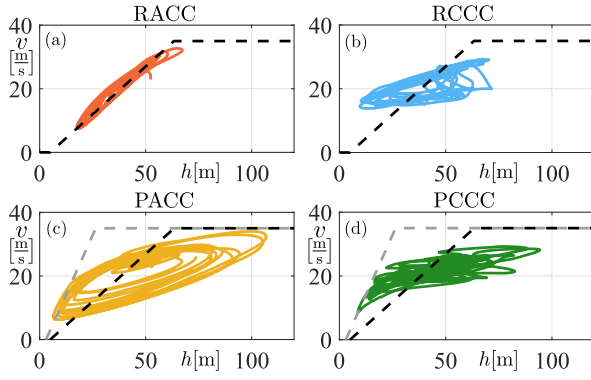
Fig. 7. Phase portraits of RACC, RCCC, PACC, PCCC. The desired range policy is plotted by black dashed lines and the safety constraints for predictive controllers are indicated by grey dashed lines. The ego vehicle is connected to vehicle 6 $(L = 6, n_\mathrm{h} = 4)$.
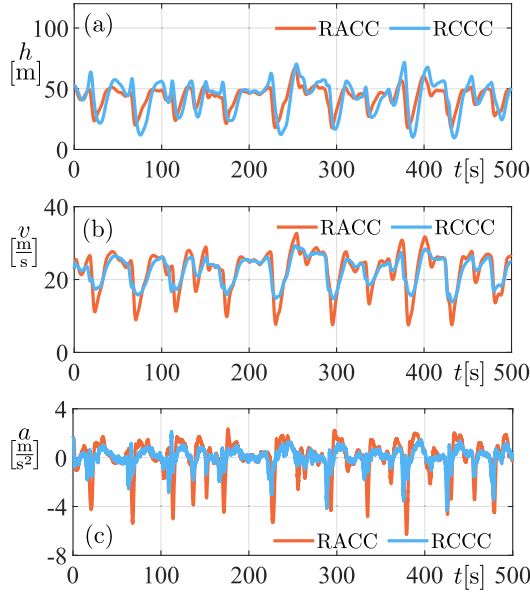

Fig. 8. Comparison of RACC and RCCC in congested scenario for $L = 6$, i.e., $n_\mathrm{h} = 4$. (a) Distance headway; (b) Speed; (c) Acceleration.

similar energy consumption in the free-flow and step scenarios, but PCCC consumes 11.9% less energy compared to RCCC in the congested scenario.

In Fig. 7, we plot the corresponding trajectories in the (h, v)-plane. This demonstrates that reactive controllers, especially RACC, adhere to the pre-defined range policy, while predictive controllers have more degrees of freedom in deviation from the nominal distance headway-speed relationship. In addition, we plot the trajectories of distance headway, velocity, and acceleration in the congested scenario in Figs. 8 and 9. Without connectivity, vehicles suffer from abrupt braking and acceleration, which leads to excessive energy consumption. With information from V2V connectivity, both reactive and predictive controllers can obtain smaller speed variations while maintaining reasonable distance headway.

For predictive controllers, we can directly show the improvement of prediction accuracy due to additional information from V2V connectivity. Fig. 10 shows the error between the prediction
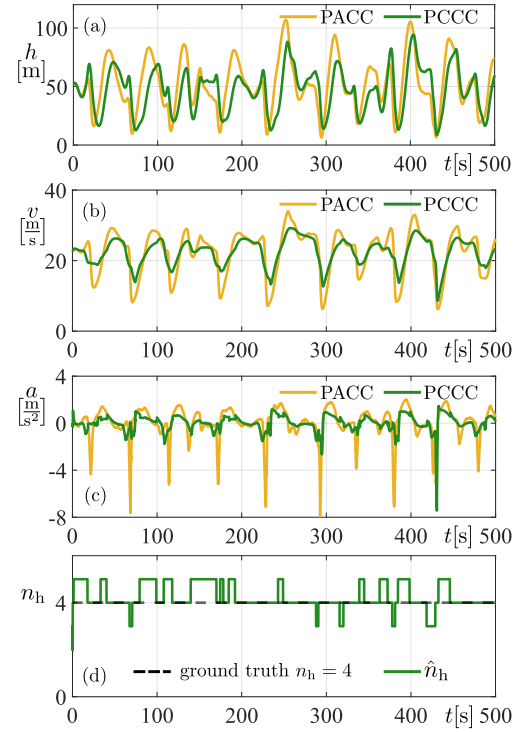
Fig. 9. Comparison of PACC and PCCC in congested scenario for $L = 6$, i.e., $n_\mathrm{h} = 4$. (a) Distance headway; (b) Speed; (c) Acceleration. (d) Estimated number of hidden vehicles.
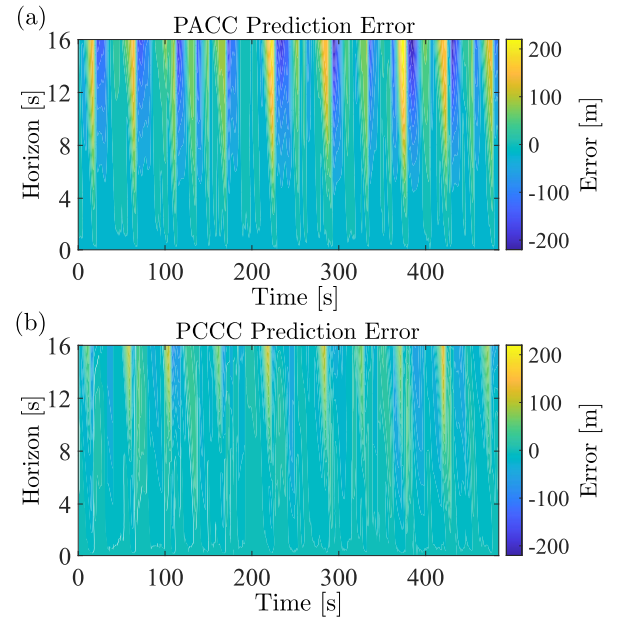
Fig. 10. (a) Prediction error for vehicle 1 of PACC based on the constant-speed assumption. (b) Prediction error of PCCC based on the car-following model.

and the ground truth

$$\mathrm{error}(t, k) = \hat{s}_1(k|t) - s_1(t + k). \tag{31}$$

for PACC and PCCC. In both cases, the accuracy is similar up to about 4 seconds time horizon. However, with additional

Fig. 11. Comparison of connectivity-based trajectory prediction using IDM and the ground truth. Dashed curves show the predicted future trajectories, while solid curves show the trajectory data collected from experiments.

Fig. 12. Comparison of the energy consumption of RACC, PACC, RCCC and PCCC as a function of the number of hidden vehicles for the three traffic datasets in Fig 4.
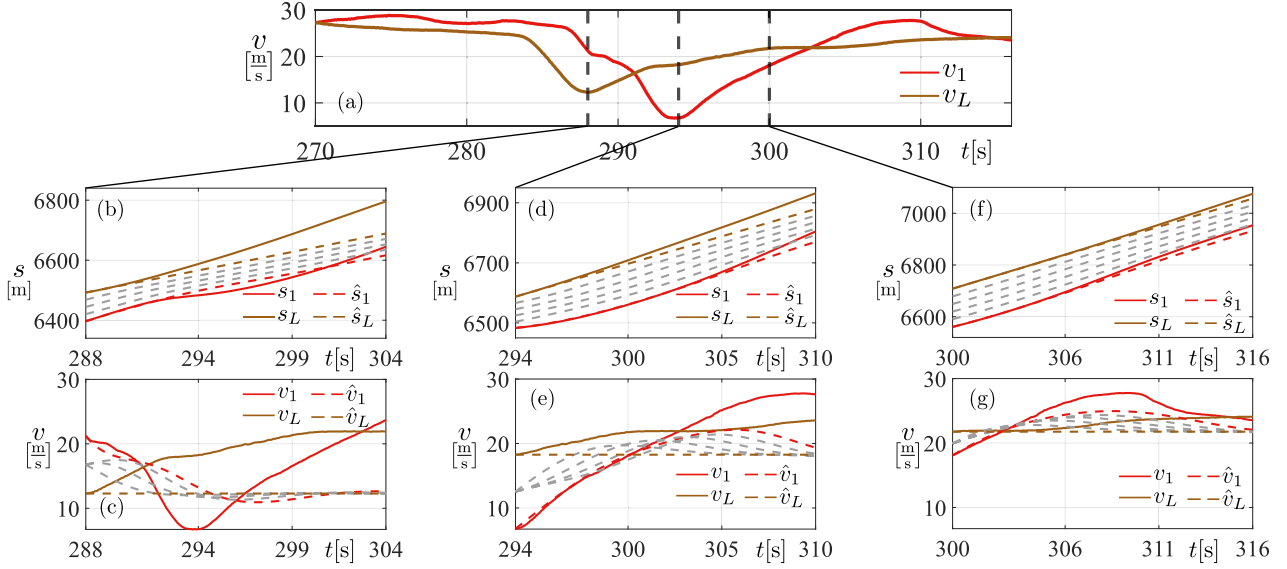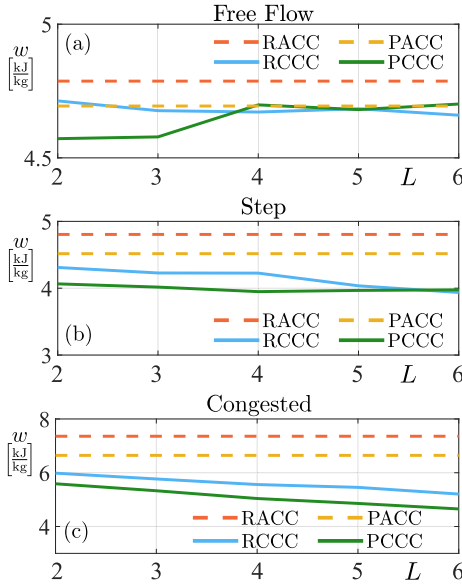
information from V2V connectivity, PCCC can achieve better prediction on a larger horizon.

While a constant-speed assumption is used in both PACC and PCCC predictions, PACC assumes constant speed for vehicle 1 while PCCC assumes constant speed for vehicle L in the distance and predicts the motion of vehicle 1 by a car-following model. The motion of vehicle 1 in the near future is affected by the *past* motion of vehicle L, hence the improvement of prediction accuracy when using data from vehicle L by connectivity.

Fig. 11 further illustrates the motion prediction in a deceleration-acceleration cycle. We show the motion predictions at $t = 288 \,[\mathrm{s}]$ (during deceleration), $t = 294 \,[\mathrm{s}]$ (at the end of deceleration), and $t = 300 \,[\mathrm{s}]$ (during acceleration). At each time instance, we plot the ground truth positions and speeds of vehicles 1 and L as solid lines, and the corresponding predictions as dashed lines. Meanwhile, we show the prediction of hidden vehicles by grey dashed lines. At $t = 288 \,[\mathrm{s}]$ the IDM model is able to predict the deceleration at the initial $4 \,[\mathrm{s}]$ horizon, but fails to predict the harsh deceleration and the acceleration afterward; see panels (b), (c). However, when the leading vehicle finishes braking and starts mildly accelerating, the IDM model is able to capture the car-following behavior and achieve a good prediction of speed for more than 6 seconds, see panels (d), (e) for $t = 294 \,[\mathrm{s}]$ and panels (f), (g) for $t = 300 \,[\mathrm{s}]$.

In practice, the number of hidden vehicles is unknown. Fig. 12 shows the sensitivity of performance (energy efficiency) to the number of hidden vehicles n_{h} in free-flow, step, and congested scenarios.

The number of hidden vehicles n_{h} is upper bounded by the communication range. It is usually large in traffic with lean penetration of connectivity and small in traffic with dense penetration of connectivity. On one hand, due to the string instability of human drivers, connecting to vehicles farther ahead may improve energy consumption. On the other hand, when the ego vehicle connects to a vehicle farther ahead (associated with larger L and n_{h}), the uncertainty of predicting the motion of vehicle 1 by human driver models also increases. For example, in the free-flow scenario, see Fig. 4(a), vehicles 4, 5, and 6 are running with small speed fluctuation, but the speed fluctuation is significantly larger for vehicle 3. If the source of fluctuation is not observable, i.e., vehicles 2 and 3 are hidden, it is hard to make precise predictions using a car-following model. Therefore, in free-flow scenarios, connection to vehicles farther ahead associated with larger n_{h} may increase the energy consumption for PCCC. However, despite the accumulation of uncertainties in the human driver model used for prediction, in all other scenarios, connecting to farther vehicles helps reduce energy consumption. Especially in the congested scenario, connecting to a vehicle in the distance ($L = 6, n_{\mathrm{h}} = 4$) saves 13.0% energy compared

to connecting to a nearby vehicle ($L = 3, n_{\mathrm{h}} = 1$) in case of RCCC, and 16.8% in case of PCCC.

In conclusion, when the speeds of the leading vehicle and the preceding vehicle are highly correlated, such as during congestion, connecting to vehicles farther ahead usually helps to save energy until the increasing uncertainty undermines the correlation and causes more energy consumption.

VI. CONCLUSIONS AND FUTURE WORK

In this paper, we proposed a framework for longitudinal control design for connected automated vehicles driving in mixed traffic consisting of connected and non-connected vehicles. The longitudinal controllers included reactive controllers, where an explicit feedback law was assigned, and predictive controllers, where the control input was optimized in a receding horizon fashion according to the predicted future motion of preceding vehicles.

The controllers realized adaptive cruise control and connected cruise control. In the latter case, beyond-line-of-sight information was obtained using vehicle-to-vehicle (V2V) communication. With lean penetration of connected vehicles in the traffic, various techniques were applied to improve energy efficiency. In reactive controllers, the controller parameters were optimized according to the observed data. In predictive controllers, the number of hidden (non-connected and undetected) vehicles was estimated online, and a car-following model was applied to predict the motion of preceding vehicles.

We conducted extensive simulations based on real human driver data for various driving scenarios. We showed that even lean penetration of connectivity can bring significant energy benefits with both reactive and predictive controllers in all driving scenarios. The influence of the number of hidden vehicles was also studied: connection to vehicles farther in the distance usually brings additional energy benefits until the increasing uncertainty undermines these benefits. The incorporation of uncertainty in car-following models remains an open question due to the stochastic nature and the heterogeneity of human driver behaviors. Currently, there are emerging interests in incorporating learning into the control loop [22], [43]. It is valuable to compare these with the proposed model-based methods in future research.

The proposed control framework can accommodate various engineering specifications, for example, different implementations of reactive control, estimation of the number of hidden vehicles, or motion prediction. The framework is also applicable to various kinds of vehicles including internal-combustion engine vehicles, electric vehicles, or hybrid electric vehicles. While this article only provides simulation results for one connected vehicle in the distance, our framework can be naturally generalized to scenarios with multiple connected vehicles in traffic. It remains for future work to investigate the safety, energy efficiency, and traffic capacity implications with various penetration rates of connected vehicles.

APPENDIX A
SAFETY MARGIN

In (24) we introduced the safety margin d_{margin} to compensate for the uncertainty in the motion of vehicle 1 immediately in front of the CAV, according to the controller proposed in [44]. Considering the randomness of vehicle motion, we use the mean of

the random process as predictions about the motion of preceding vehicles. Let $s_1(k)$, $v_1(k)$, $a_1(k)$ denote the position, velocity and acceleration of vehicle 1 at time moments $k = 0, \ldots, T - 1$. Using the notation $x_1(k) = [s_1(k),\ v_1(k)]^\top$, (20) can be written as

$$x_1(k + 1) = A\, x_1(k) + B\, a_1(k), \tag{32}$$

where

$$A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, \qquad B = \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix}. \tag{33}$$

Then $x_1 := [x_1^\top(1),\ x_1^\top(2), \ldots,\ x_1^\top(T-1)]^\top$ is given by

$$x_1 = \hat{A}\, x_1(0) + \hat{B}\, a_1, \tag{34}$$

where $a_1 = [a_1(0), a_1(1), \ldots,\ a_1(T-2)]^\top$, and

$$\hat{A} = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{T-1} \end{bmatrix}, \qquad \hat{B} = \begin{bmatrix} B & 0 & \cdots & 0 \\ AB & B & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{T-2}B & A^{T-3}B & \cdots & B \end{bmatrix}. \tag{35}$$

We split the deterministic and stochastic parts of the acceleration profile as

$$a_1 = \bar{a}_1 + \tilde{a}_1, \tag{36}$$

where the profile $\bar{a}_1$ is provided by the deterministic car-following model and, for simplicity, we consider the uniform noise profile

$$\tilde{a}_1 = [1,\ 1, \ldots,\ 1]^\top e_{a_1}, \qquad e_{a_1} \sim \mathcal{N}\left(0,\ \sigma_{a_1}^2\right). \tag{37}$$

These result in

$$\hat{B}\, a_1 = \hat{B}\, \bar{a}_1 + \hat{B}\, \tilde{a}_1 = \hat{B}\, \bar{a}_1 + \underbrace{\begin{bmatrix} B \\ AB + B \\ \vdots \\ A^{T-2}B + A^{T-3}B + \cdots B \end{bmatrix}}_{\hat{\hat{B}}} e_{a_1}. \tag{38}$$

Thus, we can derive the distribution of x_1 as

$$x_1 \sim \mathcal{N}\left(\hat{A}\, x_1(0) + \hat{B}\, \bar{a}_1,\ \hat{\hat{B}}\hat{\hat{B}}^\top \sigma_{a_1}^2\right). \tag{39}$$

Similar to the acceleration we may also split the state as

$$x_1 = \hat{x}_1 + \tilde{x}_1, \tag{40}$$

where

$$\hat{x}_1 = \hat{A}\, x_1(0) + \hat{B}\, \bar{a}_1 \tag{41}$$

represents the deterministic part as a result of the deterministic car-following model, and

$$\tilde{x}_1 \sim \mathcal{N}\left(0, \hat{\hat{B}}\hat{\hat{B}}^\top \sigma_{a_1}^2\right), \tag{42}$$
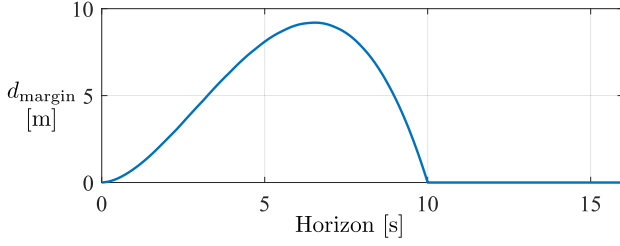
represents the stochastic part.

Fig. 13.   Safety margin over the prediction horizon.

Utilizing the notations $\tilde{x}_1(k) = [\tilde{s}_1(k),\ \tilde{v}_1(k)]^\top$ and $\tilde{x}_1 = [\tilde{x}_1^\top(1),\ \tilde{x}_1^\top(2),\dots,\ \tilde{x}_1^\top(T-1)]^\top$, the calculations above yield

$$\tilde{s}_1(k) \sim \mathcal{N}\left(0, \sigma_{s_1}^2(k)\right), \qquad \sigma_{s_1}^2(k) = b_k\,\sigma_{a_1}^2. \quad (43)$$

where b_k is the element of matrix $\hat{\hat{B}}\hat{\hat{B}}^\top$ at the $(2k-1)$-th row and $(2k-1)$-th column.

In order to compensate for the uncertainty in s_1 we enforce the probabilistic safety constraint

$$\mathbb{P}\left[s_1(k) - s(k) - l - H_{\min}\left(v(k)\right) \geq 0\right] \geq \alpha(k), \quad (44)$$

which can be rewritten as

$$\mathbb{P}\left[\tilde{s}_1(k) \geq s(k) + H_{\min}\left(v(k)\right) + l - \hat{s}_1(k)\right] \geq \alpha(k), \quad (45)$$

where $\alpha(k)$ is the probability of keeping the space headway above the minimum value given by $H_{\min}$. Using the cumulative density function $\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z e^{-\frac{t^2}{2}}\,\mathrm{d}t$ of the standard Gaussian distribution we obtain

$$1 - \Phi\left(\frac{s(k) + H_{\min}\left(v(k)\right) + l - \hat{s}_1(k)}{\sigma_{s_1}(k)}\right) \geq \alpha(k), \quad (46)$$

and exploiting that $1 - \Phi(\alpha) = \Phi(-\alpha)$ this leads to

$$\hat{s}_1(k) - s(k) - l - H_{\min}\left(v(k)\right) - \sigma_{s_1}(k)\,\Phi^{-1}\left(\alpha(k)\right) \geq 0. \quad (47)$$

Therefore, we define the safety margin

$$d_{\mathrm{margin}}(k) = \max\left\{0,\ \sigma_{s_1}(k)\,\Phi^{-1}\left(\alpha(k)\right)\right\}. \quad (48)$$

In this paper, we choose a linearly decreasing $\alpha(k)$ with $\alpha(1) = 0.99$ and $\alpha(K) = 0.5$, where $K\Delta t = 10\,[\mathrm{s}]$, to gradually loosen the probabilistic constraint within the prediction horizon. The corresponding safety margin is plotted in Fig. 13 as a function of time.

APPENDIX B
PARAMETERS OF THE REACTIVE CONTROLLERS

The acceleration limits (4), (5), parameters of range policy and speed policy are shown in Tables III, IV, V respectively.

APPENDIX C
PARAMETERS OF THE PREDICTIVE CONTROLLER

The parameters of the IDM (13) that capture the human driver behavior for the three datasets in Fig. 4 are obtained by solving

TABLE III
ACCELERATION LIMIT OF THE CAV

$u_{\min}$ [m/s^2]	m_1 [1/s]	b_1 [m/s^2]	m_2 [1/s]	b_2 [m/s^2]
−6	0.285	2	−0.121	4.83

TABLE IV
RANGE AND SPEED POLICY PARAMETERS

τ [s]	d [m]	$v_{\max}$ [m/s]
1.67	5	35

TABLE V
REACTIVE CONTROLLER PARAMETERS FOR $L = 6$, I.E., $n_{\mathrm{h}} = 4$

	Controller Type	α [1/s]	β_1 [1/s]	β_L [1/s]	σ_L [s]
Free Flow	RACC	0.4	0.6617		
	RCCC		0.3041	1.0277	5.3372
Step	RACC	0.4	0.4728		
	RCCC		0.2163	1.1459	1.7432
Congested	RACC	0.4	0.4857		
	RCCC		0.2410	0.9895	2.4331

TABLE VI
PARAMETERS OF THE INTELLIGENT DRIVER MODEL

	Free flow	Step	Congested
a_0 [m/s^2]	0.6840	2.2868	2.5732
b_0 [m/s^2]	2.9693	8.5000	8.5000
δ	3.3066	3.0000	4.3393
τ [s]	0.7154	0.9282	0.6409
d [m]	5.0001	5.0000	5.0670
$v_{\max}$ [m/s]	36.0000	32.8682	36.0000

the optimization problem

$$\min_{a_0,b_0,\delta,\tau,h_{\mathrm{st}},v_{\max}} \frac{1}{N_{\mathrm{v}}}\sum_{i=1}^{N_{\mathrm{v}}} \sqrt{\frac{1}{N_{\mathrm{s}}}\sum_{k=1}^{N_{\mathrm{s}}}\left(\hat{h}_i(k) - h_i(k)\right)^2},$$

$$\begin{aligned}
\mathrm{s.t.}\quad & 0.1 \leq a_0 \leq 4, \\
& 0.1 \leq b_0 \leq 8.5, \\
& 3 \leq \delta \leq 5, \\
& 0.1 \leq \tau \leq 4, \\
& 5 \leq d \leq 10, \\
& 30 \leq v_{\max} \leq 36.
\end{aligned} \quad (49)$$

In each dataset, there are 6 human drivers and the cost function averages the error of the IDM for all $N_{\mathrm{v}} = 5$ follower vehicles. For each vehicle, h_i represents the data while $\hat{h}_i$ is obtained from simulations (using the same initial conditions as those in the data) for total time duration N_{s}. We set lower bounds and upper bounds on the IDM parameters considering their physical meanings. We use the NOMAD optimizer [45] to solve the optimization problem (49). The resulting IDM parameters are shown in Table VI.

The remaining parameters of the predictive controllers are given in Table VII.

TABLE VII
PREDICTIVE CONTROLLER PARAMETERS

Δt [s]	q_g	q_a	q_ϵ	τ [s]	d [m]	$\tau_{\min}$ [s]	$d_{\min}$ [m]
0.1	1	960	10^6	1.67	5	0.67	3

TABLE VIII
CROSS EVALUATION OF ENERGY CONSUMPTION w USING PCCC.
$L = 2, n_h = 0$

Simulation [kJ/kg] / Parameter	Free Flow	Step	Congestion
Free Flow	4.57	4.11	5.69
Step	**4.59**	4.07	5.37
Congestion	4.58	**4.04**	**5.59**

TABLE IX
CROSS EVALUATION OF ENERGY CONSUMPTION w USING PCCC.
$L = 3, n_h = 1$

Simulation [kJ/kg] / Parameter	Free Flow	Step	Congestion
Free Flow	4.58	4.13	5.64
Step	**4.58**	4.02	5.45
Congestion	4.60	**3.98**	**5.33**

TABLE X
CROSS EVALUATION OF ENERGY CONSUMPTION w USING PCCC.
$L = 4, n_h = 2$

Simulation [kJ/kg] / Parameter	Free Flow	Step	Congestion
Free Flow	4.70	4.04	5.56
Step	**4.64**	3.95	5.19
Congestion	4.69	**3.96**	**5.04**

TABLE XI
CROSS EVALUATION OF ENERGY CONSUMPTION w USING PCCC.
$L = 5, n_h = 3$

| Simulation [kJ/kg] / Parameter	Free Flow	Step	Congestion
Free Flow	4.68	4.07	5.51
Step	**4.64**	3.97	4.98
Congestion	4.66	**3.95**	**486**

APPENDIX D
CROSS EVALUATION OF PCCC

To evaluate the sensitivity of PCCC to the IDM parameters used for prediction, we conduct simulations with all three sets of parameters in the free-flow, step, and congested scenarios, with $L = 2$ to 6 (i.e., $n_h = 0$ to 4), and compare the corresponding energy consumption w defined in (8). The result for $L = 6, n_h = 4$ is shown in Table II. The rest of the results are shown in Tables VIII, IX, X, XI separately. The minimum energy consumption for each dataset is highlighted in bold font. We observe similar trends to those in Table II. The free-flow dataset is not sensitive to the IDM parameters compared to the other two datasets. The parameters that obtained the best energy efficiency are not necessarily trained from the corresponding dataset.

REFERENCES

[1] A. Sciarretta and A. Vahidi, *Energy-Efficient Driving of Road Vehicles*. Berlin, Germany: Springer, 2020.

[2] M. Zarkadoula, G. Zoidis, and E. Tritopoulou, "Training urban bus drivers to promote smart driving: A note on a greek eco-driving pilot program," *Transp. Res. Part D*, vol. 12, no. 6, pp. 449–451, 2007.

[3] "Taxonomy and definitions for terms related to driving automation systems for on-road motor vehicles," Society of Automotive Engineers, Warrendale, PA, USA, Tech. Rep. J3016_202104, Apr. 2021.

[4] C. R. He, H. Maurer, and G. Orosz, "Fuel consumption optimization of heavy-duty vehicles with grade, wind, and traffic information," *J. Comput. Nonlinear Dyn.*, vol. 11, no. 6, 2016, Art. no. 061011.

[5] A. Sciarretta, G. De Nunzio, and L. L. Ojeda, "Optimal ecodriving control: Energy-efficient driving of road vehicles as an optimal control problem," *IEEE Control Syst. Mag.*, vol. 35, no. 5, pp. 71–90, Oct. 2015.

[6] *Taxonomy and Definitions for Terms Related to Cooperative Driving Automation for on-Road Motor Vehicles* Standard J3216_202107, Society of Automotive Engineers, Warrendale, PA, USA, 2021.

[7] S. E. Shladover, C. Nowakowski, X.-Y. Lu, and R. Ferlis, "Cooperative adaptive cruise control: Definitions and operating concepts," *Transp. Res. Rec.*, vol. 2489, no. 1, pp. 145–152, 2015.

[8] Z. Wang, G. Wu, and M. J. Barth, "A review on cooperative adaptive cruise control (CACC) systems: Architectures, controls, and applications," in *Proc. IEEE 21st Int. Conf. Intell. Transp. Syst.*, 2018, pp. 2884–2891.

[9] 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. IEEE 14th Int. Conf. Intell. Transp. Syst.*, 2011, pp. 260–265.

[10] J. Ploeg, N. van de Wouw, and H. Nijmeijer, "LP string stability of cascaded systems: Application to vehicle platooning," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 2, pp. 786–793, Mar. 2014.

[11] Y. Zheng, S. E. Li, K. Li, and W. Ren, "Platooning of connected vehicles with undirected topologies: Robustness analysis and distributed H-infinity controller synthesis," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 5, pp. 1353–1364, May 2018.

[12] Y. Zheng, S. Eben Li, J. Wang, D. Cao, and K. Li, "Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 1, pp. 14–26, Jan. 2016.

[13] Y. Zheng, S. E. Li, K. Li, F. Borrelli, and J. K. Hedrick, "Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies," *IEEE Trans. Control Syst. Technol.*, vol. 25, no. 3, pp. 899–910, May 2017.

[14] E. van Nunen, J. Reinders, E. Semsar-Kazerooni, and N. van de Wouw, "String stable model predictive cooperative adaptive cruise control for heterogeneous platoons," *IEEE Trans. Intell. Veh.*, vol. 4, no. 2, pp. 186–196, Jun. 2019.

[15] 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, 2018.

[16] S. V. D. Hoef, J. Mårtensson, D. V. Dimarogonas, and K. H. Johansson, "A predictive framework for dynamic heavy-duty vehicle platoon coordination," *ACM Trans. Cyber- Phys. Syst.*, vol. 4, no. 1, pp. 1–25, 2019.

[17] M. H. Basiri, B. Ghojogh, N. L. Azad, S. Fischmeister, F. Karray, and M. Crowley, "Distributed nonlinear model predictive control and metric learning for heterogeneous vehicle platooning with cut-in/cut-out maneuvers," in *Proc. IEEE 59th Conf. Decis. Control*, 2020, pp. 2849–2856.

[18] S. Tsugawa, S. Jeschke, and S. E. Shladover, "A review of truck platooning projects for energy savings," *IEEE Trans. Intell. Veh.*, vol. 1, no. 1, pp. 68–77, Mar. 2016.

[19] L. Bertoni, J. Guanetti, M. Basso, M. Masoero, S. Cetinkunt, and F. Borrelli, "An adaptive cruise control for connected energy-saving electric vehicles," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 2359–2364, 2017.

[20] Y. Bian, C. Du, M. Hu, S. E. Li, H. Liu, and C. Li, "Fuel economy optimization for platooning vehicle swarms via distributed economic model predictive control," *IEEE Trans. Automat. Sci. Eng.*, vol. 19, no. 4, pp. 2711–2723, Oct. 2022.

[21] Q. Lin et al., "Minimize the fuel consumption of connected vehicles between two red-signalized intersections in urban traffic," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 9060–9072, Oct. 2018.

[22] V. Jayawardana and C. Wu, "Learning eco-driving strategies at signalized intersections," in *Proc. IEEE Eur. Control Conf.*, 2022, pp. 383–390.

[23] G. Orosz, "Connected cruise control: Modelling, delay effects, and non-linear behaviour," *Veh. Syst. Dyn.*, vol. 54, no. 8, pp. 1147–1176, 2016.

[24] E. Hyeon, J. Han, D. Shen, D. Karbowski, N. Kim, and A. Rousseau, "Potential energy saving of V2V-connected vehicles in large-scale traffic," *IFAC-PapersOnLine*, vol. 55, no. 24, pp. 78–83, 2022.

[25] R. A. Dollar, T. G. Molnár, A. Vahidi, and G. Orosz, "MPC-based connected cruise control with multiple human predecessors," in *Proc. IEEE Amer. Control Conf.*, 2021, pp. 405–411.

[26] H. Zhou, A. Zhou, T. Li, D. Chen, S. Peeta, and J. Laval, "Congestion-mitigating MPC design for adaptive cruise control based on newell's car following model: History outperforms prediction," *Transp. Res. Part C*, vol. 142, 2022, Art. no. 103801.

[27] T. Ersal et al., "Connected and automated road vehicles: State of the art and future challenges," *Veh. Syst. Dyn.*, vol. 58, no. 5, pp. 672–704, 2020.

[28] G. Orosz, R. E. Wilson, and G. Stépán, "Traffic jams: Dynamics and control," *Philos. Trans. Roy. Soc. A*, vol. 368, no. 1928, pp. 4455–4479, 2010.

[29] M. Treiber, A. Hennecke, and D. Helbing, "Congested traffic states in empirical observations and microscopic simulations," *Phys. Rev. E*, vol. 62, no. 2, 2000, Art. no. 1805.

[30] A. Kesting, M. Treiber, and D. Helbing, "Enhanced intelligent driver model to access the impact of driving strategies on traffic capacity," *Philos. Trans. Roy. Soc. A*, vol. 368, no. 1928, pp. 4585–4605, 2010.

[31] J. I. Ge, S. S. Avedisov, C. R. He, W. B. Qin, M. Sadeghpour, and G. Orosz, "Experimental validation of connected automated vehicle design among human-driven vehicles," *Transp. Res. Part C*, vol. 91, pp. 335–352, 2018.

[32] A. Alan, C. R. He, T. G. Molnar, J. C. Mathew, A. H. Bell, and G. Orosz, "Experimental validation of a safe controller integration scheme for connected automated trucks," 2022, *arXiv:2212.03986*.

[33] M. Shen, C. R. He, T. G. Molnar, A. H. Bell, and G. Orosz, "Energy-efficient connected cruise control with lean penetration of connected vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 24, no. 4, pp. 4320–4332, Apr. 2023.

[34] A. D. Ames, X. Xu, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs for safety critical systems," *IEEE Trans. Autom. Control*, vol. 62, no. 8, pp. 3861–3876, Aug. 2017.

[35] N. Bekiaris-Liberis and M. Krstic, "Compensation of state-dependent input delay for nonlinear systems," *IEEE Trans. Autom. Control*, vol. 58, no. 2, pp. 275–289, Feb. 2013.

[36] Y. Zhang, J. Hu, and M. Wang, "An approach to compensate for heterogeneous feedback and actuation delays in platooning systems," *IFAC-PapersOnLine*, vol. 55, no. 36, pp. 294–299, 2022.

[37] C. R. He, J. I. Ge, and G. Orosz, "Fuel efficient connected cruise control for heavy-duty trucks in real traffic," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 6, pp. 2474–2481, Nov. 2020.

[38] L. Zhang and G. Orosz, "Beyond-line-of-sight identification by using vehicle-to-vehicle communication," *IEEE Trans. Intell. Transp. Syst.*, vol. 19, no. 6, pp. 1962–1972, Jun. 2018.

[39] L. Zhang and E. Tseng, "Motion prediction of human-driven vehicles in mixed traffic with connected autonomous vehicles," in *Proc. IEEE Amer. Control Conf.*, 2020, pp. 398–403.

[40] R. A. Dollar and A. Vahidi, "Efficient and collision-free anticipative cruise control in randomly mixed strings," *IEEE Trans. Intell. Veh.*, vol. 3, no. 4, pp. 439–452, Dec. 2018.

[41] J. Löfberg, "YALMIP: A toolbox for modeling and optimization in MATLAB," in *Proc. IEEE Int. Symp. Comput.-Aided Control Syst. Des.*, 2004, pp. 284–289.

[42] Gurobi Optimization, *Gurobi Optimizer Reference Manual*, LLC, Dayton, OH, USA, 2022.

[43] J. Wang, Y. Lian, Y. Jiang, Q. Xu, K. Li, and C. N. Jones, "Distributed DeeP-LCC for cooperatively smoothing large-scale mixed traffic flow via connected and automated vehicles," 2022, *arXiv:2210.13171*.

[44] T. Ard, R. A. Dollar, A. Vahidi, Y. Zhang, and D. Karbowski, "Microsimulation of energy and flow effects from optimal automated driving in mixed traffic," *Transp. Res. Part C*, vol. 120, 2020, Art. no. 102806.

[45] S. L. Digabel, "Algorithm 909: NOMAD: Nonlinear optimization with the mads algorithm," *ACM Trans. Math. Softw.*, vol. 37, no. 4, pp. 1–15, 2011.
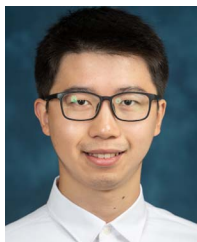
Minghao Shen (Graduate Student Member, IEEE) received the B.Eng. degree in automotive engineering from Jilin University, Changchun, China, in 2019. He is currently working toward the Ph.D. degree in mechanical engineering with the University of Michigan, Ann Arbor, MI, USA. His research interests include stochastic control, machine learning, data-driven control and their applications in the planning and control of connected and automated vehicles.

Robert Austin Dollar (Member, IEEE) received the B.S. degree in mechanical engineering from Purdue University, West Lafayette, IN, USA, in 2014, and the M.S. and Ph.D. degrees in mechanical engineering from Clemson University, Clemson, SC, USA, in 2018 and 2021, respectively. His dissertation topic was anticipative driving of road vehicles. In 2019, he visited IFP Energies Nouvelles, Rueil-Malmaison, France, as a Chateaubriand Fellow. He was a Calibration Engineer with General Motors. Recently, he has returned to GM to focus on motorsport simulation.

Tamas G. Molnar (Member, IEEE) received the B.Sc. degree in mechatronics engineering, the M.Sc. and P.hD. degrees in mechanical engineering from the Budapest University of Technology and Economics, Budapest, Hungary, in 2013, 2015 and 2018, respectively. Between 2018 and 2020, he held postdoctoral position with the University of Michigan, Ann Arbor, MI, USA . Since 2020, he has been a Postdoctoral Fellow with the California Institute of Technology, Pasadena, CA, USA. His research interests include nonlinear dynamics and control, safety-critical control, and time delay systems with applications to connected automated vehicles, robotic systems, and machine tool vibrations.

Chaozhe R. He (Member, IEEE) received the B.Sc. degree in applied mathematics from the Beijing University of Aeronautics and Astronautics, Beijing, China, in 2012, the M.Sc. and Ph.D. degrees in mechanical engineering from the University of Michigan, Ann Arbor, MI, USA, in 2015 and 2018, respectively. He is currently with Plus.AI Inc. and is working on planning and control algorithm development for automated trucking. His research interests include dynamics and control of connected automated vehicles, optimal and nonlinear control theory, and data-driven control.

Ardalan Vahidi (Senior Member, IEEE) received the B.Sc. and M.Sc. degrees in civil engineering from the Sharif University of Technology, Tehran, Iran, in 1996 and 1998, respectively, the M.Sc. degree in transportation safety from George Washington University, Washington, DC, USA, in 2002, and the Ph.D. degree in mechanical engineering from the University of Michigan, Ann Arbor, MI, USA, in 2005. From 2012 to 2013, he was a Visiting Scholar with the University of California at Berkeley, Berkeley, CA, USA. He has held scientific visiting positions at the BMW Technology Office, Mountain View, California and IFP Energies Nouvelles, Rueil-Malmaison, France. He is currently a Professor of mechanical engineering with Clemson University, Clemson, SC, USA. His recent publications span topics connected and autonomous vehicles, efficient transportation, and human performance. He is a Fellow of ASME.

Gábor Orosz (Senior Member, IEEE) received the M.Sc. degree in engineering physics from the Budapest University of Technology, Budapest, Hungary, in 2002 and the Ph.D. degree in applied mathematics from the University of Bristol, Bristol, U.K., in 2006. He held Postdoctoral positions with the University of Exeter, Exeter, U.K., and the University of California, Santa Barbara, CA, USA. In 2010, he joined the University of Michigan, Ann Arbor, MI, USA, where he is currently an Associate Professor of mechanical engineering and of civil and environmental engineering. From 2017 to 2018, he was a Visiting Professor of control and dynamical systems with the California Institute of Technology, Pasadena, CA. In 2022, he was a Visiting Professor of applied mechanics with the Budapest University of Technology, Budapest, Hungary. His research interests include nonlinear dynamics and control, time delay systems, machine learning and data-driven systems with applications to connected and automated vehicles, traffic flow, and biological networks.