

Energy-efficient Reactive and Predictive Connected Cruise Control

Minghao Shen^{a,*}, R. Austin Dollar^b, Tamas G. Molnar^c, Chaozhe R. He^d, Ardalan Vahidi^e, Gábor Orosz^{a,f}

^aDepartment of Mechanical Engineering, University of Michigan, Ann Arbor, MI 48109, USA

^bGeneral Motors, Concorde, NC 28027, USA

^cDepartment of Mechanical and Civil Engineering, California Institute of Technology, Pasadena, CA 91125, USA

^dPlusAI, Inc., Cupertino, CA 95014, USA

^eDepartment of Mechanical Engineering, Clemson University, Clemson, SC 29634, USA

^fDepartment of Civil and Environmental Engineering, University of Michigan, Ann Arbor, MI 48109, USA

Abstract

In this paper, we propose a framework for the longitudinal control of connected and automated vehicles traveling in mixed traffic consisting of connected and non-connected human-driven vehicles. Reactive and predictive controllers are proposed. Reactive controllers are given by explicit feedback control laws. In predictive controllers, the control input is optimized in a receding-horizon fashion, which depends on the predictions of motions of preceding vehicles. Beyond-line-of-sight information is obtained via vehicle-to-vehicle (V2V) communication, and is utilized in the proposed reactive and predictive controllers. Simulations utilizing real traffic data are used to show that connectivity can bring significant energy savings.

Keywords: connected automated vehicles, V2X connectivity, MPC, traffic flow models

1. 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]. As such, 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 [2]. This shows great potential for improving energy efficiency by optimizing driving profiles.

While human drivers have large variations in their driving behavior [3], 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); see Table. 1. Since energy consumption is mainly related to longitudinal motion, level 1-2 automation can already provide significant energy savings. On one hand, automated vehicles (AV) may optimize their speed profile in advance, taking into consideration the engine and transmission dynamics, and the road elevation along the route [4, 5]. On the other hand, extensive research has focused on optimizing the control input (pedal, brake and gear shift) to follow some optimal driving cycles [6]. 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)* algorithm to react to motion of preceding vehicles in traffic. ACC has a long history tracing back to the 1990s. The controller design usually falls into one of the 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 the 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 prediction of the future motions of neighboring vehicles. While predictions may

*Corresponding author.

Email addresses: mhshen@umich.edu (Minghao Shen), rdollar@clemson.edu (R. Austin Dollar), tmolnar@caltech.edu (Tamas G. Molnar), hchaozhe@umich.edu (Chaozhe R. He), avahidi@clemson.edu (Ardalan Vahidi), orosz@umich.edu (Gábor Orosz)

SAE Level	Execution of steering, acceleration/deceleration	Monitoring of driving environment	Fallback performance of dynamic driving task	System capability (driving modes)
0 (No automation)	Human	Human	Human	N/A
1 (Driver assistance)	Human & system	Human	Human	Some driving modes
2 (Partial automation)	System	Human	Human	Some driving modes
3 (Conditional automation)	System	System	Human	Some driving modes
4 (High automation)	System	System	System	Some driving modes
5 (Full automation)	System	System	System	All driving modes

Table 1: SAE Levels of Vehicle Automation

	Motion Prediction of Remote Vehicle	Motion Planning of Ego Vehicle	Control of Ego Vehicle
Status Sharing	Ego Vehicle	Ego Vehicle	Ego Vehicle
Intent Sharing	Remote Vehicle	Ego Vehicle	Ego Vehicle
Agreement Seeking	Remote Vehicle	Remote Vehicle	Ego Vehicle
Prescriptive Cooperation	Remote Vehicle	Remote Vehicle	Remote Vehicle

Table 2: SAE Levels of Cooperative Driving Automation (CDA)

significantly improve energy efficiency, an accurate prediction is very hard to make without additional information, 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, as well as facilitates cooperation among vehicles in the traffic. SAE categorized cooperative driving automation (CDA) into status-sharing, intent-sharing, agreement-seeking and prescriptive cooperation [7]; see Table 2. Many of the existing research works assume high level of cooperation, e.g., prescriptive cooperation. Assuming that an entire platoon of vehicles are connected and automated, the high level of cooperation enables centralized control over all these vehicles. Such controllers are often referred to as *cooperative adaptive cruise control (CACC)*. Similar to ACC, CACC design can also be categorized into reactive and predictive control [8, 9]. Reactive control tries to synchronize the speed of the platoon, guaranteeing string stability and maintaining desirable headway [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 the system more scalable, distributed control protocol has also been studied [17]. Research has shown that CACC and platooning bring significant energy benefits under different scenarios [18, 19, 20]. However, currently the 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.

The near future of transportation is more likely to evolve into mixed traffic. Controllers that operate under mixed traffic consisting of connected and non-connected vehicles are referred to as *connected cruise control (CCC)*. Only low-level cooperation as status-sharing is assumed and centralized control is not possible. Potentially four kinds of vehicles may participate in the mixed traffic: human-driven vehicle (HV), connected human-driven vehicle (CHV), automated vehicle (AV) and connected and automated vehicle (CAV). Without connectivity, the longitudinal controllers for AVs execute adaptive cruise control. While with connectivity, CAVs may execute more performant controllers, even with low level of cooperation such as status-sharing protocol. 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. Similar to ACC and CACC, CCC can also be categorized into reactive control and predictive control. Reactive CCC (RCCC) takes the V2V information from leading vehicles as reference signals, the objective is still to synchronize the speed in the traffic for string stability and smooth driving [21, 22, 23]. Meanwhile, predictive CCC (PCCC) can incorporate the information of preceding vehicles to make predictions on the motion of the vehicle immediately in the front [24]. 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 illustrated graphically for mixed traffic scenarios containing HVs, AVs, CHVs, and CAVs.

With all these distinctions made, this paper presents contributions to improve energy efficiency in mixed traffic as

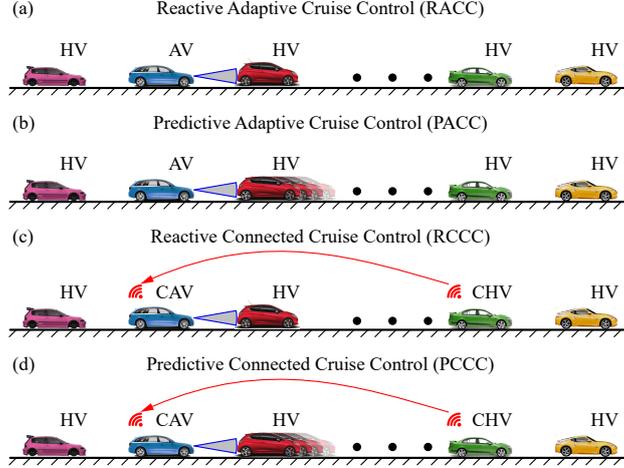


Figure 1: Illustration of longitudinal control strategies for automated vehicles (AVs) and connected automated vehicles (CAVs) traveling in mixed traffic that involves human-driven vehicles (HVs) and connected human-driven vehicles (CHVs). Predictive controllers rely on the predictions on the future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

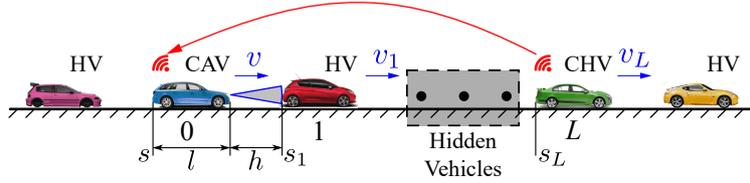


Figure 2: Connected cruise control in mixed traffic consisting of connected and non-connected vehicles.

follows:

- We provide design framework on reactive and predictive control of connected automated vehicles driving in mixed traffic consisting of connected and non-connected vehicles.
- Under both reactive and predictive controller framework, we show the significant energy benefits provided by V2V connectivity. We provide explanations to the energy savings by comparing simulated trajectories.
- We compare the reactive and predictive controllers in three typical scenarios and show the benefit of predictive controllers while utilizing real traffic data.

The remainder of this paper is organized as follows. Section 2 introduces the problem setting and longitudinal dynamics of vehicles. Section 3 discusses the design of energy-efficient reactive controllers including RACC and RCCC. Section 4 discusses predictive controller designs including PACC and PCCC. Section 5 shows the energy benefits of different controller designs with lean penetration of connected vehicles. Section 6 concludes this paper and points out future research directions.

2. 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) is driving on a flat road without elevation change, with the intention to follow human-driven traffic. The longitudinal dynamics of the CAV with respect to its position s and velocity v can be modeled as in [25]:

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

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 different torque on wheels T_w 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

$$\begin{aligned}\dot{s}(t) &= v(t), \\ \dot{v}(t) &= -f(v(t)) + \text{sat}(u(t - \sigma)),\end{aligned}\quad (2)$$

where

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

The model incorporates the delay σ in 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

$$\text{sat}(u) = \min\{\tilde{u}_{\text{max}}, \max\{u_{\text{min}}, u\}\}, \quad (4)$$

$$\tilde{u}_{\text{max}} = \min\{u_{\text{max}}, m_1v + b_1, m_2v + b_2\}, \quad (5)$$

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

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

is applied, where the term \tilde{f} tries to compensate the nonlinear physical effects f in (3). In this article, we assume that perfect compensation is possible and focus on the choice of desired acceleration a_d , which simplifies (1) to

$$\begin{aligned}\dot{s}(t) &= v(t), \\ \dot{v}(t) &= \text{sat}(a_d(t - \sigma)),\end{aligned}\quad (7)$$

Energy consumption is the main interest in this article. It is evaluated with energy consumption per unit mass

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

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 3. These four controllers are detailed in the next two sections and in Algorithms 1-4.

3. 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 only react to the vehicle immediately in the front. RACC determines the desired acceleration a_d as a function of 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. For example, optimal velocity model (OVM) yields the control algorithm

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

where the range policy $V(h)$ determines the desired velocity for headway and the speed policy $W(v_1)$. A common choice of range policy is given in [21] as

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

As is shown in Fig. 3(c), when the headway is less than the stopping distance d , the ego vehicle tends to stay still, while when the headway is larger than $d + \tau v_{\max}$, the ego vehicle intends to travel with maximum speed v_{\max} while not being influenced by the preceding vehicle. For intermediate headway distance, the desired velocity grows with constant gradient $1/\tau$ where τ is referred to as *time headway*. Moreover, the speed policy

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

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

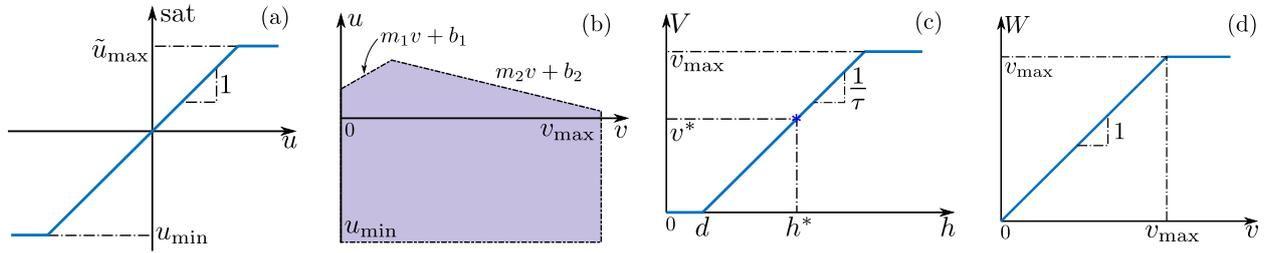


Figure 3: Nonlinearities in vehicle dynamics and optimal velocity model (OVM). (a) Saturation function (4). (b) Acceleration limits (5). (c) Range policy (11). (d) Speed policy (12).

Another widely-used car-following model is the intelligent driver model (IDM) [26]:

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

where the desired headway is calculated using the range policy

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

Here a_0 corresponds to the maximum acceleration, b_0 is the deceleration coefficient, 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

$$H(v) = d + \tau v, \quad (15)$$

which is the inverse of $V(h)$ in (11) when $0 < v < v_{\max}$. Stopping distance d and maximum speed v_{\max} have the same meaning as those in (11).

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 line of sight.

$$a_d = F(h, v, \{s_i\}_{i \in \mathcal{I}}, \{v_i\}_{i \in \mathcal{I}}), \quad (16)$$

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 even if vehicle 1 is not connected vehicle, it can still be sensed by ego vehicle using onboard sensors. For example, one can extend OVM (10) to the RCCC controller

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

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 headway. The controller does not necessarily need to respond to every reference signal immediately. Instead, the controller may wait for a while before responding to the signals from vehicles in the distance [27]. Thus, (17) can be generalized to

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

where the delay σ_i -s are additional design parameters, as oppose to σ in (7) which is a result of powertrain dynamics. In this paper, 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 Algorithm 1 and Algorithm 2, respectively. There has been extensive research on the choice of energy-optimal controller parameters of RACC and RCCC; we refer to [28] for more details.

We remark that safety is often specified as maintaining larger headway than the velocity-dependent minimum headway, e.g., $H_{\min}(v) = d_{\min} + \tau_{\min}v$, cf. (15). In order to ensure that an RACC or a RCCC controller is safe, one may select the controller parameters such that a so-called barrier function maintains positive derivative [29]. Alternatively, one may use so-called control barrier functions to modify RACC and RCCC controllers and make them safe [30, 31]. Such controller tunings and modifications are beyond the scope of this paper and are not discussed here in details.

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  $u = \tilde{f}(v) + a_d$ ;
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  $u = \tilde{f}(v) + a_d$ ;
5 end

```

4. Predictive Controllers

Apart from constructing explicit reactive control laws, one may also formulate control synthesis as an optimization problem, in which the objective functions and constraints are utilized so that control actions optimize the performance. Model predictive control (MPC) is a prevailing choice, which relies on predicting the motion of the vehicle in front of the ego vehicle. An accurate prediction is crucial, for both predictive adaptive cruise control (PACC) illustrated in Fig. 1(b) and the predictive connected cruise control (PCCC) illustrated in Fig. 1(d). One needs to make prediction on the future trajectory of preceding vehicle, and choose optimal action based on the prediction. 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 estimation on the position of preceding vehicle \hat{s}_1 differs in different connectivity structures.

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

$$\begin{aligned}
& \min \int_t^{t+T} \ell(x(\tilde{t}), \hat{x}_1(\tilde{t}), a(\tilde{t})) d\tilde{t}, \\
& \text{s.t. } G_{\text{dynamics}}(x(\tilde{t}), a(\tilde{t})) = 0, \\
& \quad G_{\text{safety}}(x(\tilde{t}), \hat{x}_1(\tilde{t})) \leq 0, \\
& \quad G_{\text{saturation}}(x(\tilde{t}), a(\tilde{t})) \leq 0, \\
& \quad x(t) = x(t), \\
& \quad a(\tilde{t}) = a_d(\tilde{t} - \sigma), \quad \forall \tilde{t} \in [t, t + \sigma].
\end{aligned} \tag{19}$$

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 $x(\tilde{t})$, and our prediction on the future motion of preceding vehicle $\hat{x}_1(\tilde{t})$. Due to the powertrain delay, when $\tilde{t} \leq t + \sigma$, the acceleration at time \tilde{t} is determined by the desired acceleration at $\tilde{t} - \sigma$. So the control input at time t shall determine the acceleration at time $t + \sigma$. In other words, $a_d(t) = a(t + \sigma|t)$.

The objective function of the MPC controller includes penalties on the 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 headway as a function of velocity. For example, in this paper, 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 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

$$\begin{aligned}
s(k+1) &= s(k) + \Delta t v(k) + \frac{1}{2} \Delta t^2 a(k), \\
v(k+1) &= v(k) + \Delta t a(k),
\end{aligned} \tag{20}$$

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

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

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

$$a(k) = a_d(k - q), \tag{22}$$

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

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

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

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

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

In summary, the MPC controller is formulated as follows

$$\begin{aligned}
& \min_{a(0|t), \dots, a(T-1+q|t), \epsilon} q_g \sum_{k=0}^T \left(\hat{h}(k|t) - H(v(k|t)) \right)^2 + q_a \sum_{k=0}^{T-1} a^2(k|t) + q_\epsilon \epsilon, \\
& \text{s.t.} \quad s(k+1|t) = s(k|t) + \Delta t v(k|t) + \frac{1}{2} \Delta t^2 a(k|t), \quad k = 0, \dots, T-1, \\
& \quad v(k+1|t) = v(k|t) + \Delta t a(k|t), \quad k = 0, \dots, T-1, \\
& \quad \hat{h}(k|t) = \hat{s}_1(k|t) - s(k|t) - l, \quad k = 0, \dots, T, \\
& \quad \hat{h}(k|t) - H_{\min}(v(k|t)) - d_{\text{margin}}(k) \geq -\epsilon, \quad k = 0, \dots, T, \\
& \quad 0 \leq v(k|t) \leq v_{\max}, \quad k = 0, \dots, T, \\
& \quad u_{\min} \leq a(k|t), \quad k = 0, \dots, T, \\
& \quad a(k+q|t) \leq m_1 v(k+q|t) + b_1, \quad k = 0, \dots, T, \\
& \quad a(k+q|t) \leq m_2 v(k+q|t) + b_2, \quad k = 0, \dots, T, \\
& \quad s_1(0|t) = s_1(t), \quad s(0|t) = s(t), \quad v(0|t) = v(t), \\
& \quad a(k|t) = a_d(t+k-q), \quad k = 0, \dots, q-1,
\end{aligned} \tag{25}$$

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.

4.1. Predictive Adaptive Cruise Control (PACC)

Accurate prediction of the preceding vehicles is the key to the success of MPC controller. 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 the current speed in the future [32, 33]:

$$\begin{aligned}
\hat{s}_1(\tilde{t}|t) &= s_1(t) + (\tilde{t} - t) v_1(t), \quad \tilde{t} \geq t, \\
\hat{v}_1(\tilde{t}|t) &= v_1(t), \quad \tilde{t} \geq t.
\end{aligned} \tag{26}$$

The corresponding PACC algorithm, which is in discrete time, is shown in Algorithm 3. With connectivity, PCCC may leverage additional information about the preceding vehicle that may potentially bring more energy benefits.

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

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

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

```

4.2. 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.

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  $\hat{n}_h$ ;
4   Apply constant speed prediction on leading vehicle  $L$ 
      
$$\hat{s}_L(t+k|t) = s_L(t) + k\Delta t v_L(t), \quad k = 1, \dots, T,$$

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

5   if  $\hat{n}_h == 0$  then
6     Simulate  $\hat{s}_1(t : t+T|t)$  following the vehicle numbered  $L$ ;
7   else
8     for  $i = 0$  to  $\hat{n}_h$  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  $a(q)$  to powertrain and braking systems;
15  Update with discrete time dynamics (20), (22);
16 end
```

We propose the PCCC control framework detailed Algorithm 4. Compared to the PACC in Algorithm 3, the only change is in the way we predict $\hat{s}_1(t : t+T)$. In PACC, without additional information, we chose to make constant speed assumption. While in PCCC, with the additional information from V2V communication, we can potentially make more accurate prediction on the future motion of preceding vehicles. In this paper, we apply constant speed assumption on the future motion 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}_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 four controllers in Table 3.

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

Table 3: Comparison of cruise control algorithms

4.3. Hidden Vehicle Estimation

The estimation of the number of hidden vehicles is based on historical data of preceding vehicles: s_1, v_1, s_L and v_L , which are recorded during driving. We denote the resulting estimation by \hat{n}_h that approximates the unknown number of hidden vehicles, $n_h = L - 2$.

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

Algorithm 5: Estimation of Number of Hidden Vehicles

Input : History 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^{prev} ;
Output: Estimated number of hidden vehicles \hat{n}_h ;

- 1 $n_{h,\text{max}} = \min \left\{ \hat{n}_h^{\text{prev}} + 1, \left\lceil \frac{s_L(t) - s_1(t)}{h_{\text{min}} + \tau_{\text{min}} v(t)} \right\rceil - 1 \right\}$;
- 2 $n_{h,\text{min}} = \max \left\{ \hat{n}_h^{\text{prev}} - 1, 0 \right\}$;
- 3 **for** $n_h = n_{h,\text{min}}$ **to** $n_{h,\text{max}}$ **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 is the index of the minimum of J .

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

$$\hat{s}_{i+1}^{(n_h)}(t - T_h) - \hat{s}_i^{(n_h)}(t - T_h) = \frac{s_L(t - T_h) - s_1(t - T_h)}{n_h + 1}, \quad i = 2, \dots, n_h + 1, \quad (27)$$

and equal velocity

$$\hat{v}_i(t - T_h) = \frac{v_1(t - T_h) + v_L(t - T_h)}{2}, \quad i = 2, \dots, n_h + 1. \quad (28)$$

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

Second, at the beginning of 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

$$\hat{n}_h^{\text{init}} = \left\lceil \frac{s_L(t) - s_1(t)}{H(v_1(t))} \right\rceil - 1. \quad (29)$$

The estimation is adjusted using Algorithm 5 as more data is collected. At 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 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 . 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^{prev} by $c(n_h)$, so \hat{n}_h^{prev} is more likely to have smaller cost function, and be chosen in the current step.

5. 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.

5.1. 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 is collected in an experiment where all vehicles were connected. The details of the experiment are described in [34].

In this paper, we consider three kinds of qualitatively different datasets: free-flow, step, and congested, as shown in Fig. 4. In the free-flow profile, the drivers are driving close to the speed limit with little variations. In the step profile, the preceding vehicles accelerate from halt. After reaching a steady-state speed, the vehicles maintain the speed for a while, 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.

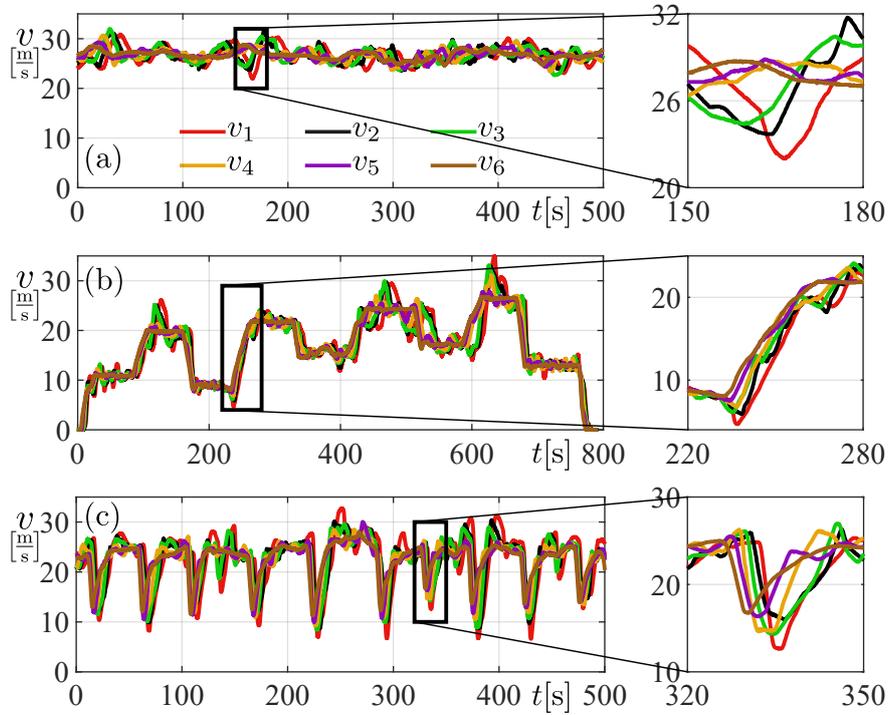


Figure 4: Three qualitatively different speed trajectories from experimental data. (a) Free flow profile. (b) Step profile. (c) Congested profile.

In RACC and PACC, the ego vehicle only responds to 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. The nonlinear physical term defined in (3) is set to

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

The acceleration limits (4), (5), parameters of range policy and speed policy are shown in Table 4, 5, and 6.

In RCCC, we fix $\alpha = 0.4$ [1/s], and the control parameters $\beta_1, \beta_L, \sigma_L$ are optimized using the method introduced in [28] in each of the three dataset types. The optimal parameters are shown in Table 6. The IDM parameters for PCCC are listed in Table. 7, and the corresponding MPC parameters are listed in Table. 8. The identification of IDM parameters are described in detail in Appendix B.

u_{\min} [m/s ²]	m_1 [1/s]	b_1 [m/s ²]	m_2 [1/s]	b_2 [m/s ²]	τ [s]	d [m]	v_{\max} [m/s]
-6	0.285	2	-0.121	4.83	1.67	5	35

Table 4: Acceleration limit of CAV.

Table 5: Parameters of range policy and speed policy.

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

Table 6: Parameters of reactive controllers for $L = 6$, i.e., $n_h = 4$.

	a_0 [m/s ²]	b_0 [m/s ²]	δ [m]	τ [s]	d [m]	v_{\max} [m/s]
Free flow	0.684	2.9693	3.3066	0.7154	5.0001	36
Step	2.2868	8.5	3	0.9282	5	32.8682
Congested	2.5732	8.5	4.3393	0.6409	5.067	36

Table 7: IDM parameters identified from free-flow, step and congested datasets.

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

Table 8: Parameters of predictive controllers.

5.2. Benefits of Connectivity

In this section, we show that connectivity brings great energy benefit. We compare the energy consumption with and without connectivity, and provide an explanation to the observed energy savings by comparing simulated trajectories as well as prediction results. The optimization problem (25) is formulated in MATLAB with Yalmip [35] and the quadratic programming problems are solved with Gurobi solver [36].

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. 5. In free flow scenario, the speed variations of preceding vehicles are small, so there is little difference on the energy efficiency of the four controllers. While in the step and congested scenarios, controllers with connectivity (RCCC and PCCC) save significant amount of energy compared to those without connectivity (RACC and PACC). Compared to RACC, RCCC saves 18.1% energy in step scenario and 29.2% energy in congested scenario. Compared to PACC, PCCC saves 12.0% energy in step scenario and 30.0% energy in congested scenario. It is also worth noting that RCCC and PCCC achieve similar energy consumption in free-flow and step scenarios, but PCCC consumes 11.9% less energy compared to RCCC in congested scenario.

In Fig. 6, we plot the 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 headway-speed relationship. In addition, we plot the trajectories of headway, velocity and acceleration in congested scenario in Fig. 7, 8. Without connectivity, vehicles suffer from abrupt brakings and accelerations, which leads to excessive energy consumption. With information from V2V connectivity, both reactive and predictive controllers can obtain smaller speed variations while maintaining reasonable headway. We also observe that predictive controllers tend to keep larger headways compared to reactive controllers.

For predictive controllers, we can directly show the improvement of prediction due to additional information from V2V connectivity. Figure 9 shows the error between the prediction and the ground truth

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

for PACC and PCCC. In both cases the accuracy is similar up to about 4 seconds time horizon. However, with additional information from V2V connectivity, PCCC can achieve better prediction on a larger horizon.

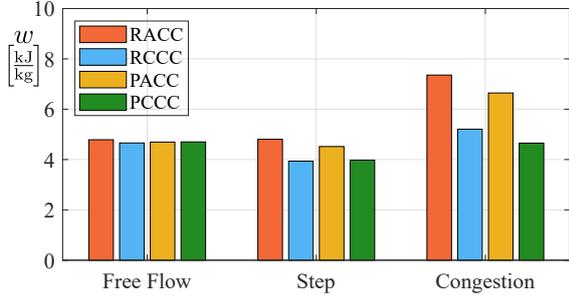


Figure 5: Comparison of energy consumption of RACC, RCCC, PACC, PCCC.

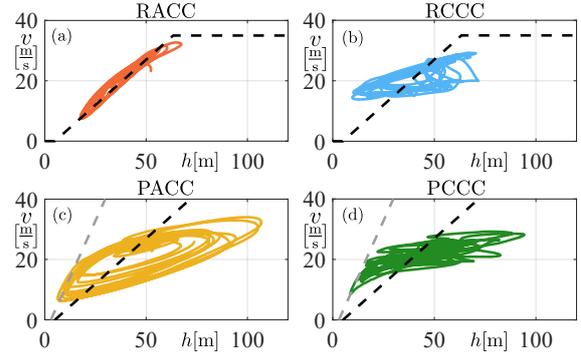


Figure 6: Phase portrait for RACC, RCCC, PACC, PCCC. Desired range policy is plotted in black dashed line and the safety constraints for predictive controllers are indicated by grey dashed line.

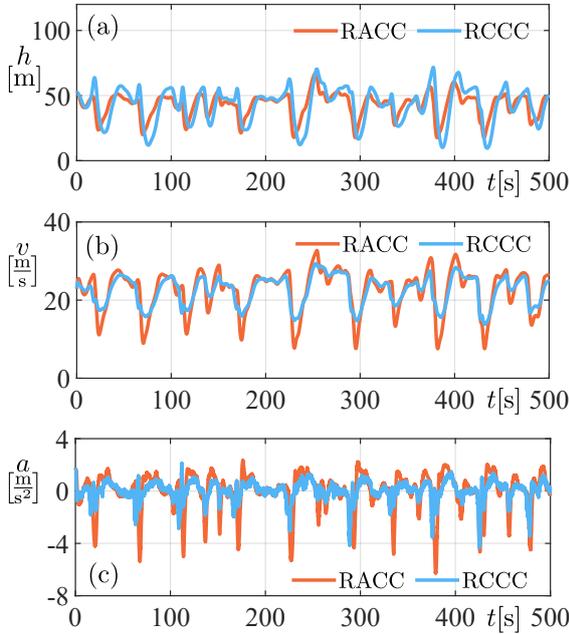


Figure 7: Comparison of RACC and RCCC in congested scenario for $L = 6$, i.e., $n_h = 4$. (a) Headway; (b) Speed; (c) Acceleration.

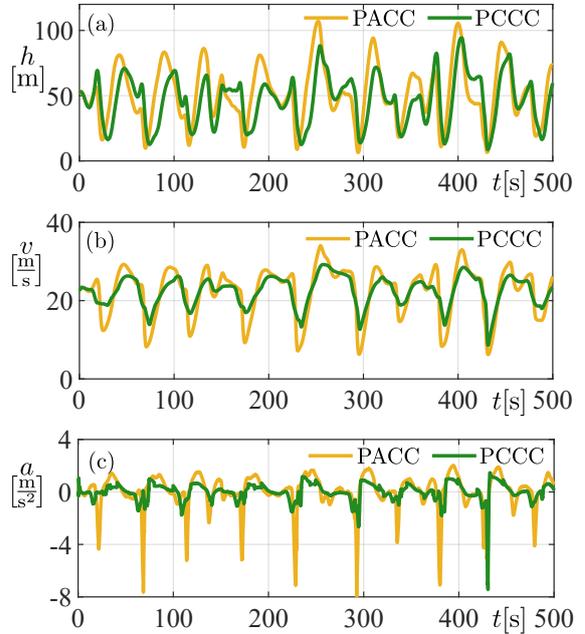


Figure 8: Comparison of PACC and PCCC in congested scenario for $L = 6$, i.e., $n_h = 4$. (a) Headway; (b) Speed; (c) Acceleration.

While 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 when using data from vehicle L by connectivity.

Figure 10 further illustrates the motion prediction in a deceleration-acceleration cycle. We show the motion predictions at $t = 288$ [s] (during deceleration), $t = 294$ [s] (at the end of deceleration), $t = 300$ [s] (during acceleration). At each time instance, we plot the ground truth positions and speeds of vehicles 1 and L by solid lines, and the corresponding predictions by dashed lines. Meanwhile we show the prediction of hidden vehicles by grey dashed lines. At $t = 288$ [s] the IDM model is able to predict the deceleration at the initial 4 [s] horizon, but fails to predict the

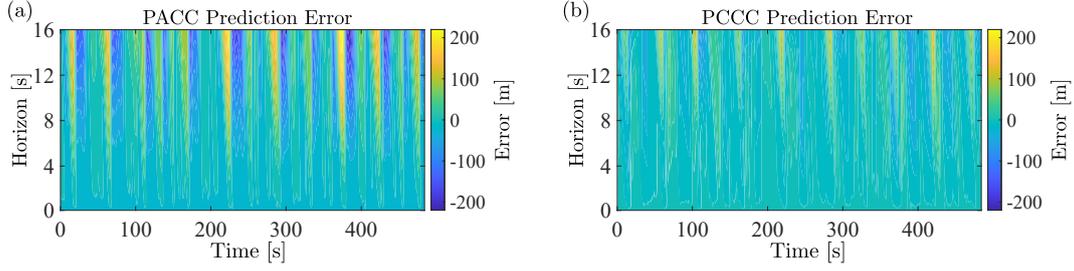


Figure 9: (a) Prediction error for vehicle 1 of PACC based on constant speed assumption. (b) Prediction error of PCCC based on car-following model.

harsh deceleration and the acceleration afterwards; 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 $t = 294$ [s] and panels (f), (g) for $t = 300$ [s].

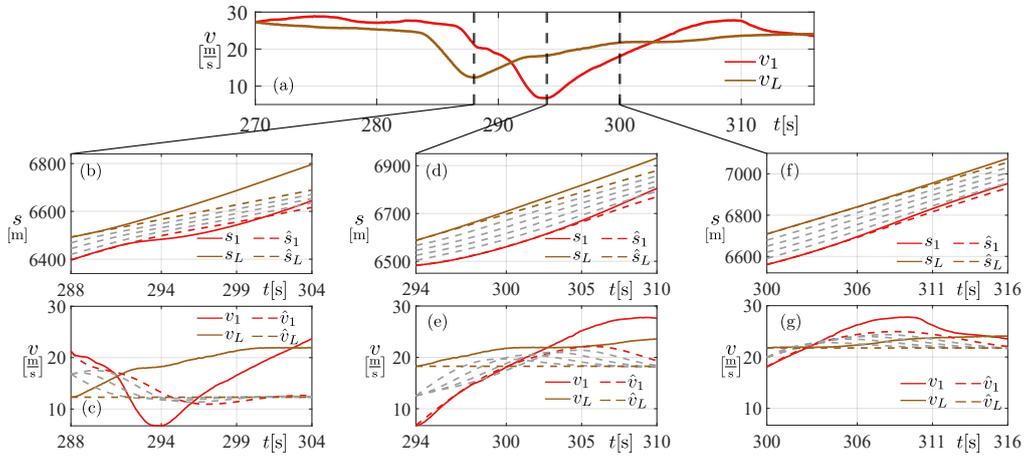


Figure 10: Comparison of IDM prediction and the ground truth. Dashed curves show the predicted future trajectories, while solid curves show the trajectory data collected from experiments.

In practice, the number of hidden vehicles is unknown. Fig. 11 shows the sensitivity to the number of hidden vehicles n_h in free-flow, step, and congested scenarios. In general, with connectivity, significant energy can be saved in all three scenarios. However, in free flow scenarios, connection to vehicles farther ahead associated with larger n_h may increase the energy consumption for PCCC. While in other scenarios, connecting to farther vehicles helps reduce the energy consumption. Especially in congested scenario, connecting to a vehicle in the distance ($n_h = 4$) saves 13.0% energy compared to connecting to a nearby vehicle ($n_h = 1$) in case of RACC, and 16.8% in case of PCCC.

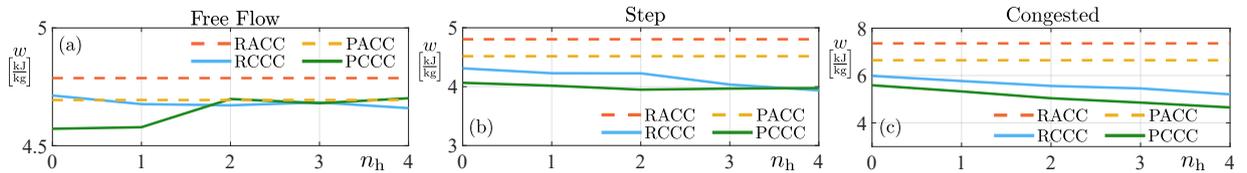


Figure 11: Comparison of the energy consumption of RACC, PACC, RCCC and PCCC as a function of the number of hidden vehicles for the three traffic datasets in Fig 4.

However, with more hidden vehicles involved in the prediction, more uncertainties are introduced. 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., vehicle 2 and 3 are hidden vehicles, it is hard to make a precise predictions using car-following model. In conclusion, when the speeds of the leading vehicle and the preceding vehicle are highly correlated, connecting to vehicles farther ahead usually helps saving energy until the increasing uncertainty undermines the correlation and causes more energy consumption.

6. Conclusions and Future Work

In this paper, we proposed a framework for longitudinal control design for 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 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 the energy efficiency. In reactive controllers, the controller parameters were optimized according to the observed data. In predictive controllers, the number of hidden 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 proposed control framework can accommodate various engineering specifications, for example, different implementations of reactive control law, estimation of the number of hidden vehicles, or motion predictors. The framework is also applicable to various kinds of vehicles including internal-combustion engine vehicles, electric vehicles, or hybrid electric vehicles. While, the design framework in this paper only considers longitudinal motion in single lane, it is promising to consider both longitudinal and lateral motion on multiple lanes in future research.

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 automated vehicle [37]. 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 $k = 0, \dots, T-1$. Using the notation $x_1(k) = [s_1(k), v_1(k)]^\top$, (20) can be written as

$$x_1(k+1) = A x_1(k) + B a_1(k), \quad (\text{A.1})$$

where

$$A = \begin{bmatrix} 1 & \Delta t \\ 0 & 1 \end{bmatrix}, \quad B = \begin{bmatrix} \frac{1}{2}\Delta t^2 \\ \Delta t \end{bmatrix}. \quad (\text{A.2})$$

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

$$x_1 = \hat{A} x_1(0) + \hat{B} a_1, \quad (\text{A.3})$$

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

$$\hat{A} = \begin{bmatrix} A \\ A^2 \\ \vdots \\ A^{T-1} \end{bmatrix}, \quad \hat{B} = \begin{bmatrix} B & 0 & \dots & 0 \\ AB & B & \dots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ A^{T-2}B & A^{T-3}B & \dots & B \end{bmatrix}. \quad (\text{A.4})$$

We split deterministic and stochastic part of the acceleration profile as

$$a_1 = \hat{a}_1 + \tilde{a}_1, \quad (\text{A.5})$$

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

$$\tilde{a}_1 = [1, 1, \dots, 1]^\top e_{a_1}, \quad e_{a_1} \sim \mathcal{N}(0, \sigma_{a_1}^2). \quad (\text{A.6})$$

These result in

$$\hat{B} a_1 = \hat{B} \bar{a}_1 + \hat{B} \tilde{a}_1 = \hat{B} \bar{a}_1 + \underbrace{\begin{bmatrix} B \\ AB + B \\ \vdots \\ A^{T-2}B + A^{T-3}B + \dots + B \end{bmatrix}}_{\hat{B}} e_{a_1}. \quad (\text{A.7})$$

Thus, we can derive the distribution of x_1 as

$$x_1 \sim \mathcal{N}(\hat{A} x_1(0) + \hat{B} \bar{a}_1, \hat{B} \hat{B}^\top \sigma_{a_1}^2). \quad (\text{A.8})$$

Similar to the acceleration we may also split the state as

$$x_1 = \hat{x}_1 + \tilde{x}_1, \quad (\text{A.9})$$

where

$$\hat{x}_1 = \hat{A} x_1(0) + \hat{B} \bar{a}_1, \quad (\text{A.10})$$

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

$$\tilde{x}_1 \sim \mathcal{N}(0, \hat{B} \hat{B}^\top \sigma_{a_1}^2), \quad (\text{A.11})$$

represents the stochastic part. 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

$$\tilde{s}_1(k) \sim \mathcal{N}(0, \sigma_{s_1}^2(k)), \quad \sigma_{s_1}^2(k) = b_k \sigma_{a_1}^2. \quad (\text{A.12})$$

where b_k is the element of matrix $\hat{B} \hat{B}^\top$ 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

$$\mathbb{P}[s_1(k) - s(k) - l - H_{\min}(v(k)) \geq 0] \geq \alpha(k), \quad (\text{A.13})$$

which can be rewritten as

$$\mathbb{P}[\tilde{s}_1(k) \geq s(k) + H_{\min}(v(k)) + l - \hat{s}_1(k)] \geq \alpha(k). \quad (\text{A.14})$$

Using the cumulative density function $\Phi(z) = \frac{1}{\sqrt{2\pi}} \int_{-\infty}^z e^{-\frac{t^2}{2}} dt$ of the standard Gaussian distribution we obtain

$$1 - \Phi\left(\frac{s(k) + H_{\min}(v(k)) + l - \hat{s}_1(k)}{\sigma_{s_1}(k)}\right) \geq \alpha(k), \quad (\text{A.15})$$

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

$$\hat{s}_1(k) - s(k) - l - H_{\min}(v(k)) - \sigma_{s_1}(k) \Phi^{-1}(\alpha(k)) \geq 0. \quad (\text{A.16})$$

Therefore, we define the safety margin

$$d_{\text{margin}}(k) = \max\{0, \sigma_{s_1}(k) \Phi^{-1}(\alpha(k))\}. \quad (\text{A.17})$$

In this paper, we choose a linearly decreasing $\alpha(k)$ with $\alpha(1) = 0.99$ and $\alpha(K) = 0.5$, where $K\Delta t = 10$ [s], to gradually loosen the probabilistic constraint within the prediction horizon. The corresponding safety margin is plotted in Fig. A.12 as a function of time.

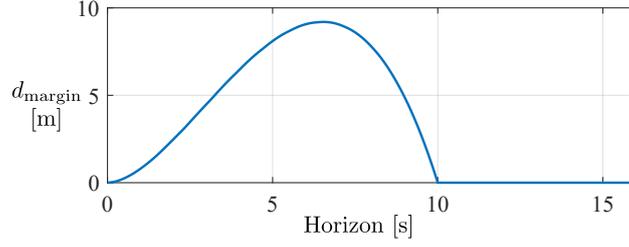


Figure A.12: Safety margin within prediction horizon.

Appendix B. IDM parameter identification

The IDM parameters are obtained via solving the optimization problem

$$\begin{aligned}
 \min_{a_0, b_0, \delta, \tau, h_{st}, v_{\max}} \quad & \frac{1}{N_v} \sum_{i=1}^{N_v} \sqrt{\frac{1}{N_s} \sum_{k=1}^{N_s} (\hat{h}_i(k) - h_i(k))^2}, \\
 \text{s.t.} \quad & 0.1 \leq a_0 \leq 4, \\
 & 0.1 \leq b_0 \leq 8.5, \\
 & 3 \leq \delta \leq 5, \\
 & 0.1 \leq \tau \leq 4, \\
 & 5 \leq d \leq 10, \\
 & 30 \leq v_{\max} \leq 36.
 \end{aligned} \tag{B.1}$$

In each dataset, there are 6 human drivers and the cost function averages the performance of IDM model on all $N_v = 5$ 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 length N_s . We set some lower bounds and upper bounds on the IDM parameters considering their physical meanings. We use NOMAD optimizer [38] to solve the optimization problem (B.1). The resulting IDM parameters are shown in Table 7.

Acknowledgements

This research was supported by the University of Michigan’s Center for Connected and Automated Transportation through the US DOT grant 69A3551747105.

References

- [1] A. Sciarretta, A. Vahidi, *Energy-Efficient Driving of Road Vehicles*, Springer, 2020.
- [2] B. Saltsman, Impacts of connectivity and automation on vehicle operations, in: *Global Symposium on Connected Vehicle and Infrastructure*, University of Michigan Transportation Research Institute, 2014.
- [3] M. Zarkadoula, G. Zoidis, E. Tritopoulou, Training urban bus drivers to promote smart driving: A note on a greek eco-driving pilot program, *Transportation Research Part D* 12 (6) (2007) 449–451.
- [4] C. R. He, H. Maurer, G. Orosz, Fuel Consumption Optimization of Heavy-Duty Vehicles With Grade, Wind, and Traffic Information, *Journal of Computational and Nonlinear Dynamics* 11 (6), 061011.
- [5] C. R. He, A. Alan, T. G. Molnár, S. S. Avedisov, A. H. Bell, R. Zukouski, M. Hunkler, J. Yan, G. Orosz, Improving fuel economy of heavy-duty vehicles in daily driving, in: *2020 American Control Conference (ACC)*, IEEE, 2020, pp. 2306–2311.
- [6] A. Sciarretta, G. De Nunzio, L. L. Ojeda, Optimal ecodriving control: Energy-efficient driving of road vehicles as an optimal control problem, *IEEE Control Systems Magazine* 35 (5) (2015) 71–90. doi: 10.1109/MCS.2015.2449688.
- [7] SAE, Taxonomy and definitions for terms related to cooperative driving automation for on-road motor vehicles, Tech. rep., Technical Report (2021).
- [8] S. E. Shladover, C. Nowakowski, X.-Y. Lu, R. Ferlis, Cooperative adaptive cruise control: Definitions and operating concepts, *Transportation Research Record* 2489 (1) (2015) 145–152. doi: 10.3141/2489-17.

- [9] Z. Wang, G. Wu, M. J. Barth, A review on cooperative adaptive cruise control (CACC) systems: Architectures, controls, and applications, in: 2018 21st International Conference on Intelligent Transportation Systems (ITSC), 2018, pp. 2884–2891. doi:10.1109/ITSC.2018.8569947.
- [10] J. Ploeg, N. van de Wouw, H. Nijmeijer, Lp string stability of cascaded systems: Application to vehicle platooning, *IEEE Transactions on Control Systems Technology* 22 (2) (2014) 786–793. doi:10.1109/TCST.2013.2258346.
- [11] Y. Zheng, S. E. Li, K. Li, W. Ren, Platooning of connected vehicles with undirected topologies: Robustness analysis and distributed h-infinity controller synthesis, *IEEE Transactions on Intelligent Transportation Systems* 19 (5) (2018) 1353–1364. doi:10.1109/TITS.2017.2726038.
- [12] Y. Zheng, S. Eben Li, J. Wang, D. Cao, K. Li, Stability and scalability of homogeneous vehicular platoon: Study on the influence of information flow topologies, *IEEE Transactions on Intelligent Transportation Systems* 17 (1) (2016) 14–26. doi:10.1109/TITS.2015.2402153.
- [13] Y. Zheng, S. E. Li, K. Li, F. Borrelli, J. K. Hedrick, Distributed model predictive control for heterogeneous vehicle platoons under unidirectional topologies, *IEEE Transactions on Control Systems Technology* 25 (3) (2017) 899–910. doi:10.1109/TCST.2016.2594588.
- [14] E. van Nunen, J. Reinders, E. Semsar-Kazerouni, N. van de Wouw, String stable model predictive cooperative adaptive cruise control for heterogeneous platoons, *IEEE Transactions on Intelligent Vehicles* 4 (2) (2019) 186–196. doi:10.1109/TIV.2019.2904418.
- [15] J. Guanetti, Y. Kim, F. Borrelli, Control of connected and automated vehicles: State of the art and future challenges, *Annual Reviews in Control* 45 (2018) 18–40.
- [16] S. V. D. Hoef, J. Mårtensson, D. V. Dimarogonas, K. H. Johansson, A predictive framework for dynamic heavy-duty vehicle platoon coordination, *ACM Transactions on Cyber-Physical Systems* 4 (1) (2019) 1–25.
- [17] M. H. Basiri, B. Ghoghogh, N. L. Azad, S. Fischmeister, F. Karray, M. Crowley, Distributed nonlinear model predictive control and metric learning for heterogeneous vehicle platooning with cut-in/cut-out maneuvers, in: 59th IEEE Conference on Decision and Control (CDC), 2020, pp. 2849–2856. doi:10.1109/CDC42340.2020.9304180.
- [18] L. Bertoni, J. Guanetti, M. Basso, M. Masoero, S. Cetinkunt, F. Borrelli, An adaptive cruise control for connected energy-saving electric vehicles, *IFAC-PapersOnLine* 50 (1) (2017) 2359–2364, 20th IFAC World Congress. doi:https://doi.org/10.1016/j.ifacol.2017.08.425.
- [19] Y. Bian, C. Du, M. Hu, S. E. Li, H. Liu, C. Li, Fuel economy optimization for platooning vehicle swarms via distributed economic model predictive control, *IEEE Transactions on Automation Science and Engineering* (2021) 1–13. doi:10.1109/TASE.2021.3128920.
- [20] Q. Lin, S. E. Li, X. Du, X. Zhang, H. Peng, Y. Luo, K. Li, Minimize the fuel consumption of connected vehicles between two red-signalized intersections in urban traffic, *IEEE Transactions on Vehicular Technology* 67 (10) (2018) 9060–9072. doi:10.1109/TVT.2018.2864616.
- [21] G. Orosz, Connected cruise control: modelling, delay effects, and nonlinear behaviour, *Vehicle System Dynamics* 54 (8) (2016) 1147–1176. doi:10.1080/00423114.2016.1193209.
- [22] L. Zhang, G. Orosz, Motif-based design for connected vehicle systems in presence of heterogeneous connectivity structures and time delays, *IEEE Transactions on Intelligent Transportation Systems* 17 (6) (2016) 1638–1651. doi:10.1109/TITS.2015.2509782.
- [23] L. Zhang, G. Orosz, Consensus and disturbance attenuation in multi-agent chains with nonlinear control and time delays, *International Journal of Robust and Nonlinear Control* 27 (5) (2017) 781–803.
- [24] R. A. Dollar, T. G. Molnár, A. Vahidi, G. Orosz, MPC-based connected cruise control with multiple human predecessors, in: 2021 American Control Conference (ACC), 2021, pp. 405–411. doi:10.23919/ACC50511.2021.9483272.
- [25] T. Ersal, I. Kolmanovsky, N. Masoud, N. Ozay, J. Scruggs, R. Vasudevan, G. Orosz, Connected and automated road vehicles: state of the art and future challenges, *Vehicle System Dynamics* 58 (5) (2020) 672–704. doi:10.1080/00423114.2020.1741652.
- [26] M. Treiber, A. Hennecke, D. Helbing, Congested traffic states in empirical observations and microscopic simulations, *Physical review E* 62 (2) (2000) 1805.
- [27] M. Shen, T. G. Molnár, C. R. He, A. H. Bell, M. Hunkler, D. Oppermann, R. Zukouski, J. Yan, G. Orosz, Saving energy with delayed information in connected vehicle systems, in: 2021 American Control Conference (ACC), IEEE, 2021, pp. 1625–1630.
- [28] M. Shen, C. R. He, T. Molnar, A. H. Bell, G. Orosz, Energy-efficient connected cruise control with lean penetration of connected vehicles, arXiv preprint arXiv:2205.03473.
- [29] C. R. He, G. Orosz, Safety guaranteed connected cruise control, in: 21st IEEE International Conference on Intelligent Transportation Systems, 2018, pp. 549–554.
- [30] A. D. Ames, X. Xu, J. W. Grizzle, P. Tabuada, Control barrier function based quadratic programs for safety critical systems, *IEEE Transactions on Automatic Control* 62 (8) (2017) 3861–3876. doi:10.1109/TAC.2016.2638961.
- [31] A. Alan, A. J. Taylor, C. R. He, A. D. Ames, G. Orosz, Control barrier functions and input-to-state safety with application to automated vehicles, arXiv preprint arXiv:2206.03568.
- [32] C. R. He, G. Orosz, Saving fuel using wireless vehicle-to-vehicle communication, in: 2017 American control conference (ACC), IEEE, 2017, pp. 4946–4951.
- [33] C. R. He, J. I. Ge, G. Orosz, Fuel efficient connected cruise control for heavy-duty trucks in real traffic, *IEEE Transactions on Control Systems Technology* 28 (6) (2020) 2474–2481. doi:10.1109/TCST.2019.2925583.
- [34] J. I. Ge, S. S. Avedisov, C. R. He, W. B. Qin, M. Sadeghpour, G. Orosz, Experimental validation of connected automated vehicle design among human-driven vehicles, *Transportation Research Part C* 91 (2018) 335–352. doi:https://doi.org/10.1016/j.trc.2018.04.005.
- [35] J. Löfberg, YALMIP: A toolbox for modeling and optimization in matlab, in: In Proceedings of the CACSD Conference, Taipei, Taiwan, 2004.
- [36] Gurobi Optimization, LLC, Gurobi Optimizer Reference Manual (2022). URL <https://www.gurobi.com>
- [37] T. Ard, R. A. Dollar, A. Vahidi, Y. Zhang, D. Karbowski, Microsimulation of energy and flow effects from optimal automated driving in mixed traffic, *Transportation Research Part C* 120 (2020) 102806.
- [38] S. Le Digabel, Algorithm 909: NOMAD: Nonlinear optimization with the mads algorithm, *ACM Transactions on Mathematical Software (TOMS)* 37 (4) (2011) 1–15.