2021 American Control Conference (ACC)
New Orleans, USA, May 25-28, 2021

MPC-Based Connected Cruise Control with Multiple Human Predecessors

R. Austin Dollar[1], Tamás G. Molnár[2,3], Ardalan Vahidi[1], Gábor Orosz[2,4]

Abstract— **Model predictive control is applied to regulate the longitudinal motion of a connected automated vehicle in mixed traffic scenarios. A prediction method is proposed to enable model predictive control in low-automation, medium-connectivity situations using instantaneous motion information from multiple predecessor vehicles. This includes detection of unconnected vehicles that may be mixed between connected ones. Simulations using real human driver data for the predecessors show that, if the drivers are well-characterized on average, a hidden unconnected vehicle can be detected over 90 % of the time. Moreover, the resulting preview can recover 46 % of the gap in energy performance between single-predecessor prediction and ideal preview. Results are also compared to a classical controller that utilizes instantaneous information from multiple predecessors.**

I. Introduction

Advanced cruise control systems using various techniques have been studied extensively. Several results in both simulations [1] and experiments [2], [3] have promised energy benefits and collision avoidance guarantees [4]. Adaptive cruise control (ACC) has even made its way into common commercial vehicles. It uses measurements of the preceding vehicle's position and speed to guide the host vehicle's acceleration. While ACC can theoretically mitigate traffic jams, commercial systems have been shown to lack this property [5]. When the vehicles are equipped with vehicle-to-vehicle (V2V) connectivity, connected cruise control (CCC) algorithms can be developed to improve the performance of connected automated vehicles (CAVs). For example, utilizing the preceding vehicle's acceleration may enable the CAV to use shorter headways [6], [7]. Moreover, incorporating motion information from multiple connected vehicles ahead can lead to significant energy savings compared to ACC as demonstrated by recent experiments [8], [9].

In addition to these classical controllers, model predictive control (MPC) has also been implemented for cruise control. MPC can explicitly consider constraints [10], making it well-suited to requirements like collision avoidance and obeying speed limits. Moreover, MPC can utilize preview information about the preceding vehicle's motion [1]. That is, if the preceding vehicle is connected and uses a predictive algorithm, it can share future intentions and the CAV can use this information in its longitudinal controller to adjust its motion in anticipation of future acceleration or braking. This approach can also be extended for a platoon of CAVs [11] that is often referred to as cooperative adaptive cruise control (CACC) [11]. A review of several simulation and experimental studies [6] lists energy improvements of up to 50 % from various forms of anticipative car-following, although the results depend heavily on the scenario and preview quality. Since the preceding vehicle's future trajectory sharing cannot be generally assumed in today's traffic, recent research efforts focus on predicting the motion of the preceding vehicle based on current and/or historical measurements [12], [13].

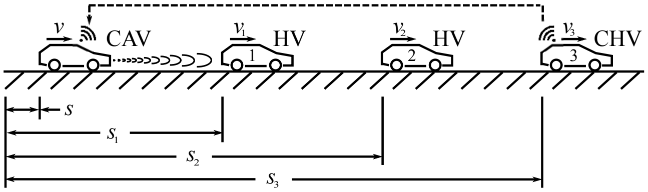

Fig. 1. A connected automated vehicle (CAV) following two conventional human-driven vehicles (HVs) and a connected human-driven vehicle (CHV).

In this paper, we explore the advantages of combining the multi-predecessor connected cruise control algorithms [14] with MPC [6] for scenarios with lean penetration of automation and connectivity. That is, we focus on the longitudinal control design of a CAV when following unconnected human-driven vehicles (HVs) while a connected human-driven vehicle (CHV) further downstream shares its motion information via V2V communication; see Fig. 1. This additional information is incorporated via preview. The intelligent driver model (IDM) [15] is used to predict the motion of the CAV's immediate predecessor given the motion of a CHV further ahead. An identification algorithm is also proposed to determine the number of "hidden" unconnected human-driven vehicles that are occluded from the CAV's sensors. This approach contrasts with that of [12], which derives a road speed profile from recent CHV motion without explicitly identifying such hidden vehicles. Experimental data from real vehicles is utilized to evaluate the energy consumption of different control strategies.

The paper is outlined as follows. In Section II the collection of human predecessor data is described which is used in the rest of the paper to test the proposed algorithms. Section III describes the vehicle dynamics, energy consumption, and driver behavior. Section IV gives the details of the classical and model predictive controllers while the results are presented in Section V. Finally, Section VI summarizes the paper and outlines the steps for practical implementation.

[1] Austin Dollar and Ardalan Vahidi are with Department of Mechanical Engineering, Clemson University, SC 29634, USA. rdollar@clemson.edu, avahidi@clemson.edu

[2] Tamás Molnár and Gábor Orosz are with Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109, USA. molnart@umich.edu, orosz@umich.edu

[3] Tamás Molnár is also with the Department of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125, USA.

[4] Gábor Orosz is also with the Department of Civil and Environmental Engineering, University of Michigan, Ann Arbor, MI 48109, USA.

978-1-6654-4197-1/$31.00 ©2021 AACC

II. Data Collection and Simulation Scenario

In this paper, we utilize the measurement data collected by the experiments reported in [8], where eight vehicles performed car-following on a single-lane straight road. Here we focus on four members of the eight-vehicle string, where three human-driven vehicles were followed by a connected automated vehicle (CAV) as illustrated in Fig. 1.

During the experiment, the GPS position and speed shown in Fig. 2 were recorded for each vehicle, which allows one to analyze the human driving behavior, design control algorithms for the CAV, and evaluate the corresponding energy consumption. The GPS position and speed data were shared amongst the vehicles via vehicle-to-vehicle (V2V) connectivity every 0.1 s. This way, the CAV was able to respond to the motion of multiple preceding vehicles using the control algorithm summarized in Section IV-A.

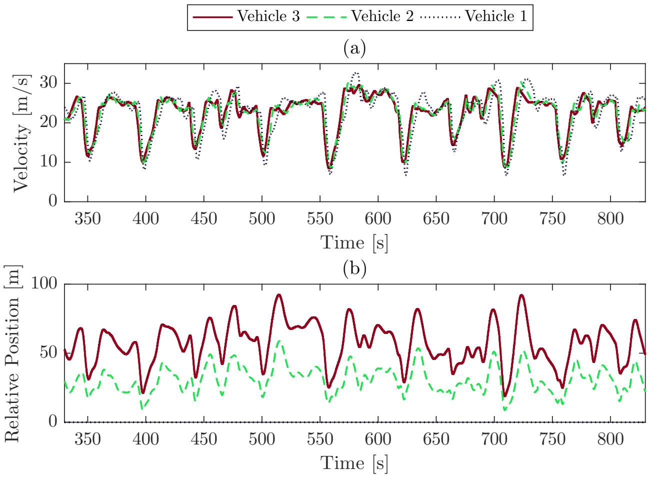

Fig. 2. Experimental data for the CAV's predecessors. (a) speed data for three consecutive vehicles. (b) distance between vehicle 1 and vehicle 2 (green) and distance between vehicle 1 and vehicle 3 (red).

Although all human-driven vehicles were connected in the experiment, our goal here is to relax this assumption to mixed connectivity. Therefore, in the proposed control algorithms we assume that vehicle 2 in Fig. 1 is not equipped with connectivity and is occluded from the CAV's sensors. Hence the CAV can access the position s_1 and velocity v_1 of the preceding human-driven vehicle 1 and the position s_3 and velocity v_3 of the distant connected human-driven vehicle 3 only. That is, the human-driven vehicle 2 remains hidden for the CAV and its existence can only be inferred from the motions of vehicles 1 and 3.

The classical controller implemented in the experiments of [8] will be used as a benchmark when evaluating the performance of the proposed MPC algorithm. In the evaluations of MPC, two vehicles are simulated numerically: the CAV and vehicle 1. Unlike the CAV that uses a closed-loop controller, the experimental speed trajectory is imposed on vehicle 1. Figure 2 shows the measured velocity and position trajectories, with position expressed relative to vehicle 1.

III. Modeling

Models required for this study include longitudinal vehicle dynamics, wheel-to-distance energy consumption, and driver behavior. This section addresses these topics in order.

A. Vehicle Dynamics with Delay

The dynamics of the CAV is modeled using the delayed double integrator

$$\begin{aligned}
\dot{s}(t) &= v(t),\\
\dot{v}(t) &= a(t),\\
a(t) &= \min\left\{\max\left\{u(t-\sigma),\ -\underline{u}\right\},\ \overline{u}\big(v(t)\big)\right\}.
\end{aligned} \tag{1}$$

Here the dot represents derivative with respect to the time t, while s, v and a denote the position, velocity and acceleration of the vehicle, respectively. The acceleration command u from the cruise controller becomes the actual acceleration after delay σ, which is considered to be 0.6 s for the CAV and incorporates the powertrain and communication delays. This delay matches that used for simulation in [16], which obtained a mildly longer delay of 0.8 s to 0.9 s in experimental data. The maximal acceleration command $\overline{u}$ represents the torque limitation of the engine that depends on the velocity v and $\underline{u}$ represents the braking torque limitation; see Fig. 6(a). The resistance terms arising from rolling resistance, air resistance and grade are neglected for simplicity. These can be compensated using lower-level controllers in practical implementations, as demonstrated in [3]. We remark that the model (1) is also used for the controller's prediction of the human-driven vehicles 1 and 2 where u is obtained from car-following models. In this case the delay is neglected for simplicity since including it changed the model fitting metric in (5) by less than 0.5 %. The predecessor vehicles' experimental trajectories are used as the simulation testbed, capturing more complex longitudinal phenomena like gear shifts and introducing realistic mismatch to the predictions.

In order to design and evaluate the control algorithms we discretize time as $t = k\Delta t$, $k \in \mathbb{Z}$ where the time step Δt is chosen such that $q = \sigma/\Delta t$ is an integer number. Applying the Euler discretization to (1) results in the discrete time system

$$\begin{aligned}
s(k+1) &= s(k) + v(k)\Delta t + a(k)\Delta t^2/2,\\
v(k+1) &= v(k) + a(k)\Delta t,\\
a(k) &= \min\left\{\max\left\{\delta_q(k),\ -\underline{u}\right\},\ \overline{u}\big(v(k)\big)\right\},\\
\delta_q(k+1) &= \delta_{q-1}(k),\\
&\ \vdots\\
\delta_1(k+1) &= \delta_0(k),\\
\delta_0(k) &= u(k),
\end{aligned} \tag{2}$$

where we introduced the notation $x(k) := x(k\Delta t)$. In this paper we choose $\Delta t = 0.2$ s which results in $q = 3$ making the state space of (2) five dimensional. This choice provides us with the required accuracy while being computationally feasible for real-time implementation on real automobiles. For example, when choosing $\Delta t = 0.1$ s the resulting trajectories are indistinguishable from those obtained for $\Delta t = 0.2\,\mathrm{s}$, while the computational time increases significantly due to the eight dimensional state space and greater number of control moves along horizons of equal time.

B. Energy Consumption

Each algorithm's energy performance is assessed based on the mass-normalized energy E expended at the wheels over the total simulated time t_f:

$$E = \int_0^{t_\mathrm{f}} \max\left\{ \big(a\left(t\right) + a_\mathrm{r} + c_\mathrm{r} v^2\left(t\right)\big),\ 0\right\} v\left(t\right) \mathrm{d}t. \tag{3}$$

This energy model accounts for rolling resistance via a_r, aerodynamic drag via c_r, and braking losses via the integrand's non-negativity. However, it does not attempt to capture changes in powertrain efficiency that may be highly vehicle-specific. Here we use the vehicle parameters $a_\mathrm{r} = 1.47 \times 10^{-1}\,\mathrm{m/s^2}$ and $c_\mathrm{r} = 2.75 \times 10^{-4}\,\mathrm{m^{-1}}$ which correspond to the values of a Ford Escape; see [1].

C. Human Driver Behavior

The Intelligent Driver Model (IDM) [15] is used to predict the HVs' future motion. This model determines the n-th vehicle's acceleration using the following closed form law:

$$u_n = a_0 \left[1 - \left(\frac{v_n}{v_\mathrm{max}}\right)^{\delta} - \left(\frac{d_\mathrm{des}}{d_n}\right)^2 \right], \tag{4a}$$

where the desired distance is given by

$$d_\mathrm{des} = d_\mathrm{st} + \max\left\{0,\ \tau_\mathrm{h} v_n - \frac{v_n \dot{d}_n}{\sqrt{4 a_0 b_0}}\right\}. \tag{4b}$$

The distance between vehicle n and its predecessor $n+1$ is calculated from the positions as $d_n = s_{n+1} - s_n - \ell$ where ℓ denotes vehicle length. This yields the relative velocity $\dot{d}_n = v_{n+1} - v_n$. When simulating the IDM online as preview for MPC, we also use the Euler discretization of (2) with the discretization step $\Delta t = 0.2\,\mathrm{s}$ while omitting the delay ($q = 0$).

The IDM contains six parameters which are identified by using the data of five human drivers that participated in the experiment of [8]. The parameters were found using the global optimizer NOMAD [17] while solving the optimization problem

$$\begin{aligned}
\min_{x_\mathrm{p}} \quad & J_\mathrm{p} = \frac{1}{N_v} \sum_{n=1}^{N_v} \sqrt{\frac{1}{N_s} \sum_{k=1}^{N_s} \Big(\hat{d}_n(k) - d_n(k)\Big)^2}, \\
\text{s.t.} \quad & 0.1\,\mathrm{m/s^2} \le a_0 \le 4\,\mathrm{m/s^2}, \\
& 0.1\,\mathrm{m/s^2} \le b_0 \le 8.5\,\mathrm{m/s^2}, \\
& 0.1\,\mathrm{m} \le d_\mathrm{st} \le 10\,\mathrm{m}, \\
& 0.1\,\mathrm{s} \le \tau_\mathrm{h} \le 4\,\mathrm{s}, \\
& 0 \le v_\mathrm{max} \le 36\,\mathrm{m/s}, \\
& 1 \le \delta \le 10, \\
& x_\mathrm{p} = \begin{bmatrix} a_0 & b_0 & d_\mathrm{st} & \tau_\mathrm{h} & v_\mathrm{max} & \delta \end{bmatrix},
\end{aligned} \tag{5}$$

where the cost function J_p compares the distance $\hat{d_n}$ obtained by simulating the IDM (4) to d_n given by the measurement data. The simulation was run for $N_s = 5001$ time steps and the error was averaged for $N_v = 5$ human-driven vehicles. A discretization step of $0.1\,\mathrm{s}$, that is the nominal step in the experimental data, was used during the parameter identification process. While this differs from the $0.2\,\mathrm{s}$ time step used for MPC preview generation, the impact of this time step change on the root mean squared error of each human driver's preceding gap was less than $0.5\,\%$.

Table I lists the identified IDM parameters. Notice that the identified value of b_0 is more severe than a typical human deceleration. However, [18] also found that a cluster of human drivers is best modeled with a similarly large $b_0 \approx 7\,\mathrm{m/s^2}$. This suggests that for some drivers, an uncomfortably large acceleration value optimizes b_0. We remark that one may also find more individualized parameters but the corresponding models may not generalize well for other experiments.

TABLE I
IDM PARAMETERS

Parameter	Description	Result
a_0	Maximum acceleration	2.43 m/s^2
b_0	Deceleration coefficient	8.5 m/s^2
d_st	Stopping distance	3.3 m
τ_h	Time headway	0.76 s
v_max	Speed limit	36 m/s
δ	Exponent	6.13

Although the IDM is shown to improve MPC performance in this study, the proposed framework does not fundamentally demand a particular driver model. For example, a neural network model like [19] with similar inputs and outputs to the IDM could be substituted.

IV. Control Design

This section will describe the multi-predecessor classical controller used in [8], [14] and the proposed MPC algorithm. Both controllers are designed around three requirements:

1) Target a headway that is a linear function of speed with stopping distance $d_\mathrm{st} = 5\,\mathrm{m}$ and time headway $T = 1.67\,\mathrm{s}$.
2) Maintain a headway that is no less than a linear function of speed with stopping distance $\underline{d} = 3\,\mathrm{m}$ and additive time headway $\underline{T} = 0.67\,\mathrm{s}$.
3) Maintain a headway that is no greater than a linear function of speed with stopping distance $\overline{d} = 10\,\mathrm{m}$ and additive time headway $\overline{T} = 2.86\,\mathrm{s}$.

A. Classical Controller

As a benchmark for evaluating the performance of MPC, we describe a classical controller [14] that was implemented on a CAV [8] during the experiments discussed in Section II. Given that the CAV is able to respond to y preceding vehicles via connectivity, the classical controller prescribes the CAV's acceleration by the following control input:

$$u = \alpha\big(V(d) - v\big) + \sum_{n=1}^{y} \beta_n \big(W(v_n) - v\big). \tag{6}$$

The first term in (6) ensures that the CAV keeps a desired distance by adjusting its speed v to a distance-dependent desired speed V given by the range policy

$$V(d) = \min\{\max\{0, (d - d_{\mathrm{st}})/T\}, v_{\max}\}. \tag{7}$$

This range policy suggests that if the distance d drops below the value d_{st}, the CAV shall stop, while above d_{st} the CAV shall increase its speed linearly with gradient $1/T$ until it reaches the speed limit $v_{\max}$. The second term in (6) ensures that the CAV adjusts its speed v to the speed v_n of the n-th preceding vehicle or to the speed limit, according to

$$W(v_n) = \min\{v_n, v_{\max}\}. \tag{8}$$

The controller involves $y+1$ control gains: α and β_n, $n = 1, \ldots, y$.

The parameters of the controller were tuned experimentally in [8]. Accordingly, we selected the following parameters and simulated the CAV's motion (1) with control law (6) as benchmark: $d_{\mathrm{st}} = 5\,\mathrm{m}$, $T = 1.67\,\mathrm{s}$ (see requirement 1 above), $v_{\max} = 35.76\,\mathrm{m/s}$, $\alpha = 0.4\,\mathrm{s}^{-1}$, $\beta_1 = 0.2\,\mathrm{s}^{-1}$, $\beta_2 = 0.0\,\mathrm{s}^{-1}$, and $\beta_3 = 0.6\,\mathrm{s}^{-1}$.

B. Optimal Controller

A decentralized MPC is adapted from [20] and applied here. It uses the model (2) to minimize the CAV's acceleration command given the preceding vehicle's predicted trajectory. This process involves three main tasks. First, any unconnected predecessors between the sensed and connected predecessors must be identified. Then, the immediate predecessor's trajectory needs to be predicted based on the available information. Finally, a receding horizon optimal control problem has to be solved for the acceleration command. Figure 3 lays out the architecture of the overall process.

1) Predecessor Identification: It is possible that one or more hidden unconnected vehicles exist between the immediate sensed predecessor and the next connected vehicle ahead, numbered l. Figure 1 depicts the case $l = 3$; however, the value of l is unknown to the CAV. Hidden vehicles are detected by simulating the string from vehicle l to vehicle 1 and choosing the number of hidden vehicles n_{h} that best explains vehicle 1's past motion. A relatively long past simulation horizon of $M\Delta t = 23\,\mathrm{s}$ is used with a shorter $K\Delta t = 3\,\mathrm{s}$ detection horizon to compute the cost for hidden vehicle identification. This allows the imprecise initial conditions to decay before the string model is evaluated. Figure 4 shows the relationship between these historical horizons, which are not to be confused with MPC's future prediction horizon.

Considering the mean squared error of vehicle 1's estimated position $\hat{s}_1$ relative to its actual position s_1 we solve the optimization problem

$$\begin{aligned} n_{\mathrm{h}}^*(k) =& \arg \min_{n_{\mathrm{h}}(k)} \frac{k_{\mathrm{s}}}{K} \sum_{j=M-K+1}^{M} \left(\hat{s}_1\left(n_{\mathrm{h}}(k), j\right) - s_1(j)\right)^2, \\ & \text{s.t. } 0 \le n_{\mathrm{h}} \le \min_j \frac{s_l(j) - s_1(j) - \ell - d_{\mathrm{st}}}{\ell + d_{\mathrm{st}}}, \end{aligned} \tag{9}$$

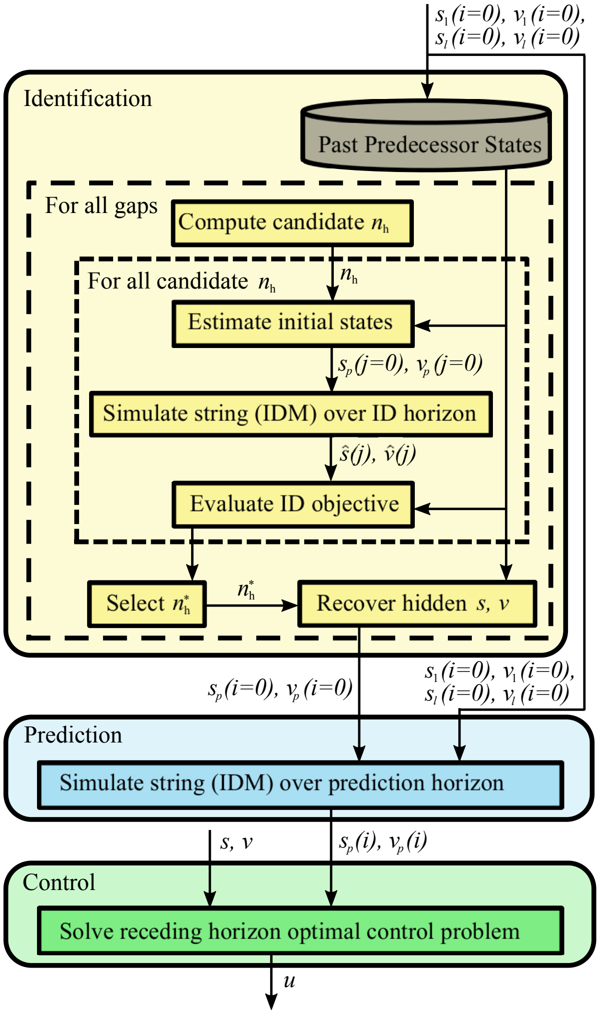
$s_1(i{=}0)$, $v_1(i{=}0)$,
$s_l(i{=}0)$, $v_l(i{=}0)$
Identification
Past Predecessor States
For all gaps
Compute candidate n_{h}
For all candidate n_{h}
n_{h}
Estimate initial states
$s_p(j{=}0)$, $v_p(j{=}0)$
Simulate string (IDM) over ID horizon
$\hat{s}(j)$, $\hat{v}(j)$
Evaluate ID objective
Select n_{h}^*
n_{h}^*
Recover hidden s, v
$s_p(i{=}0)$, $v_p(i{=}0)$
$s_1(i{=}0)$, $v_1(i{=}0)$,
$s_l(i{=}0)$, $v_l(i{=}0)$
Prediction
Simulate string (IDM) over prediction horizon
s, v
$s_p(i)$, $v_p(i)$
Control
Solve receding horizon optimal control problem
u

Fig. 3. The optimal controller's architecture involving the identification of hidden vehicles, the prediction of the preceding vehicle's future motion and the optimal control of the connected automated vehicle.

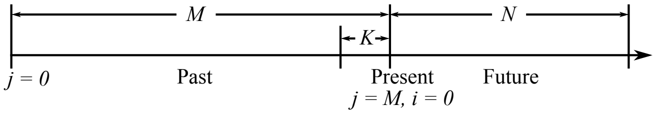

Fig. 4. Timeline of the simulation, detection, and prediction horizons of lengths M, K, and N steps, respectively.

where

$$k_{\mathrm{s}} = \begin{cases} 1 & \text{if} \quad n_{\mathrm{h}}(k) = n_{\mathrm{h}}^*(k-1), \\ 1.5 & \text{otherwise}, \end{cases} \tag{10}$$

and s_l denotes the position of the leading CHV. The minimum possible number of hidden vehicles n_{h} is 0 and the maximum is determined by assuming that vehicles of length $\ell = 5\,\mathrm{m}$ are separated by the stopping distance d_{st}. The switching penalty k_{s} prevents the number of detected vehicles n_{h}^* at the current step k from switching unless a large difference in objective supports the switch. This is necessary since changes in the number of detected vehicles can cause the acceleration command to change abruptly, as will be demonstrated in Section V-A.

The ID simulations, used to identify hidden vehicles, are executed using the IDM (4) and the double integrator (1) without delay for the longitudinal dynamics. The initial position $s_p(0)$ and speed $v_p(0)$ for hidden vehicle p are assumed to be

$$s_p\left(0\right) = s_1\left(0\right) + p\frac{s_l\left(0\right) - s_1\left(0\right)}{n_{\mathrm{h}} + 1}, \tag{11}$$

and

$$v_p\left(0\right) = \frac{v_l\left(0\right) + v_1\left(0\right)}{2}, \tag{12}$$

respectively, where $v_l\left(0\right)$ denotes vehicle l's speed and the states of vehicles 1 and l are available from past data. After the number of hidden vehicles n_{h}^* is determined, the final state of the ID simulation for $n_{\mathrm{h}} = n_{\mathrm{h}}^*$ is taken as the current hidden vehicle state.

2) Prediction: Three different types of prediction, listed below, are considered in this paper in order to obtain a preview of vehicle 1's future motion over the horizon of length $N\Delta t = 16\,\mathrm{s}$; see Fig. 4.

- *Full preview*: In this case we consider that vehicle 1's actual future motion is available to the CAV. This is not possible in practice but serves as an upper bound on performance. Although MPC that shares such future trajectories has been proposed [1], [11], such preview is still subject to change in closed-loop operation because of disturbances.
- *Driver model*: In a partially-connected environment where most connected vehicles are not automated, only the instantaneous position and speed of the CHVs and vehicle 1 are known. Sequential simulations from the front to rear of the string are therefore carried out assuming constant speed for the lead CHV. This process yields vehicle 1's predicted trajectory as demonstrated in Fig. 5. Subsequent vehicles' accelerations are calculated from the IDM (4). For the simulation shown in Fig. 5, the two vehicles following the lead CHV are predicted to temporarily increase their speeds.
- *Probability Model*: Restricting the CAV to use data from vehicle 1 only precludes the use of a driver model to account for vehicle 1's gap ahead. In [1] a probability model based on the preceding vehicle's speed and brake light state was used to operate under such conditions. After substituting chance constraints for the worst-case collision avoidance constraint used in [1], this probabilistic prediction method is also evaluated as a baseline to show the benefits of connectivity.

3) Optimal Control Problem: With vehicle 1's future trajectory now predicted, it is possible to solve the receding horizon optimal control problem. This controller is based on the MPC used in [20], which balances time headway tracking with acceleration minimization at each prediction step i. The time headway $T = 1.67\,\mathrm{s}$ and stopping distance $d_{\mathrm{st}} = 5\,\mathrm{m}$ are used to meet requirement 1 set at the beginning of this section. Although the design goal is saving energy, speed smoothing is chosen over fuel or energy minimization in order to arrive at a convex quadratic program, which can be solved in polynomial time [21]. In [22] it was shown that most fuel consumed by heavy-duty trucks in traffic was due to the speed fluctuations. Moreover, [3] demonstrated that a speed-smoothing MPC using preview information can reduce the energy consumption of real passenger vehicles relative to a non-predictive baseline controller.

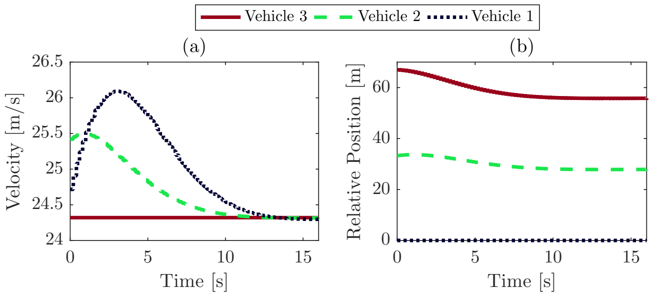

Fig. 5. IDM-based future velocity (a) and future position (b) for the three predecessors, beginning with a constant-speed assumption for the distant CHV and resulting in a more complex trajectory for the immediate predecessor. Positions are shown relative to vehicle 1.

Here we solve the optimal control problem

$$\begin{aligned}
\min_{u(i)} J = {} & q_{\mathrm{g}}\big(s_1(N) - s(N) - \ell - d_{\mathrm{st}} - Tv(N)\big)^2 \\
& + \sum_{i=0}^{N-1} \Big[q_{\mathrm{g}}\big(s_1(i) - s(i) - \ell - d_{\mathrm{st}} - Tv(i)\big)^2 \\
& \qquad + q_{\mathrm{a}}\big(u^2(i) + a^2(i)\big)\Big], \\
\text{s.t.} \quad & 0 \le v \le v_{\max}, \\
& 0 \le s_1 - s - \ell - \underline{T}v - \underline{d} - d_{\mathrm{c}}, \\
& -\underline{u} \le u \le \overline{u}\left(v\right), \\
& -\underline{u} \le a \le \overline{u}\left(v\right), \\
& \overline{u}\left(v\right) = \min\left\{m_1 v + b_1, m_2 v + b_2\right\},
\end{aligned} \tag{13}$$

where the model (2) without saturation establishes the relationship between the states s, v, and a and the control input u. The constants $m_1 = 0.285\,\mathrm{s}^{-1}$, $m_2 = -0.121\,\mathrm{s}^{-1}$, $b_1 = 2\,\mathrm{m/s}^2$, and $b_2 = 4.83\,\mathrm{m/s}^2$ define a piecewise linear approximation $\overline{u}\left(v\right)$ of the vehicle's maximal acceleration capacity and $\underline{u} = 8.5\,\mathrm{m/s}^2$ is the vehicle's braking capacity, as shown in Fig. 6(a). The resulting constraints were used in [1] to approximate the maximum acceleration obtained from a powertrain model.

Notice that requirement 2 is coded as a constraint in the optimization problem (13). Additionally, with the gap weight fixed at $q_{\mathrm{g}} = 1$, the acceleration weight q_{a} was tuned to meet requirement 3. This results in $q_{\mathrm{a}} = 960$ for probabilistic preview, $q_a = 1200$ for IDM-based preview, and $q_a = 200$ for full preview.

A chance constraint after [23] is applied to prevent violations of the collision constraint due to mismatch between predicted and actual acceleration of the preceding vehicle. The safety distance d_{c}, shown in Fig. 6(b) as the function of time, prevents constraint violations with a specified probability. This probability is chosen to be 0.99999 at the first prediction stage and 0.5 after 10 s in the future.

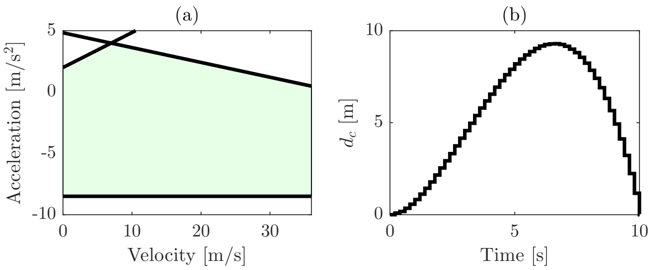

Fig. 6. The acceleration constraints (a) and the chance constraint (b) used for collision avoidance.

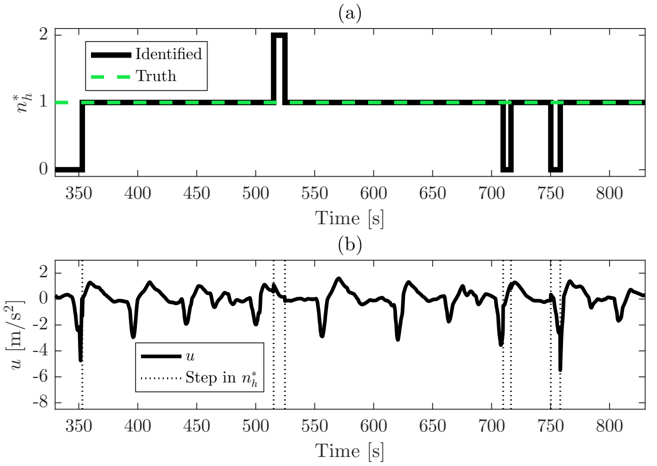

Fig. 7. Detection accuracy (a) and the impact of detection switching on the acceleration command (b).

V. Results

Results are presented in three main performance areas: detection accuracy, energy consumption, and headway variations.

A. Detection Accuracy

The detection component correctly identified the presence of a single unconnected vehicle (vehicle 2 in Fig. 1) 90.6 % of the time as shown in Fig. 7. Transitions between different numbers of identified vehicles can cause jerks in the acceleration command from MPC, highlighting the importance of avoiding gratuitous transitions.

B. Energy Consumption

Table II lists the normalized energy consumption for each car-following system along with the mean gap. As expected, MPC using full preview of vehicle 1's trajectory (MPC Full Prev.) delivered the lowest energy consumption. Such accurate preview is not available in practice because it is non-causal; even preview from intention-sharing closed-loop systems is subject to change in the presence of disturbances. The other results in Table II demonstrate the utility of multiple-predecessor connectivity. Such systems consumed 5.0 % to 12.7 % less energy compared to the probability-model system (MPC 1 Pred.) that received information from only one predecessor. Identification of the hidden vehicle between vehicles 1 and 3 is also important to energy efficiency. When identification was not used (MPC ID Off), the CAV consumed 10.3 % more energy than it did when vehicle 2 was connected (MPC Ideal ID). Introducing identification (MPC ID On) eliminated 80.1 % of the difference in energy consumption between these two cases.

TABLE II

Energy and Headway Results

Controller	Energy [J/kg]	Mean Gap [m]	Gap RMSE [m]
MPC 1 Pred.	6180	50.34	14.29
Classical	5868	42.45	6.95
MPC ID Off	5948	34.81	13.77
MPC ID On	5505	47.32	12.98
MPC Ideal ID	5395	48.29	12.29
MPC Full Prev.	4698	43.84	8.29

The classical controller (Classical) compares most directly to the MPC with identification (MPC ID On) since they both assume that vehicle 1 can be sensed and vehicle 3 is connected. In this comparison, receding horizon optimization and IDM-based string preview helped reduce energy consumption by 6.2 %.

C. Headway Variations

In typical car-following applications, shorter headways are desirable because they enable greater traffic flow at equal speed. Headway must, however, be long enough for safety. Table II shows the average headway for the various controllers, while headway signals are plotted in Fig. 8. With no attempt made to identify the hidden predecessor, the MPC consistently commanded excessively high acceleration due to its prediction based on the wide apparent gap between vehicle 1 and vehicle 3. This reduced the mean gap in Table II and sometimes caused the CAV to violate the headway specification as shown in Fig. 8(c). Identifying the hidden vehicle based on historical data enabled MPC to meet the headway specification without re-tuning.

The root mean squared error (RMSE) of the gap relative to the reference (defined by requirement 1) measures the quality of gap control. Even with full preview, MPC tended to trade off gap tracking to reduce acceleration and deviated more from its gap reference compared to classical control as shown in Table II. An advantage of MPC is its ability to explicitly constrain minimum gap, which makes deviation from the reference safer. Among the three MPC systems that used IDM-based prediction, identifying the hidden vehicle improved gap tracking RMSE by 5.7 % and complete knowledge of vehicle 2's existence improved it by 10.7 %.

VI. Conclusion

A method for leveraging multi-predecessor connectivity in MPC-based connected cruise control was proposed assuming lean penetration of connectivity and automation. Possible unconnected vehicles between connected ones were systematically identified by analyzing past connected vehicle motions. The results were compared against single-predecessor MPC-based cruise control, multi-predecessor classical control, and MPC with ideal preview using real human driver data for

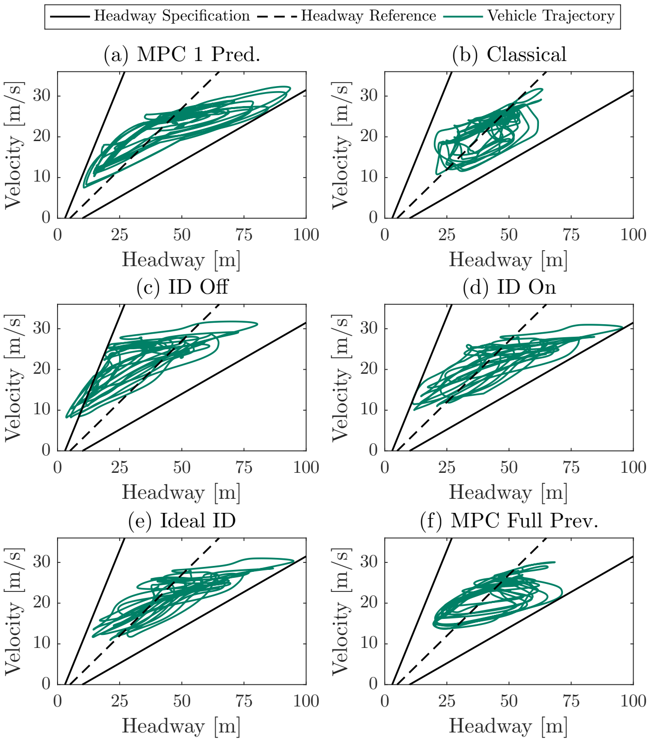

Fig. 8. Headway tracking comparison with (a) MPC considering vehicle 1 with a probability model, (b) classical control considering vehicles 1 and 3, (c) MPC without identifying vehicle 2, (d) MPC identifying vehicle 2 automatically, (e) MPC given the existence and current state of vehicle 2, and (f) MPC with full preview of vehicle 1's future motion.

the predecessors. In general, multi-predecessor connectivity improved control performance even though only the immediate preceding vehicle appeared directly in the MPC's optimal control problem, because connectivity enables more accurate preview of the preceding vehicle's motion. Classical control using multiple predecessors performed better than MPC with information from only one predecessor, although MPC using multiple predecessors performed better still. MPC's performance depended largely on correct identification of hidden unconnected vehicles, which the proposed algorithm accomplished 90 % of the time.

Identification of appropriate human driver model (IDM) parameters was important to achieving these positive results. Therefore, practical implementation may require online updating of IDM parameters as new vehicles enter the string or the traffic environment changes. Fortunately, individualized IDM parameters were not needed here so data from connected vehicles only are likely sufficient. This feature and assessment on various datasets are planned for further research on this topic.

References

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

[2] R. E. Stern, S. Cui, M. L. Delle Monache, R. Bhadani, M. Bunting, M. Churchill, N. Hamilton, H. Pohlmann, F. Wu, B. Piccoli, B. Seibold, J. Sprinkle, and D. B. Work, "Dissipation of stop-and-go waves via control of autonomous vehicles: Field experiments," *Transp. Res. C*, vol. 89, pp. 205–221, 2018.

[3] T. Ard, L. Guo, R. A. Dollar, A. Fayazi, N. Goulet, Y. Jia, B. Ayalew, and A. Vahidi, "Energy and flow effects of optimal automated driving in mixed traffic: Vehicle-in-the-loop experimental results," *arXiv preprint arXiv:2009.07872*, 2020.

[4] A. D. Ames, J. W. Grizzle, and P. Tabuada, "Control barrier function based quadratic programs with application to adaptive cruise control," in *53rd IEEE Conf. Decision Control*. IEEE, 2014, pp. 6271–6278.

[5] G. Gunter, D. Gloudemans, R. E. Stern, S. McQuade, R. Bhadani, M. Bunting, M. L. Delle Monache, R. Lysecky, B. Seibold, J. Sprinkle, B. Piccoli, and D. B. Work, "Are commercially implemented adaptive cruise control systems string stable?" *arXiv preprint arXiv: 1905.02108*, 2019.

[6] A. Sciarretta and A. Vahidi, *Energy-Efficient Driving of Road Vehicles: Toward Cooperative, Connected, and Automated Mobility*. Springer, 2019.

[7] J. I. Ge and G. Orosz, "Dynamics of connected vehicle systems with delayed acceleration feedback," *Transp. Res. C*, vol. 46, pp. 46–64, 2014.

[8] 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. C*, vol. 91, pp. 335–352, 2018.

[9] I. Mahdinia, R. Arvin, A. J. Khattak, and A. Ghiasi, "Safety, energy, and emissions impacts of adaptive cruise control and cooperative adaptive cruise control," *Transp. Res. Rec.*, vol. 2674, no. 6, pp. 253–267, 2020.

[10] J. M. Maciejowski, *Predictive control: with constraints*. Pearson Education, 2002.

[11] Y. Zhou, M. Wang, and S. Ahn, "Distributed model predictive control approach for cooperative car-following with guaranteed local and string stability," *Transp. Res. B*, vol. 128, pp. 69–86, 2019.

[12] M. A. S. Kamal, T. Hayakawa, and J. Imura, "Road-speed profile for enhanced perception of traffic conditions in a partially connected vehicle environment," *IEEE Trans. Vehicular Technol.*, vol. 67, no. 8, pp. 6824–6837, 2018.

[13] D. Moser, R. Schmied, H. Waschl, and L. del Re, "Flexible spacing adaptive cruise control using stochastic model predictive control," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 1, pp. 114–127, 2017.

[14] L. Zhang and G. Orosz, "Motif-based design for connected vehicle systems in presence of heterogeneous connectivity structures and time delays," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 6, pp. 1638–1651, 2016.

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

[16] X. A. Ji, T. G. Molnár, S. S. Avedisov, and G. Orosz, "Feed-forward neural networks with trainable delay," in *Learning for Dynamics and Control*. PMLR, 2020, pp. 127–136.

[17] S. Le Digabel, "Algorithm 909: NOMAD: Nonlinear optimization with the MADS algorithm," *ACM Trans. Mathematical Software (TOMS)*, vol. 37, no. 4, pp. 1–15, 2011.

[18] M. Pourabdollah, E. Bjärkvik, F. Fürer, B. Lindenberg, and K. Burgdorf, "Calibration and evaluation of car following models using real-world driving data," in *2017 IEEE 20th Int. Conf. Intell. Transp. Syst. (ITSC)*. IEEE, 2017, pp. 1–6.

[19] A. Khodayari, A. Ghaffari, R. Kazemi, and R. Braunstingl, "A modified car-following model based on a neural network model of the human driver effects," *IEEE Trans. Syst., Man, Cybernetics A: Syst. and Humans*, vol. 42, no. 6, pp. 1440–1449, 2012.

[20] 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. C*, vol. 120, p. 102806, 2020.

[21] Y. Nesterov and A. Nemirovskii, *Interior-point polynomial algorithms in convex programming*. SIAM, 1994.

[22] 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, 2019.

[23] R. A. Dollar and A. Vahidi, "Automated vehicles in hazardous merging traffic: A chance-constrained approach," *IFAC-PapersOnLine*, vol. 52, no. 5, pp. 218–223, 2019.