Data-Driven Combined Longitudinal and Lateral Control for the Car Following Problem

Leilei Cui, *Member, IEEE*, Sayan Chakraborty, *Graduate Student Member, IEEE*,
Kaan Ozbay, *Member, IEEE*, and Zhong-Ping Jiang, *Fellow, IEEE*

***Abstract*— This article studies the problem of data-driven combined longitudinal and lateral control of autonomous vehicles (AVs) such that the AV can stay within a safe but minimum distance from its leading vehicle and, at the same time, in the lane. Most of the existing methods for combined longitudinal and lateral control are either model-based or developed by purely data-driven methods such as reinforcement learning. Traditional model-based control approaches are insufficient to address the adaptive optimal control design issue for AVs in dynamically changing environments and are subject to model uncertainty. Moreover, the conventional reinforcement learning approaches require a large volume of data, and cannot guarantee the stability of the vehicle. These limitations are addressed by integrating the advanced control theory with reinforcement learning techniques. To be more specific, by utilizing adaptive dynamic programming (ADP) techniques and using the motion data collected from the vehicles, a policy iteration algorithm is proposed such that the control policy is iteratively optimized in the absence of the precise knowledge of the AV's dynamical model. Furthermore, the stability of the AV is guaranteed with the control policy generated at each iteration of the algorithm. The efficiency of the proposed approach is validated by the integrated simulation of SUMO and CommonRoad.**

***Index Terms*— Adaptive dynamic programming (ADP), combined longitudinal and lateral control, connected vehicles.**

I. Introduction

THE increased demand for car travel cannot be met by the existing transportation infrastructure without major expansion projects that are both very expensive and environmentally unsustainable. To address the congestion and safety problems caused by the increased demand for car-based trips, one approach is to reduce the headway between vehicles, thereby increasing road capacity without the need to build new highways. Cooperative adaptive cruise control (CACC) is an effective technology to reduce the intervehicle distance with the feedback of the inter-vehicle distance and the relative velocity [1], [2]. For CACC, most existing methods focus on the longitudinal control of autonomous vehicles (AVs). These approaches assume that the vehicles move along a straight road segment [3], [4], [5], [6]. However, real-world roads are often curved. To safely operate on roadways with realistic geometries, vehicles must not only maintain the desired inter-vehicle distance but also stay within their lanes. Therefore, combined longitudinal and lateral control of AVs is a significant research topic that needs further investigation.

To achieve combined longitudinal and lateral control for AVs, one method is to decompose the system into two independent subsystems: the longitudinal and lateral subsystems [7], and design controllers for each separately. For example, in [8], a proportional–integral controller was proposed for longitudinal movement, while the lateral dynamics were modeled as a linear time-varying system, with model predictive control (MPC) adopted for lateral control. In [9], based on the Lyapunov approach, controllers were designed independently for both directional movements. Similarly, [10], [11] adopted linear time-invariant models for both longitudinal and lateral movements, applying the MPC framework to control the vehicle in each direction separately. Kianfar et al. [12] also applied the MPC method to both types of movement, considering string stability for longitudinal movement. Both the Lyapunov approach and sliding mode control were applied for longitudinal and lateral control in [13]. Another approach involves designing the longitudinal controller using CACC techniques [5], [6], [14], [15] to mitigate stop-and-go waves along the upstream of the platoon. Additionally, longitudinal control of connected and AVs (CAVs) with guaranteed string stability was studied in [16] and [17]. The lateral controller can be designed using lane-keeping techniques [18], [19], [20], [21], [22] based on vehicle lateral dynamics [23], [24]. However, one major limitation of this approach is that, in reality, the lateral and longitudinal movements of vehicles are coupled, and ignoring these coupling effects may compromise AV performance [25].

To address the coupling effects, the longitudinal and lateral subsystems should be modeled as a fully coupled system,

Received 17 May 2024; revised 21 October 2024; accepted 3 January 2025. Date of publication 11 February 2025; date of current version 25 April 2025. This work was supported in part by NSF under Grant ECCS-2210320 and Grant CNS-2227153; and in part by the Connected Cities for Smart Mobility toward Accessible and Resilient Transportation (C2SMART), the Tier 1 University Center Awarded by U.S. Department of Transportation through the University Transportation Centers Program. Recommended by Associate Editor C. Nielsen. *(Corresponding author: Leilei Cui.)*

Leilei Cui is with the Massachusetts Institute of Technology, Cambridge, MA 02139 USA (e-mail: llcui@mit.edu).

Sayan Chakraborty is with the Control and Networks Laboratory, Department of Electrical and Computer Engineering, Tandon School of Engineering, New York University, Brooklyn, NY 11201 USA (e-mail: sc8804@nyu.edu).

Kaan Ozbay is with the C2SMARTER Center, Department of Civil and Urban Engineering, Tandon School of Engineering, New York University, Brooklyn, NY 11201 USA (e-mail: kaan.ozbay@nyu.edu).

Zhong-Ping Jiang is with the Control and Networks Laboratory, Department of Electrical and Computer Engineering, Department of Civil and Urban Engineering, Tandon School of Engineering, New York University, Brooklyn, NY 11201 USA (e-mail: zjiang@nyu.edu).

Digital Object Identifier 10.1109/TCST.2025.3539216

with corresponding controllers designed simultaneously. For example, MPC methods were applied in [26], [27], and [28], while the Lyapunov and backstepping approaches were used in [29] to design a nonlinear controller for the coupled longitudinal and lateral dynamics. In [30] and [31], a nonlinear controller was designed using the Lyapunov approach. The combined longitudinal and lateral control methods mentioned above rely on lane detection. An alternative approach is to directly follow a preceding vehicle that is driving properly, without relying on lane detection. Ploeg et al. [4] proposed a look-ahead approach to follow the leading vehicle. The undesirable cutting-corner phenomenon may occur when the following vehicle follows the leading vehicle directly. To overcome the cutting-corner limitation, an extended look-ahead approach was proposed in [32] and [33]. These two papers considered only the vehicle's kinematic model, but at high speeds, the vehicle's dynamics cannot be neglected. Wang et al. [34], applied the extended look-ahead approach to achieve the combined longitudinal and lateral control considering the nonlinear dynamics of the following AV. All of these combined control methods are based on the assumption of an accurate dynamic model of the AV, and thus, their performance heavily depends on the model's accuracy. In reality, such an accurate model is hard to obtain due to the complex tire friction dynamics and great variability among different vehicles even if they belong to the same vehicle class.

These major limitations can be addressed through learning-based control approaches. Recently, the rapid development of deep learning methods has inspired numerous studies on learning-based vehicle control, as reviewed in [35] and many related references therein. However, deep learning methods require a large amount of data and cannot guarantee the stability of AVs under the learned control policy. Compared to conventional reinforcement learning approaches, adaptive dynamic programming (ADP) offers an effective learning-based method for finding near-optimal control policies without requiring precise knowledge of system dynamics [36], [37], [38]. By collecting input-state data along the system's trajectories and iteratively applying this data, ADP can find near-optimal controllers. In [36] and [37], it was theoretically shown that each iteration yields a suboptimal controller with improved performance, and as the iterations tend to infinity, these suboptimal controllers converge to the optimal one. A key advantage of ADP over traditional reinforcement learning algorithms is that it guarantees stability and performance for the closed-loop system. This makes ADP particularly attractive in transportation, where safety is a top priority. In [39], [40], [41] and [42], ADP was applied to design longitudinal controllers, where a linearized model of AVs was used, and only local stability properties were established for the closed-loop vehicle system.

In this article, we depart from linear thinking and will study the combined longitudinal and lateral control problem for AVs modeled by nonlinear systems. Specifically, we propose a data-driven adaptive optimal controller design for the nonlinear vehicle model without assuming exact knowledge of the model. A key strategy is to leverage the adaptive learning methodology developed in [43], which integrates ADP with output regulation theory [44]. First, considering the nonlinear dynamics and utilizing the extended look-ahead approach [32], [33], the nonlinear dynamics of the tracking error between the following and leading vehicles is derived. To minimize the tracking error as quickly as possible while reducing the energy consumption of the AV, the Hamilton–Jacobi–Bellman (HJB) equation [45] is adopted to solve the corresponding optimal control problem. Then, a model-based policy iteration algorithm is proposed to iteratively optimize the control policy. Finally, by the ADP technique, a two-phase learning-based policy iteration algorithm is proposed using the motion data from the leading and following vehicles. In particular, the first phase is to design a feedforward controller such that the closed-loop system reaches the equilibrium when the tracking error is zero. The second phase is to obtain the optimal feedback controller by the real-time and historical motion data. The integrated simulations of SUMO [46] and CommonRoad [47] are applied to evaluate the performance of the proposed method under different traffic scenarios, e.g., driving on different shapes of roads, string stability evaluation, robustness evaluation, and adaptivity to different road conditions. The main contributions of this article are summarized as follows.

1) Considering the coupling effects between the longitudinal and lateral movements, a nonlinear controller is designed for the AV without linearizing the dynamics of the AV.
2) A data-driven solution is proposed for the combined longitudinal and lateral control of the AV. In particular, via ADP techniques, a near-optimal controller is learned from the motion data without assuming the precise knowledge of the vehicle model.
3) Comprehensive simulations validate the efficacy of the proposed approach under various scenarios.

The rest of the article is organized as follows. In Section II, the nonlinear dynamics of the following vehicle is derived. Section III presents the error system related to longitudinal and lateral control. In Section IV, the optimal control problem for combined longitudinal and lateral control is formulated, and a model-based policy iteration approach is proposed to iteratively optimize the control policy. In Section V, a two-phase data-driven policy iteration algorithm is introduced. Numerical results using a combination of SUMO and CommonRoad simulations are provided in Section VI. Finally, concluding remarks are offered in Section VII.

Notations: $|\cdot|$ denotes the Euclidean norm of a vector or the induced 2-norm of a matrix. $\|\cdot\|_\infty$ denotes the supreme norm of a function. $\|\cdot\|_{\ell_2}$ denotes the norm of a square-summable sequence. $\otimes$ denotes the Kronecker product. $\text{vec}(A) = [a_1^T, \ldots, a_n^T]^T$, where $A \in \mathbb{R}^{m\times n}$ and $A = [a_1, \ldots, a_n]$. $[a]_i$ denotes the ith entry of the vector a, and $[A]_i$ denotes the ith row of the matrix A.

II. Model Description and Problem Formulation

A. Model Description of the Leading Vehicle

In Fig. 1, vehicle F is the following AV, and vehicle L is the preceding/leading vehicle. The constant radius of the path

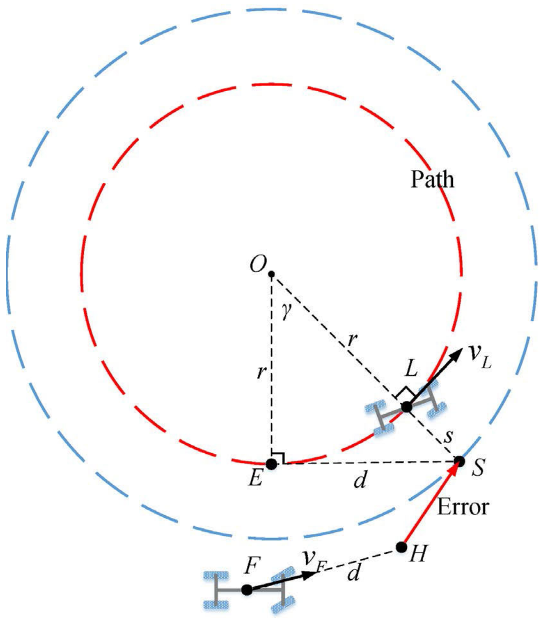


Fig. 1. Leading vehicle and the following vehicle (AV).

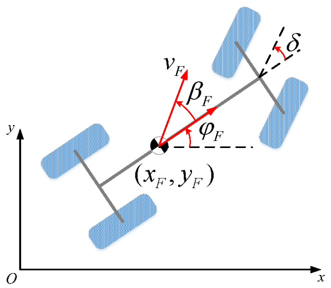


Fig. 2. Model description of the following vehicle. φ_F is the yaw angle and β_F is the slip angle. The traveling angle of the vehicle is $\varphi_F + \beta_F$.

is denoted as r, and we use $\kappa = 1/r$ to denote the curvature of the path. In Fig. 2, the variables x_w, y_w, and φ_w ($w \in \{L, F\}$) are the x coordinate, y coordinate, and yaw angle of the corresponding vehicles. Let β_w denote the slip angle of the corresponding vehicles. The variables v_w, ω_w, and $v_{\beta,w}$ are the velocity, yaw angular velocity, and slip angular velocity of the corresponding vehicles. We denote by a_w and $a_{\varphi,w}$ the accelerations along the directions of v_w and ω_w, respectively. For the leading vehicle L, its motion is characterized by the following differential equations:

$$\begin{aligned}\dot{x}_L &= v_L \cos(\varphi_L + \beta_L)\\ \dot{y}_L &= v_L \sin(\varphi_L + \beta_L)\\ \dot{\varphi}_L &= \omega_L\\ \dot{v}_L &= a_L\\ \dot{\omega}_L &= a_{\varphi,L}\\ \dot{\beta}_L &= v_{\beta,L}.\end{aligned} \tag{1}$$

To make the equation more compact, we define $p_L = [x_L, y_L, \varphi_L]^T$, $q_L = [v_L, \omega_L, \beta_L,]^T$ and $b_L = [a_L, a_{\varphi,L}, v_{\beta,L}]^T$. Define the state of the leading vehicle as $\chi_L = [p_L^T, q_L^T]^T$. Then, the motion of the leading vehicle is

$$\dot{\chi}_L = f_L(\chi_L) + g_L b_L \tag{2}$$

where $f_L(\cdot)$ is a function from $\mathbb{R}^6$ to $\mathbb{R}^6$, and $g_L \in \mathbb{R}^{6\times 3}$ is a constant matrix. They can be obtained from (1).

B. Model Description of the Following Vehicle

The kinematics of the following vehicle is:

$$\begin{aligned}\dot{x}_F &= v_F \cos(\varphi_F + \beta_F)\\ \dot{y}_F &= v_F \sin(\varphi_F + \beta_F)\\ \dot{\varphi}_F &= \omega_F.\end{aligned} \tag{3}$$

According to [48, Ch. 2] and [47], the bicycle model of the vehicle is

$$\dot{v}_F = u_1 \tag{4a}$$

$$\dot{\omega}_F = \lambda_1 \frac{\omega_F}{v_F} + \lambda_2 \beta_F + \lambda_3 u_2 \tag{4b}$$

$$\dot{\beta}_F = -\omega_F + \theta_1 \frac{\omega_F}{v_F^2} + \theta_2 \frac{\beta_F}{v_F} + \theta_3 \frac{1}{v_F} u_2 \tag{4c}$$

where λ_i and θ_i ($i = 1 \sim 3$) are determined by the physical parameters listed in Table I, and their detailed expressions are presented in Appendix A. Equation (4a) is for longitudinal motion; (4b) and (4c) are for lateral motion. The control inputs of the following vehicle are defined as:

$$u_1 = a_F, \quad u_2 = \delta. \tag{5}$$

The lateral dynamics in (4b) and (4c) can be expressed as the following compact form:

$$\begin{bmatrix}\dot{\omega}_F\\ \dot{\beta}_F\end{bmatrix} = f_l(v_F, \omega_F) + g_l(v_F)\begin{bmatrix}u_2\\ \beta_F\end{bmatrix} \tag{6}$$

where $f_l(v_F, \omega_F) : \mathbb{R}^2 \to \mathbb{R}^2$ and $g_l(v_F) : \mathbb{R} \to \mathbb{R}^{2\times 2}$ whose expressions are in Appendix A.

C. Problem Formulation

As shown in Fig. 1, the leading vehicle is moving along the path. By receiving the position and velocity of the leading vehicle via vehicle-to-vehicle (V2V) communication, the following vehicle aims to keep a safe distance from the leading vehicle and stay in the lane.

Most of the conventional approaches for the combined longitudinal and lateral control problem are model-based. That is the accurate dynamic model in (4) is required to design the controller. The performance of the model-based controller highly relies on the accuracy of the vehicle dynamics. In reality, the physical parameters vary for different AVs. The accurate dynamic model may not be attainable and as a result, the conventional model-based controller may not be practically applicable. Hence, motivated by the limitations of the model-based control approach, in this article, we investigate the following data-driven control problem.

Problem: In the absence of precise knowledge of the following vehicle's dynamics, learn directly from data a suboptimal controller that forces the vehicle F to maintain a safe distance from the vehicle L, while moving along the path marked by the red color in Fig. 1.

III. Tracking Error of the Following Vehicle

In Fig. 1, when the leading vehicle is moving at a constant speed, and the following vehicle accurately tracks the leading vehicle, we have $v_F = v_L$, $\omega_F = \omega_L$, $a_F = a_L = 0$, $a_{\varphi,F} = a_{\varphi,L} = 0$, and $\dot{\beta}_F = v_{\beta,L} = 0$. Under this condition, it is observed from (4) and Fig. 1 that the steering angle of the following vehicle u_2 and the slip angle β_F are nonzero when it moves along a circle. Consequently, a feedforward controller u_d is required to calculate the driving input when vehicle F accurately tracks vehicle L, and it is for eliminating steady-state tracking errors. Under this situation and from (4), u_d satisfies

$$u_{1,d} = 0 \tag{7a}$$

$$0 = f_l(v_L, \omega_L) + g_l(v_L)[u_{2,d}, \beta_{F,d}]^T \tag{7b}$$

or equivalently

$$u_{1,d} = 0 \tag{8a}$$

$$\left[u_{2,d}(v_L, \omega_L), \beta_{F,d}(v_L, \omega_L)\right]^T = -g_l^{-1}(v_L) f_l(v_L, \omega_L). \tag{8b}$$

It can be checked from (A.3) that the determinant of g_l is nonzero. Hence, g_l is invertible.

As seen in Fig. 1, the look-ahead point H is in the direction of the velocity v_F. A velocity-dependent intervehicle spacing policy is adopted as the safe headway

$$d = d_s + v_F t_s \tag{9}$$

where d_s is the standstill distance and t_s is the time headway. As studied by [49], string stability of the platoon is guaranteed by the velocity-dependent spacing policy.

If vehicle F follows vehicle L directly, it will cut the corner of the path [32]. To avoid the cutting-corner problem, a virtual point S is defined, which is in the direction of OL, and vehicle F follows the virtual point S instead of L. E is the desired position of vehicle F and ES $= d$ is tangent to the path. OES forms a right triangle, and L is on the line OS with LS $= s = (d^2 + r^2)^{1/2} - r$. To follow vehicle L and move along the path, vehicle F has to move to point E, and if this happens, S coincides with the look-ahead point H. By the geometry of the triangle OES, γ and its related trigonometric functions are

$$\begin{aligned}\gamma &= \arctan(\kappa d)\\ \sin(\gamma) &= \frac{\kappa d}{\sqrt{1+\kappa^2 d^2}}\\ \cos(\gamma) &= \frac{1}{\sqrt{1+\kappa^2 d^2}}.\end{aligned} \tag{10}$$

According to [32] and [33], the positions of the virtual point and look-ahead point in Fig. 1 are

$$\begin{aligned}S &= \begin{bmatrix} x_L \\ y_L \end{bmatrix} + s\begin{bmatrix} \sin(\varphi_L + \beta_L) \\ -\cos(\varphi_L + \beta_L) \end{bmatrix}\\ H &= \begin{bmatrix} x_F \\ y_F \end{bmatrix} + d\begin{bmatrix} \cos(\varphi_F + \beta_F) \\ \sin(\varphi_F + \beta_F) \end{bmatrix}.\end{aligned} \tag{11}$$

Since it is desired that the look-ahead point H coincides with the virtual point S, the error variables related to the deviation of H from S are

$$\begin{aligned}e_1 &= x_L + s\sin(\varphi_L + \beta_L) - x_F - d\cos(\varphi_F + \beta_F)\\ e_2 &= y_L - s\cos(\varphi_L + \beta_L) - y_F - d\sin(\varphi_F + \beta_F)\\ e_3 &= (\varphi_L + \beta_L - \gamma) - (\varphi_F + \beta_F)\\ e_4 &= v_L - v_F\\ e_5 &= \omega_L - \omega_F\\ e_6 &= \beta_{F,d} - \beta_F\end{aligned} \tag{12}$$

where e_1 and e_2 are the position tracking errors along x and y coordinate, respectively; e_3 is the traveling-angle tracking error that vanishes to zero when the traveling direction of vehicle F aligns with ES; e_4 is the velocity tracking error; e_5 is the angular velocity tracking error; and e_6 is the slip-angle tracking error. Furthermore, by transforming the position tracking errors (e_1 and e_2) from the world frame to vehicle F's body frame rotated by γ, we define z_1 and z_2 as

$$\begin{bmatrix} z_1 \\ z_2 \end{bmatrix} = \begin{bmatrix} \cos(\varphi_F + \beta_F + \gamma) & \sin(\varphi_F + \beta_F + \gamma) \\ -\sin(\varphi_F + \beta_F + \gamma) & \cos(\varphi_F + \beta_F + \gamma) \end{bmatrix}\begin{bmatrix} e_1 \\ e_2 \end{bmatrix}. \tag{13}$$

Subtracting (7) from (4) and substituting (12) into it, we have

$$\dot{v}_F = u_{1,e} \tag{14a}$$

$$\dot{\omega}_F = \lambda_1 \frac{\omega_L e_4 - v_L e_5}{v_L(v_L - e_4)} - \lambda_2 e_6 + \lambda_3 u_{2,e} \tag{14b}$$

$$\begin{aligned}\dot{\beta}_F = e_5 &+ \theta_1 \frac{-v_L^2 e_5 + 2 v_L \omega_L e_4 - e_4^2 \omega_L}{v_L^2 (v_L - e_4)^2}\\ &+ \theta_2 \frac{\beta_{F,d} e_4 - v_L e_6}{v_L(v_L - e_4)} + \frac{\theta_3 u_{2,d} e_4}{v_L(v_L - e_4)} + \frac{\theta_3}{v_L - e_4} u_{2,e}.\end{aligned} \tag{14c}$$

where

$$u_e = u - u_d. \tag{15}$$

Therefore, the control input $u = u_d + u_e$ has two parts: the feedforward controller u_d in (8), and the feedback controller u_e to be designed later. Define the error state of the system as $e = [z_1, z_2, e_3, e_4, e_5, e_6]^T$. Then, the error system is

$$\dot{e} = f_e(\chi_L, e) + g_e(\chi_L, e) u_e + h_e(\chi_L, e) b_L. \tag{16}$$

Here, $f_e(\cdot, \cdot)$ is a function from $\mathbb{R}^6 \times \mathbb{R}^6$ to $\mathbb{R}^6$, $g_e(\cdot, \cdot)$ is a function from $\mathbb{R}^6 \times \mathbb{R}^6$ to $\mathbb{R}^{6\times 2}$, and $h_e(\cdot, \cdot)$ is a function from $\mathbb{R}^6 \times \mathbb{R}^6$ to $\mathbb{R}^{6\times 3}$. The expressions of $f_e(\chi_L, e)$, $g_e(\chi_L, e)$ and $h_e(\chi_L, e)$ are in (B.6). The detailed derivation of the error system (16) is postponed to Appendix B.

Remark 1: The variables z_1 and z_2 in (13) are the tracking errors expressed in the vehicle F's body frame rotated by γ. By this way, the influence of e_3 on the evolution of z_1 and z_2 can be established [see (B.6)]. The transformation is widely used in vehicle control [50].

IV. Model-Based Policy Iteration for Feedback Controller Design

As seen in Fig. 3, the control framework of the following vehicle consists of the feedforward controller and feedback controller. The feedforward controller is designed in (8). In this section, a model-based policy iteration method is proposed to design an optimal feedback controller $u_e^*(t) = \alpha^*(\chi_L(t), e(t))$. When the leading vehicle reaches the equilibrium, it keeps

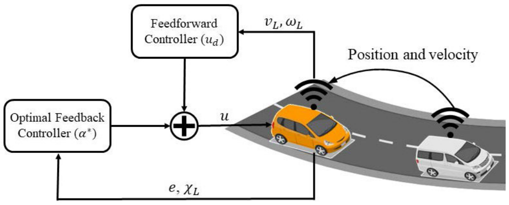

Feedforward Controller (u_d)
v_L, ω_L
Position and velocity
Optimal Feedback Controller (α^*)
u
e, χ_L

Fig. 3. Control framework of the following vehicle. Through V2V communication, the following vehicle receives the leading vehicle's position and velocity. The following vehicle's position and velocity are measured by on-board sensors. The input signal of the feedforward controller is the leading vehicle's velocity v_L and angular velocity ω_L. The input signals of the feedback controller are the tracking error [denoted as e and calculated by (12) and (13)] and leading vehicle's state (denoted as χ_L). The driving input u consists of the longitudinal acceleration (denoted as u_1) and steering angle (denoted as u_2).

the lane at a constant speed, i.e., $a_L = a_{\varphi,L} = v_{\beta,L} = 0$. Under this situation, to track the leading vehicle as accurately as possible with minimum energy consumption, the following optimal control problem is formulated to design the feedback controller:

$$\min_{u_e} J(\chi_L(0), e(0), u_e) = \int_0^\infty e^T Qe + u_e^T Ru_e \mathrm{d}t \tag{17a}$$

$$\text{s.t. } \dot{\chi}_L = f_L(\chi_L) \tag{17b}$$

$$\dot{e} = f_e(\chi_L, e) + g_e(\chi_L, e)u_e \tag{17c}$$

where Q and R are constant real symmetric and positive definite matrices with compatible dimensions. Equations (17b) and (17c) follow from (2) and (16) with $b_L = 0$. $\chi_L(0)$ and $e(0)$ are the initial states for the leading vehicle and the error system. Let $\mathbb{X} \subset \mathbb{R}^6$ denote the compact set containing all the possible initial states of the leading vehicle. The class of signals $\mathcal{I}(\mathbb{X})$ consists of all the trajectories of the leading vehicle, i.e., solutions $\chi_L(t) = \chi_L(t, \chi_L(0))$ of (17b) starting at $\chi_L(0) \in \mathbb{X}$.

Following [45, Ch. 5], the solution of the optimal control problem (17) depends on the solvability of the following HJB equation:

$$\frac{\partial V^{*T}}{\partial \chi_L} f_L + \frac{\partial V^{*T}}{\partial e} f_e - \frac{1}{4}\left(\frac{\partial V^{*T}}{\partial e} g_e\right) R^{-1} \left(\frac{\partial V^{*T}}{\partial e} g_e\right)^T + e^T Qe = 0 \tag{18}$$

where $V^*(\chi_L, e)$ is the optimal value function. The optimal feedback controller minimizing (17a) is obtained by

$$\alpha^*(\chi_L, e) = -\frac{1}{2} R^{-1} g_e(\chi_L, e)^T \frac{\partial V^*(\chi_L, e)}{\partial e}. \tag{19}$$

It is seen from (19) that the optimal controller is obtained once the HJB equation (18) is solved. However, (18) is a nonlinear partial differential equation, which is hard to solve directly. The requirement of the precise dynamic model (f_e, g_e) for solving (18) makes it worse. Next, we will propose a model-based policy iteration algorithm to iteratively approximate the solution of (18). Before the statement of the model-based policy iteration, the following definition is introduced.

Definition 1: A controller $\alpha(\chi_L, e) : \mathbb{X} \times \mathbb{R}^6 \to \mathbb{R}^3$ is admissible with respect to the cost J in (17a) if the following properties hold.

1) α is continuous on $\mathbb{X} \times \mathbb{R}^6$.

2) With $u_e(t) = \alpha(\chi_L(t), e(t))$ and for any $\chi_L \in \mathcal{I}(\mathbb{X})$, the error system (17c) is globally asymptotically stable at the origin.

3) $J(\chi_L(0), e(0), u_e) < \infty$ for all $(\chi_L(0), e(0)) \in \mathbb{X} \times \mathbb{R}^6$.

Algorithm 1 Model-Based Policy Iteration

```
Let $\alpha_1(\chi_L, e)$ be any admissible controller. Choose a sufficiently small threshold $\zeta > 0$. $i \leftarrow 1$.
repeat
    Policy evaluation: solve the following partial differential equation to obtain $V_i(\chi_L, e)$
        $\frac{\partial V_i^T}{\partial \chi_L} f_L + \frac{\partial V_i^T}{\partial e}(f_e + g_e \alpha_i) + e^T Qe + \alpha_i^T R \alpha_i = 0.$   (20)
    Policy improvement: update the control law to $\alpha_{i+1}(\chi_L, e)$ by
        $\alpha_{i+1}(\chi_L, e) = -\frac{1}{2} R^{-1} g_e^T \frac{\partial V_i}{\partial e}.$   (21)
    $i \leftarrow i + 1$
until $\|\alpha_{i+1} - \alpha_i\|_\infty \leq \zeta$
```

Generally speaking, in Fig. 1, with an admissible controller, the following vehicle will asymptotically move to the desired position E. Policy iteration starts with some (non-optimal) admissible policy and then calculates the value of each state given that policy, which is called policy evaluation. The policy itself is then updated to minimize the cost. The detailed steps of policy iteration are in Algorithm 1.

When $u_{e,i}(\chi_L, e)$ and $V_i(\chi_L, e)$ are updated by Algorithm 1, the following lemma guarantees that the updated controller is admissible, provides an improvement, and converges to the optimal one.

Lemma 1 [43]: When $\alpha_1(\chi_L, e)$ is admissible, and $V_i(\chi_L, e)$ and $\alpha_i(\chi_L, e)$ are obtained by solving (20) and (21), the following properties hold.

1) $V^*(\chi_L, e) \leq V_{i+1}(\chi_L, e) \leq V_i(\chi_L, e)\ \forall(\chi_L, e) \in \mathbb{X} \times \mathbb{R}^n$.

2) $\alpha_i(\chi_L, e)$ is admissible for all $i \geq 1$.

3) $\lim_{i\to\infty} V_i(\chi_L, e) = V^*(\chi_L, e)$ and $\lim_{i\to\infty} \alpha_i(\chi_L, e) = \alpha^*(\chi_L, e)$.

It is seen from the first statement of Lemma 1 that the performance of the updated controller $\alpha_i(\chi_L, e)$ is continuously refined at each iteration. In addition, the controller obtained at each iteration stabilizes and converges the optimal solution as the iteration tends to infinity. These properties are critical for the stability and performance of the AV. Nonetheless, the accurate dynamic model (f_e, g_e) is still required to implement (20) and (21). In Section V, a data-driven policy iteration algorithm is developed in the absence of accurate knowledge of the dynamic model (f_e, g_e).

V. Two-Phase Data-Driven Policy Iteration

It follows from (15) that the controller $u = u_d + u_e$ for the following vehicle has two parts: feedforward controller u_d designed by (8) and feedback controller $\alpha^*(\chi_L, e)$ designed by Algorithm 1. However, the feedforward controller design in (8) relies on the model of the following vehicle

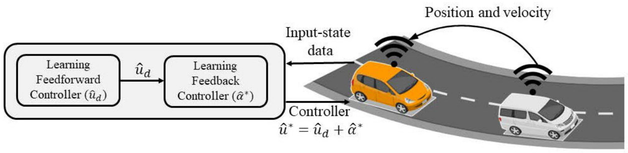

Position and velocity
Input-state data
Learning Feedforward Controller ($\hat{u}_d$)
$\hat{u}_d$
Learning Feedback Controller ($\hat{\alpha}^*$)
Controller
$\hat{u}^* = \hat{u}_d + \hat{\alpha}^*$

Fig. 4. Two-phase learning framework. Input-state data include the leading vehicle's position and velocity, and the following vehicle's position, velocity, and driving input. "Hat" distinguishes the learning-based near-optimal controller from the optimal controller.

(f_l and g_l). The model-based policy iteration in Algorithm 1 also requires the system model (f_e and g_e). In reality, such an accurate model is hard to obtain due to the parameter variations (e.g., the following vehicle drives on different road conditions). To overcome this limitation, in this section, using input-state data collected from the following vehicle and the leading vehicle, we will design a two-phase data-driven controller design method in the absence of accurate knowledge of the system model. The first phase is to design the feedforward controller $\hat{u}_d$ and the second phase is to iteratively approximate the optimal feedback controller $\alpha^*(\chi_L, e)$. The learning framework is shown in Fig. 4.

A. Phase One: Design the Feedforward Controller Using Input-State Data

To get the feedforward controller in (8), the key is to find the approximations of f_l and g_l. In (6), $f_l(\cdot) : \mathbb{R}^2 \to \mathbb{R}^2$ and $g_l : \mathbb{R}^2 \to \mathbb{R}^{2\times 2}$ are nonlinear functions. Let $t_1 < \cdots < t_k < t_{k+1} < \cdots < t_{K+1}$ denote the boundaries of the sampling intervals. Integrating (6) from t_k to t_{k+1}, we have

$$\begin{aligned}&[\omega_F(t), \beta_F(t)]^T\Big|_{t=t_k}^{t_{k+1}}\\ &\quad= \int_{t_k}^{t_{k+1}} f_l(v_F(t), \omega_F(t))\mathrm{d}t\\ &\qquad+ \int_{t_k}^{t_{k+1}} [u_2(t), \beta_F(t)] \otimes I_2 \operatorname{vec}(g_l(v_F(t)))\mathrm{d}t. \end{aligned} \tag{22}$$

According to the approximation theory [51], over any compact set $[v_F, \omega_F]^T \in \mathcal{V} \subset \mathbb{R}^2$, f_l and g_l can be expressed as

$$\begin{aligned} f_l(v_F, \omega_F) &= \sigma_f^T \Phi_f(v_F, \omega_F) + e_f(v_F, \omega_F)\\ \operatorname{vec}(g_l(v_F)) &= \sigma_g^T \Phi_g(v_F) + e_g(v_F) \end{aligned} \tag{23}$$

where $\sigma_f \in \mathbb{R}^{2\times N_f}$ and $\sigma_g \in \mathbb{R}^{4\times N_g}$ are the weighting matrices; $\Phi_f(v_F, \omega_F)$ and $\Phi_f(v_F)$ are N_f-dimensional and N_g-dimensional vectors of linearly independent basis functions; $e_f(v_F, \omega_F)$ and $e_g(v_F)$ are the truncation errors, which uniformly converge to zero over $\mathcal{V}$ as N_f and N_g tend to infinity. Plugging (23) into (22) yields

$$\begin{aligned}&[\omega_F(t), \beta_F(t)]^T\Big|_{t=t_k}^{t_{k+1}}\\ &= \int_{t_k}^{t_{k+1}} \Phi_f(v_F(t), \omega_F(t))^T \otimes I_2\mathrm{d}t \operatorname{vec}(\sigma_f^T)\\ &\quad+ \int_{t_k}^{t_{k+1}} \Phi_g(v_F(t))^T \otimes [u_2(t), \beta_F(t)] \otimes I_2\mathrm{d}t \operatorname{vec}(\sigma_g^T)\\ &\quad+ \int_{t_k}^{t_{k+1}} e_f(v_F(t), \omega_F(t)) + [u_2(t), \beta_F(t)] \otimes I_2 e_g(v_F(t))\mathrm{d}t. \end{aligned} \tag{24}$$

Stacking (24) for $k = 1, \ldots, K$ yields

$$\Lambda\Big[\operatorname{vec}\big(\sigma_f^T\big)^T, \operatorname{vec}\big(\sigma_g^T\big)^T\Big]^T + \varepsilon = \Xi \tag{25}$$

where

$$\Lambda = \begin{bmatrix} \int_{t_1}^{t_2} \Phi_f^T \otimes I_2\mathrm{d}t & \int_{t_1}^{t_2} \Phi_g^T \otimes [u_2(t), \beta_F(t)] \otimes I_2\mathrm{d}t\\ \cdots & \cdots\\ \int_{t_K}^{t_{K+1}} \Phi_f^T \otimes I_2\mathrm{d}t & \int_{t_K}^{t_{K+1}} \Phi_g^T \otimes [u_2(t), \beta_F(t)] \otimes I_2\mathrm{d}t \end{bmatrix} \tag{26a}$$

$$\Xi = \big[[\omega_F(t), \beta_F(t)]_{t=t_1}^{t_2}, \ldots, [\omega_F(t), \beta_F(t)]_{t_K}^{t_{K+1}}\big]^T \tag{26b}$$

$$\begin{aligned}\varepsilon = \Big[&\int_{t_1}^{t_2} \big(e_f + [u_2(t), \beta_F(t)] \otimes I_2 e_g\big)^T \mathrm{d}t, \ldots,\\ &\int_{t_K}^{t_{K+1}} \big(e_f + [u_2(t), \beta_F(t)] \otimes I_2 e_g\big)^T \mathrm{d}t\Big]^T. \end{aligned} \tag{26c}$$

Assumption 1: There exist $K > 0$ and $\eta > 0$, such that the following inequality holds:

$$\frac{1}{K}\Lambda^T\Lambda \geq \eta I_{2N_f+4N_g}. \tag{27}$$

Remark 2: Assumptions 1 and 2 are similar to the persistent excitation (PE) condition in adaptive control (see [52]), and can be checked by the smallest singular value of $\Lambda^T\Lambda$. To satisfy the PE-like condition, one can add exploration signals to the behavior policy, for example, sinusoidal signals and random signals.

With Assumption 1, $\hat{\sigma}_f$ and $\hat{\sigma}_g$, the estimations of σ_f and σ_g, can be obtained as the least-squares solution to (25), i.e.,

$$\Big[\operatorname{vec}\big(\hat{\sigma}_f^T\big)^T, \operatorname{vec}\big(\hat{\sigma}_g^T\big)^T\Big]^T = (\Lambda^T\Lambda)^{-1}\Lambda^T\Xi. \tag{28}$$

Hence, the functions $f_l(v_F, \omega_F)$ and $g_l(v_F)$ over the compact set $\mathcal{V}$ can be approximated by

$$\begin{aligned} \hat{f}_l(v_F, \omega_F) &= \hat{\sigma}_f^T \Phi_f(v_F, \omega_F)\\ \operatorname{vec}(\hat{g}_l(v_F)) &= \hat{\sigma}_g^T \Phi_g(v_F). \end{aligned} \tag{29}$$

The following proposition ensures that the approximation errors $(f_l(v_F, \omega_F) - \hat{f}_l(v_F, \omega_F))$ and $(g_l(v_F) - \hat{g}_l(v_F))$ uniformly converge to zero as $N_f, N_g \to \infty$.

Proposition 1: Under Assumption 1, when $\hat{\sigma}_f$ and $\hat{\sigma}_g$ are calculated by (28), the following limits hold over the compact set $[v_F, \omega_F]^T \in \mathcal{V}$:

$$\begin{aligned} \lim_{N_f, N_g\to\infty} \big\| f_l(v_F, \omega_F) - \hat{f}_l(v_F, \omega_F)\big\|_\infty &= 0\\ \lim_{N_f, N_g\to\infty} \big\| g_l(v_F) - \hat{g}_l(v_F)\big\|_\infty &= 0. \end{aligned} \tag{30}$$

Proof: See Appendix C. □

Therefore, with the obtained estimations $\hat{f}_l$ and $\hat{g}_l$, according to (8), the feedforward controller can be approximated as

$$u_{1,d} = 0 \tag{31a}$$

$$[\hat{u}_{2,d}(v_L, \omega_L), \hat{\beta}_{F,d}(v_L, \omega_L)]^T = -\hat{g}_l^{-1}(v_L)\hat{f}_l(v_L, \omega_L). \tag{31b}$$

Proposition 2: Under Assumption 1, when $\hat{\sigma}_F$ and $\hat{\sigma}_g$ are calculated by (28) and $[\hat{u}_{2,d}(v_L, \omega_L), \hat{\beta}_{F,d}(v_L, \omega_L)]^T$

is calculated by (31), the following limits hold over $[v_L, \omega_L]^T \in \mathcal{V}$:

$$\lim_{N_f, N_g \to \infty} \left\| \hat{u}_{2,d}(v_L, \omega_L) - u_{2,d}(v_L, \omega_L) \right\|_\infty = 0$$
$$\lim_{N_f, N_g \to \infty} \left\| \hat{\beta}_{F,d}(v_L, \omega_L) - \beta_{F,d}(v_L, \omega_L) \right\|_\infty = 0. \quad (32)$$

Proof: Since $u_{2,d}(v_L, \omega_L)$ and $\beta_{F,d}(v_L, \omega_L)$ in (7) are analytic in f_l and g_l, and g_l is invertible, by Proposition 1, Proposition 2 follows readily. □

B. Phase Two: Design the Optimal Feedback Controller Using Input-State Data

Based on the model-based policy iteration in Algorithm 1, we will develop a data-driven approach to learn iteratively the optimal controller from the motion data from both the following and leading vehicles. Let $\nu_i(t) = u_e(t) - \alpha_i(\chi_L(t), e(t))$ denote the difference between the exploratory/behavior policy and the policy $\alpha_i(\chi_L, e)$ obtained at the ith iteration of Algorithm 1. Then, (17c) can be rewritten as

$$\dot{e} = f_e(\chi_L, e) + g_e(\chi_L, e)\alpha_i + g_e(\chi_L, e)\nu_i. \quad (33)$$

Along the trajectories of (33), the derivative of $V_i(\chi_L, e)$ can be expressed as

$$\dot{V}_i = \frac{\partial V_i^T}{\partial \chi_L} f_L + \frac{\partial V_i^T}{\partial e}(f_e + g_e \alpha_i + g_e \nu_i). \quad (34)$$

Recall that $V_i(\chi_L, e)$ is the value function at the ith iteration of Algorithm 1. Substituting (20) and (21) into (34), we obtain

$$\dot{V}_i = -e^T Q e - \alpha_i^T R \alpha_i - 2\alpha_{i+1}^T R \nu_i. \quad (35)$$

Integrating both sides of (35) from t_p to t_{p+1}, we have

$$V_i(t_{p+1}) - V_i(t_p) + 2\int_{t_p}^{t_{p+1}} \alpha_{i+1}^T R \hat{\nu}_i \mathrm{d}t$$
$$= \int_{t_p}^{t_{p+1}} -e^T Q e - \hat{\alpha}_i^T R \hat{\alpha}_i \mathrm{d}t + \Delta_{i,p} \quad (36)$$

where $t_1 < \cdots < t_p < t_{p+1} < \cdots < t_{P+1}$ are the boundaries of the sampling intervals. Notice that $V_i(t) = V_i(\chi_L(t), e(t))$. In addition, $\hat{\nu}_i = u - \hat{u}_d - \hat{\alpha}_i$, and

$$\Delta_{i,p} = \int_{t_p}^{t_{p+1}} 2\alpha_{i+1}^T R(u_d - \hat{u}_d + \alpha_i - \hat{\alpha}_i)$$
$$+ (\hat{\alpha}_i - \alpha_i)^T R(\alpha_i + \hat{\alpha}_i)\mathrm{d}t \quad (37)$$

where $\hat{\alpha}_i$ is the approximation of α_i. By the approximation theory (see [51]), V_i and α_i can be approximated by the linear combinations of a set of linearly independent basis functions, i.e.,

$$V_i(\chi_L, e) = \rho_i^T \Phi_V(\chi_L, e) + e_{V,i}(\chi_L, e)$$
$$\alpha_i(\chi_L, e) = \mu_i^T \Phi_\alpha(\chi_L, e) + e_{\alpha,i}(\chi_L, e) \quad (38)$$

where $\Phi_V(\chi_L, e)$ is an N_V-dimensional vector of linearly independent basis functions; $\Phi_\alpha(\chi_L, e)$ is an N_α-dimensional vector of linearly independent basis functions; $\rho_i \in \mathbb{R}^{N_V}$ and $\mu_i \in \mathbb{R}^{N_\alpha \times 2}$ are weighting matrices; $e_{V,i}(\chi_L, e) \in \mathbb{R}$ and $e_{\alpha,i}(\chi_L, e) \in \mathbb{R}^2$ are approximation truncation errors that converge to zero uniformly over any compact set $(\chi_L, e) \in \mathcal{S} \subset \mathbb{R}^6 \times \mathbb{R}^6$ as $N_V, N_\alpha \to \infty$. Plugging (38) into (36), we have

$$\rho_i^T \left(\Phi_V(t_{p+1}) - \Phi_V(t_p)\right) + 2\int_{t_p}^{t_{p+1}} \Phi_\alpha^T \mu_{i+1} R \hat{\nu}_i \mathrm{d}t$$
$$= \int_{t_p}^{t_{p+1}} -e^T Q e - \hat{\alpha}_i^T R \hat{\alpha}_i \mathrm{d}t + \Delta_{i,p} + \xi_{i,p} \quad (39)$$

where

$$\xi_{i,p} = e_{V,i}(t_p) - e_{V,i}(t_{p+1}) - 2\int_{t_p}^{t_{p+1}} e_{\alpha,i+1}^T R \hat{\nu}_i \mathrm{d}t. \quad (40)$$

Let $\Delta_i = [\Delta_{i,1}, \ldots, \Delta_{i,p}, \ldots, \Delta_{i,P}]^T$ and $\xi_i = [\xi_{i,1}, \ldots, \xi_{i,p}, \ldots, \xi_{i,P}]^T$. Then, stacking the equations (39) for $p = 1, \ldots, P$, we have

$$\Theta_i \begin{bmatrix} \rho_i \\ \mathrm{vec}(\mu_{i+1}) \end{bmatrix} = \Gamma_i + \Delta_i + \xi_i \quad (41)$$

where

$$\Theta_i = \begin{bmatrix} (\Phi_V(t_2) - \Phi_V(t_1))^T & 2\int_{t_1}^{t_2} (\hat{\nu}_i^T R) \otimes \Phi_\alpha^T \mathrm{d}t \\ \cdots & \cdots \\ (\Phi_V(t_{P+1}) - \Phi_V(t_P))^T & 2\int_{t_P}^{t_{P+1}} (\hat{\nu}_i^T R) \otimes \Phi_\alpha^T \mathrm{d}t \end{bmatrix} \quad (42a)$$

$$\Gamma_i = \left[-\int_{t_1}^{t_2} e^T Q e + \hat{\alpha}_i^T R \hat{\alpha}_i \mathrm{d}t, \ldots,\right.$$
$$\left. -\int_{t_P}^{t_{P+1}} e^T Q e + \hat{\alpha}_i^T R \hat{\alpha}_i \mathrm{d}t\right]^T. \quad (42b)$$

In (41), the property of Kronecker product that for any matrices X, Y, Z with compatible dimensions, $\mathrm{vec}(XYZ) = (Z^T \otimes X)\,\mathrm{vec}(Y)$, is used.

Assumption 2: There exist $P > 0$, and $\eta > 0$, such that for any $i \geq 1$, the following inequality holds:

$$\frac{1}{P}\Theta_i^T \Theta_i \geq \eta I_{N_V + 2N_\alpha}. \quad (43)$$

Under Assumption 2, by the least-squares method and (41), the approximations of ρ_i and μ_{i+1} are

$$\begin{bmatrix} \hat{\rho}_i \\ \mathrm{vec}(\hat{\mu}_{i+1}) \end{bmatrix} = (\Theta_i^T \Theta_i)^{-1} \Theta_i^T \Gamma_i. \quad (44)$$

With the approximated weighting matrices $\hat{\rho}_i$ and $\mathrm{vec}(\hat{\mu}_{i+1})$, the value function and the updated controller are

$$\hat{V}_i(\chi_L, e) = \hat{\rho}_i^T \Phi_V(\chi_L, e) \quad (45a)$$
$$\hat{\alpha}_{i+1}(\chi_L, e) = \hat{\mu}_{i+1}^T \Phi_\alpha(\chi_L, e). \quad (45b)$$

The following proposition ensures that at each iteration, the approximated value function and feedback controller are accurate enough as long as the number of basis functions is large enough.

Proposition 3: For each step of data-driven policy iteration, the following properties hold:

$$\lim_{N_f, N_g, N_V, N_\alpha \to \infty} \hat{V}_i(\chi_L, e) = V_i(\chi_L, e)$$
$$\lim_{N_f, N_g, N_V, N_\alpha \to \infty} \hat{\alpha}_{i+1}(\chi_L, e) = \alpha_{i+1}(\chi_L, e) \quad (46)$$

where V_i and α_{i+1} are the solutions to (20) and (21).

Algorithm 2 Two-Phase Data-Driven Policy Iteration

```
Phase-one: learning the feedforward controller
Set the sampling intervals $t_1, \cdots, t_k, t_{k+1}, \cdots, t_{K+1}$.
Choose driving inputs $u_{exp,1}(t)$ to explore system (6) and collect input-state data.
Construct the matrices $\Lambda$ and $\Xi$ by (26a) and (26b), respectively.
Solve (28) and get $\hat{f}_l$ and $\hat{g}_l$ by (29).
Get the expressions of $\hat{u}_d$ and $\hat{\beta}_{F,d}$ by (31).
Phase-two: learning the optimal feedback controller
Choose an initial admissible controller $\alpha_1(\chi_L, \omega_L)$.
Set the sampling intervals $t_1, \cdots, t_p, t_{p+1}, \cdots, t_{P+1}$.
Set the small threshold $\zeta > 0$.
Set the exploration signals $\psi(t)$.
Using $u_{exp,2}(t) = \hat{u}_d(v_L(t), \omega_L(t)) + \alpha_1(\chi_L(t), e(t)) + \psi(t)$ to explore the system and collect input-state data.
repeat
    Construct the matrices $\Theta_i$ and $\Gamma_i$ by (42a) and (42b), respectively.
    Solve (44) and get the updated controller $\hat{\alpha}_{i+1}$ by (45b).
    $i \leftarrow i + 1$
until $|\hat{\mu}_{i+1} - \hat{\mu}_i| \leq \zeta$
Update controller $\hat{u}_i(\chi_L, e) = \hat{\alpha}_i(\chi_L, e) + \hat{u}_d(v_L, \omega_L)$.
```

TABLE I

MODEL PARAMETERS

Parameters	Description	Value
C_f $[N/rad]$	Front cornering stiffness	20.90
C_r $[N/rad]$	Rear cornering stiffness	20.90
M $[kg]$	Mass of the vehicle	1226
I_z $[kg \cdot m^2]$	Yaw inertia of the vehicle	1539
l_f $[m]$	Length from front axle to CoM	0.88
l_r $[m]$	Length from back axle to CoM	1.51
g_a $[m/sec^2]$	Gravitational acceleration	9.81
c	Friction coefficient	1.04
t_s $[sec]$	Time headway	1
d_s $[m]$	Standstill distance	5

Proof: See Appendix D. □

By Lemma 1 and Proposition 3, the optimal feedback control policy $\alpha^*(\chi_L, e)$ and the value function $V^*(\chi_L, e)$ can be approximated as close as possible by increasing the number of linearly independent basis functions and the iterations. The detailed data-driven policy iteration is shown in Algorithm 2.

VI. SUMO Testing

In this section, we use the integrated simulation of SUMO [46] and CommonRoad [47] to test the efficiency of the two-phase data-driven policy iteration approach for the combined longitudinal and lateral control. The single-track model for vehicle 1 in CommonRoad is adopted as the dynamic model of the following vehicle. The parameters of the following vehicle and the intervehicle spacing policy are shown in Table I. The intervehicle spacing policy is adopted from [16].

A. Training Process

The initial admissible controller α_1 for the policy iteration algorithm is designed by the feedback linearization technique [53] with an approximate value of the dynamical parameters of the vehicle. For the phase-one learning, the exploration driving input $u_{\text{exp},1}$ is set as the summation of sinusoidal signals with different frequencies, i.e.,

$$[u_{\text{exp},1}(t)]_j = c_j \cdot \sum_{l=1}^{500} \sin(w_{j,l} t), \quad j = 1, 2 \tag{47}$$

where $c_1 = 0.1$, $c_2 = 0.001$, $[u_{\text{exp},1}]_j$ denotes the jth element of the vector $u_{\text{exp},1}(t)$, and $w_{j,l}$ is independently and randomly sampled from the uniform distribution over $[-250, 250]$. Then, with the learned feedforward controller $\hat{u}_d(v_L, \omega_L)$, the exploratory policy for the phase-two learning is set as

$$\begin{aligned}[u_{\text{exp},2}(t)]_j &= [\alpha_1(\chi_L(t), e(t)) + \hat{u}_d(v_L(t))]_j \\ &\quad + m_j \cdot \sum_{l=1}^{500} \sin\left(w'_{j,l} t\right), \quad j = 1, 2, 3 \end{aligned} \tag{48}$$

where $m_1 = 0.5$, $m_2 = 0.02$, and $w'_{j,l}$ is randomly sampled from the uniform distribution over $[-250, 250]$.

The performance index for the optimal feedback control design is $J = \int_0^\infty e^T Q e + u_e^T R u_e \mathrm{d}t$, where $Q = 10 \times I_6$ and $R = I_2$. When collecting data, the leading vehicle is moving around a circle with the radius of $r = 100$ m at a constant speed of $v_L = 20$ m/sec. The integration interval is $t_{k+1} - t_k = t_{p+1} - t_p = 0.01$ s. The vector of basis functions for $\hat{f}_l$ is $\Phi_f(v_F, \omega_F) = [\omega_F / v_F, \omega_F, \omega_F / v_F^2]^T$. The vector of basis functions for $\hat{g}_l$ is $\Phi_g(v_F) = [1, 1/v_F]^T$. Define $\text{Poly}_m(x_1, \ldots, x_n)$ as the vector consisting of all the polynomial terms of $(x_1 + \cdots + x_n)^m$. $\text{CS}_1 = [1 - \cos e_3, \sin e_3]$ and $e_{456} = [e_4, e_5, e_6]$. The vector of basis functions for $\hat{V}_i(\chi_L, e)$ is

$$\begin{aligned}\Phi_V(\chi_L, e) = \big[&\text{Poly}_3(e), \text{Poly}_2(e), \text{Poly}_2(e) \otimes \text{CS}_1, \\ &\text{Poly}_1(e), \text{Poly}_1(e) \otimes \text{CS}_1, e_{456}/(v_L - e_4), 1\big]^T.\end{aligned}$$

The vector of basis functions for $\hat{\alpha}_i(\chi_L, e)$ is

$$\begin{aligned}&\Phi_\alpha(\chi_L, e) \\ &= \big[\text{Poly}_2(e), \text{Poly}_1(e), \text{Poly}_1(e) \otimes \text{CS}_1, e_{456}/(v_L - e_4), 1\big]^T.\end{aligned}$$

The selection of the basis functions is inspired by the Weierstrass approximation theorem [54], that is any continuous function over a compact set can be uniformly approximated by polynomials. We use third-order polynomials of error states e to approximate the value functions, and second-order polynomials of error states e to approximate the control policies. The additional items, e.g., $[e_4/(v_L - e_4), e_5/(v_L - e_4), e_6/(v_L - e_4)]$ and $\text{Poly}_1(\text{e}) \otimes [1 - \cos(e_3), \sin(e_3)]$, are inspired by the expression of the error dynamics in (B.6).

The collected input-state data includes p_L, q_L, p_F, q_F, and u. The input-state data collected from 0 to 0.5 s is used for phase-one learning and the data collected from 0.5 to 10.5 s are used for phase-two learning. The trajectories of the leading vehicle and the following vehicle during the data-collection

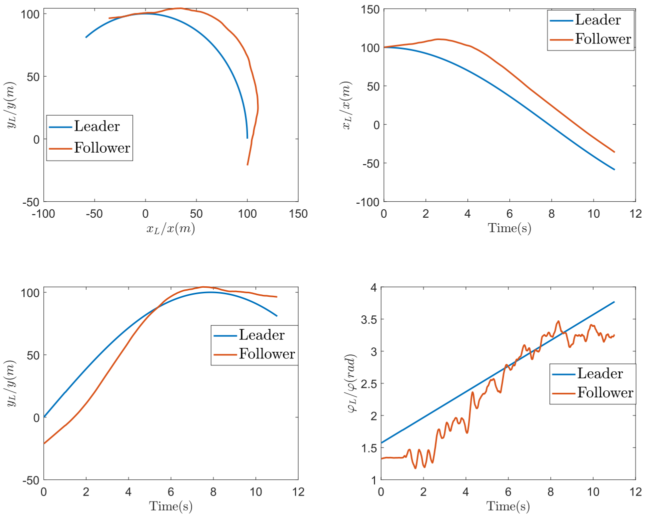
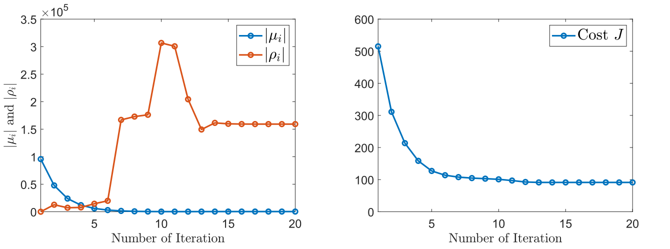


Fig. 5. Trajectory of the vehicles during data collection.



(a) (b)

Fig. 6. Evolution of the weights ($|\mu_i|$ and $|\rho_i|$) and cost J during learning process. (a) Evolution of $|\mu_i|$ and $|\rho_i|$. (b) Evolution of cost J.

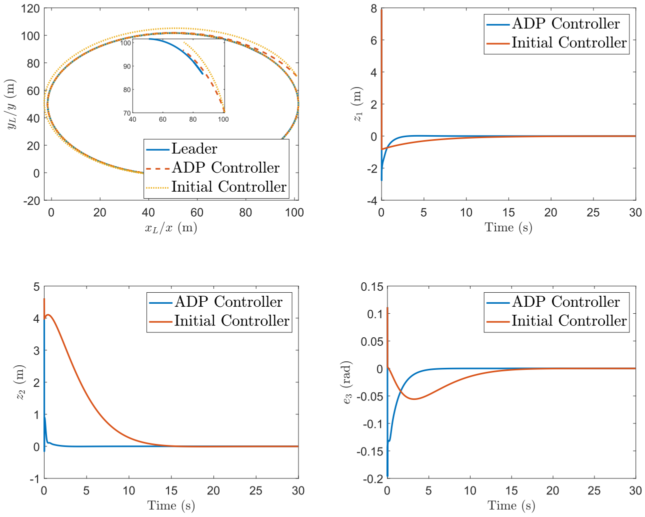


Fig. 7. Trajectories and tracking errors of the following vehicle when the leading vehicle is moving along a circular road. The initial controller is designed by feedback linearization to start Algorithm 2, and the ADP controller is learned by Algorithm 2.

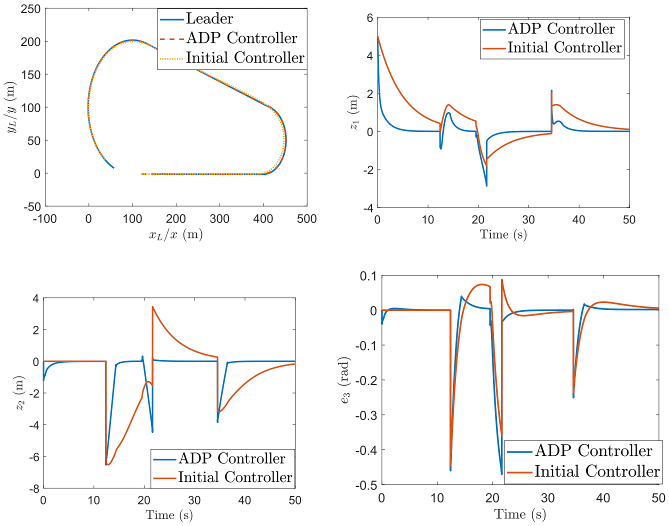


Fig. 8. Trajectories and tracking errors of the following vehicle when the leading vehicle is moving along a mixed road. The initial controller is designed by feedback linearization to start Algorithm 2, and the ADP controller is learned by Algorithm 2.

phase are shown in Fig. 5. The chattering phenomena during data collection in Fig. 5 is caused by the added exploratory signals in (49) and (50), which are designed to guarantee the PE condition in Assumptions 1 and 2. The evolution of the weights of the control policy and the value function are shown in Fig. 6(a). After the 20th iteration of Algorithm 2, the weights reach the stopping threshold of the algorithm $|\hat{\mu}_{i+1} - \hat{\mu}_i| < 10^{-3}$. In the next sections, the learned ADP controller $u_{20} = \hat{u}_d + \hat{u}_{e,20}$ will be applied under different traffic scenarios to test its performance. Starting from the same initial state, the cost for the controller $\hat{u}_i$ learned at the ith iteration is plotted in Fig. 6(b). The monotonically decreasing cost demonstrates that the controller is iteratively refined by the proposed policy iteration algorithm, which aligns with the theoretical results in Lemma 1. Compared with the traditional deep reinforcement learning algorithms which require a large volume of data to train the deep neural network, the proposed algorithm only requires 10.5 s motion data to learn the control policy. Therefore, the proposed data-driven control method is more data efficient.

B. Performance Evaluation for Roads of Different Shapes

The learned ADP controller is first evaluated on a circle road and compared with the initial controller which starts Algorithm 2. The center of the circle is (50, 50) m and the radius is 51.6 m. The leading vehicle is moving at a constant speed of $v_L = 20$ m/s. The trajectories of the vehicles and the evolution of the error states are plotted in Fig. 7. When the simulation starts, the following vehicle is located in a different lane from the leading vehicle. After around 5 s, with the ADP controller, the following vehicle moves along the same lane with the leading vehicles, and all the errors converge to zero (see Fig. 7). Compared with the initial controller in Fig. 7 from $t = 0$ s to $t = 10$ s, the tracking errors of the following vehicle with the ADP controller converge to zero faster. The value of the performance index of the ADP controller is $J = 43.6067$, while that of the initial controller is $J = 622.6262$. Hence, the performance is improved by Algorithm 2 with respect to the designed quadratic performance index.

Second, the ADP controller is evaluated on a mixed road consisting of straight lines and circles as shown in Fig. 8. The radius of the larger circle is 101.6 m, and that of the smaller circle is 51.6 m. Two circles are linked by tangent lines, and the length of the bottom line is 300 m. The leading vehicle is

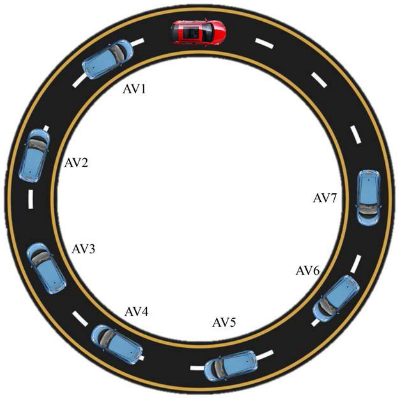
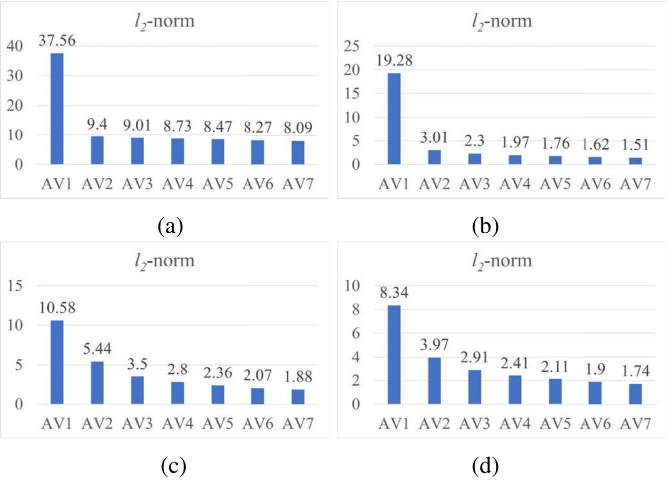


Fig. 9. Platoon consists of seven AVs and a leading vehicle on a circle road. The red one is the leading vehicle moving at a constant speed of $v_L = 20$ m/s. The radius of the circle is 100 m.

moving at 20 m/s along the road. The leading vehicle starts at the left end of the bottom line and moves anticlockwise. The trajectories and the tracking errors are plotted in Fig. 8. When moving on the bottom line, with the ADP controller, the following vehicle quickly catches up with the leading vehicle and keeps the lane accurately. At around 10 s, the leading vehicle moves onto the smaller circle. At around 20 s, the leading vehicle moves from the smaller circle to the straight section, and at around 35 s, it moves from the straight section to the larger circle. Due to the sudden changes in the curvature of the road in these instances, the tracking errors of the following vehicle suddenly change to non-zero values. Then, these tracking errors quickly converge to zero with the help of the ADP controller. As a comparison, when driven by the initial controller, the tracking errors converge slowly, and cannot stay at zero due to the frequent changes in road geometry. In conclusion, the proposed two-phase data-driven algorithm can generate a near-optimal controller for the combined longitudinal and lateral control task. The learned controller not only guarantees stability, but also improves the performance of the following vehicle with respect to the designed performance index.

C. Analysis of String Stability on a Circle Road

In Section VI-B, the ADP controller is evaluated under simple scenarios where only one AV is following the leading vehicle, and the stability of the following vehicle is guaranteed by Lemma 1 and Proposition 3. When multiple AVs form a platoon, string stability is critical for the attenuation of disturbances. In particular, string stability ensures that the magnitude of disturbances decreases when they propagates along the vehicles within the platoon. In this section, we will numerically analyze the string stability of the homogeneous CAV platoon in Fig. 9. Especially, the influence of the weighting matrices of the performance index in (17) on the ℓ_2 string stability will be studied. All the AVs in Fig. 9 have the same physical parameters as in Table I and adopt the same controller generated by Algorithm 2. Following [4], the definition of ℓ_2 string stability for 2-D combined longitudinal and lateral control is as follows.



(a) (b)

(c) (d)

Fig. 10. ℓ_2 norm of each AV ($\|z_l\|_{\ell_2}, l = 1, \ldots, 7$) under the ADP controllers associated with different weighting matrices. (a) $Q = 0.1\ I_6$. (b) $Q = 1I_6$. (c) $Q = 10I_6$. (d) $Q = 100I_6$.

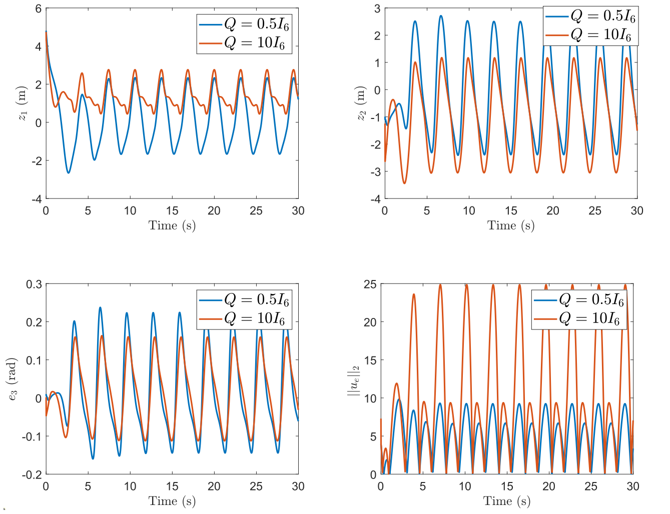


Fig. 11. Tracking errors of the following vehicle when the velocity of the leading vehicle is oscillating.

Definition 2: For a platoon consisting of N CAVs, it is strictly ℓ_2 string stable if

$$\|z_{l+1}\|_{\ell_2} \leq \|z_l\|_{\ell_2} \quad \forall l \in \{1, 2, \ldots, N\} \tag{49}$$

where $z_l(t) = [z_{l,1}(t), z_{l,2}(t)]^T$ is the 2-D tracking error of the lth CAV defined in (13), and

$$\|z_l\|_{\ell_2} = \sqrt{\int_0^\infty z_l^T(t) z_l(t) \mathrm{d}t}. \tag{50}$$

For the platoon in Fig. 9 starting from the same non-equilibrium state, we compare the string stability under different ADP controllers associated with the weighting matrices $Q = 0.1I_6$, $Q = 1I_6$, $Q = 10I_6$, and $Q = 100I_6$. It is seen from Fig. 10 that for all the Q matrices, the ℓ_2-norm keeps decreasing, i.e., (49) holds. This simulation shows that the ℓ_2 string stability is guaranteed by the ADP controllers.

D. Robustness Evaluation

In this simulation, we evaluate the robustness of the learned controller against exogenous disturbance from the leading

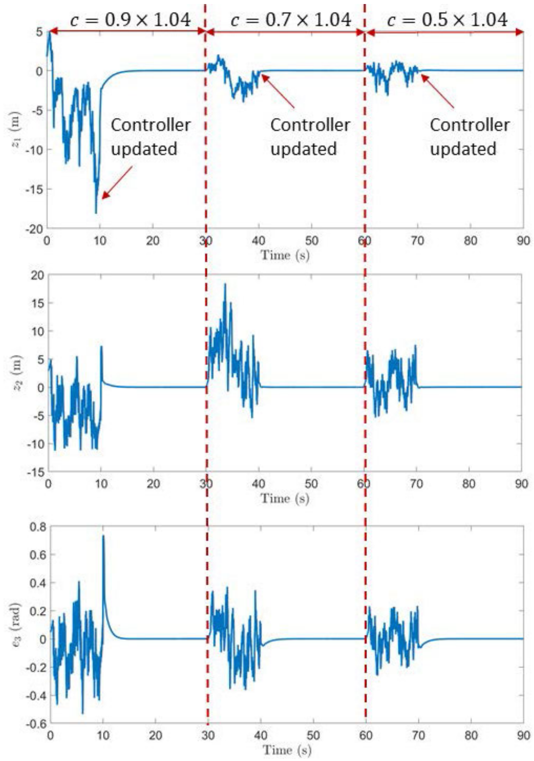


Fig. 12. Tracking errors of the following vehicle when the road condition is changing. The friction coefficient of the road changes at $t = 0$ s, $t = 30$ s, and $t = 60$ s. The controller is updated by Algorithm 2 at $t = 10$ s, $t = 40$ s, and $t = 70$ s.

vehicle. We compare the robustness of the two ADP controllers learned with different Q matrices. To be more specific, when $Q = 0.5I_6, 10I_6$, and $R = I_2$ is fixed, two different ADP controllers are generated separately by Algorithm 2 following the same procedure as Section VI-A. The robustness of the two ADP controllers is evaluated separately on the same circle road with the radius $r = 101.6$ m. When the simulation starts, the velocity of the leading vehicle oscillates, i.e., $v_L(t) = 10 + 10\sin(2t)$ m/s. To avoid the singularity when computing the feedforward and feedback controllers, the measurement of the velocity $v_L(t)$ $(v_F(t))$ is set as $\text{sign}(v_L(t))\cdot 0.1$ $(\text{sign}(v_F(t))\cdot 0.1)$ when $|v_L(t)| < 0.1$ $(|v_F(t)| < 0.1)$.

The tracking errors and the norm of the driving input u_e are shown in Fig. 11. It is shown that for $Q = 10I_6$, the tracking errors are attenuated more compared with $Q = 0.5\ I_6$. However, this attenuation is at the cost of increasing the magnitude of the control input as shown in Fig. 11, which usually causes more energy consumption. Indeed, in the definition of the performance index (17), Q is related to the tracking errors, and R is related to the energy consumption. Increasing Q will amplify the magnitude of minimizing tracking errors, accompanied by increasing driving inputs.

E. Adaption to Different Road Conditions

In this simulation, we analyze the adaptivity of the proposed data-driven control method to different road conditions with changing friction coefficients. The leading vehicle is driving at a speed of $v_L(t) = 20$ m/s on a circle with the radius $r = 100$ m. Initially, the friction coefficient is set as $c = 0.9 \times 1.04$ when $t = 0 \sim 30$ s. Then, the friction coefficient is changed to $c = 0.7 \times 1.04$ during $t = 30 \sim 60$ s. Finally, the friction coefficient is switched to $c = 0.5 \times 1.04$. After driving on the changed road for 10 s, the data are collected and the controller is updated by Algorithm 2. It is seen from Fig. 12 that with the updated controller, the tracking errors are quickly converging to zero on different road conditions with different friction coefficients. Therefore, using vehicle data and the PI algorithm allows the controller to be refined and adapt to different road conditions.

F. Discussion

In Section VI-C, the controller is learned from a pair of leading-following vehicles and deployed to homogeneous AVs. However, in practice, AVs are often heterogeneous, with varying dynamics and control requirements. A better solution would be to learn individual controllers for each AV within a platoon of heterogeneous vehicles. When scaling the proposed data-driven control approach to larger platoons of heterogeneous AVs, several limitations and challenges emerge. First, the computational complexity of the algorithm and the volume of data required for training may increase significantly as the number of AVs grows. Additionally, the method relies on reliable V2V communication for data collection and control coordination. In practice, communication delays, packet losses, or other disruptions could degrade the performance of the control strategy, leading to suboptimal outcomes. Addressing these scalability and robustness issues may require further refinement of the algorithm, such as incorporating distributed control techniques or enhancing its robustness to handle intermittent data more effectively.

VII. Conclusion

In this article, a data-driven adaptive optimal control approach is proposed to force an AV to stay within a safe but minimum distance from its leading vehicle and, at the same, in the lane. In particular, using the motion data of the leading and following vehicles, we propose a two-phase data-driven policy iteration algorithm to learn adaptive suboptimal controllers in the absence of the exact model knowledge of the AV. Compared with traditional model-based control approaches, the proposed data-driven algorithm avoids the need to build a mathematical model for the AV and identify the physical parameters of the model. Compared with conventional reinforcement learning approaches, the proposed physics-inspired learning algorithm can guarantee the stability of the system consisting of the leading and following vehicles and can generate a sequence of suboptimal controllers that converge to the unknown optimal controller. We compared the proposed learning-based control method with a particular nonlinear control approach known as feedback linearization and tested the controller under various traffic scenarios via SUMO simulation. The results have shown that using only 10.5 s data from the vehicles, the proposed approach can learn an improved controller and the AV can accurately track the leading vehicle.

Appendix A
Auxiliary Results for Vehicle Dynamics

The expressions of the variables related to the physical parameters are

$$\lambda_1 = -\frac{cM(C_f l_f + C_r l_r) g_a l_f l_r}{I_z(l_f + l_r)}, \quad \lambda_2 = \frac{cM(C_r - C_f) g_a l_r l_f}{I_z(l_f + l_r)}$$
$$\lambda_3 = \frac{cMC_f g_a l_f l_r}{I_z(l_f + l_r)} \tag{A.1}$$

and

$$\theta_1 = \frac{c(C_r - C_f) g_a l_f l_r}{l_f + l_r}, \quad \theta_2 = -\frac{c g_a (C_r l_f + C_f l_r)}{l_f + l_r},$$
$$\theta_3 = \frac{cC_f g_a l_r}{l_f + l_r}. \tag{A.2}$$

The expressions of $f_l(v_F, \omega_F)$ and $g_l(v_F)$ are

$$f_l(v_F, \omega_F) = \begin{bmatrix} \lambda_1 \dfrac{\omega_F}{v_F} \\ -\omega_F + \theta_1 \dfrac{\omega_F}{v_F^2} \end{bmatrix}, \quad g_l(v_F) = \begin{bmatrix} \lambda_3 & \lambda_2 \\ \dfrac{\theta_3}{v_F} & \dfrac{\theta_2}{v_F} \end{bmatrix}. \tag{A.3}$$

Appendix B
Derivation of the Error System

Considering (9), the derivative of γ is

$$\dot{\gamma} = \frac{\kappa t_s \dot{v}_F}{1 + \kappa^2 d^2} = \cos^2(\gamma) \kappa t_s \dot{v}_F. \tag{B.1}$$

From Fig. 1, it is seen that $s = (1/\cos(\gamma) - 1)/\kappa$. Then, it follows from (9) and (10) that:

$$s = \begin{cases} 0, & \kappa = 0 \\ \dfrac{-1 + \sqrt{1 + \kappa^2 d^2}}{\kappa}, & \kappa \neq 0 \end{cases} \tag{B.2a}$$
$$\dot{s} = \frac{\kappa d t_s}{\sqrt{1 + \kappa^2 d^2}} \dot{v}_F = \sin(\gamma) t_s \dot{v}_F. \tag{B.2b}$$

Following (1), (3), (B.1), and (B.2b), the derivatives of the tracking errors are

$$\begin{aligned} \dot{e}_1 &= v_L \cos(\varphi_L + \beta_L) + t_s \sin(\gamma) \sin(\varphi_L + \beta_F) \dot{v}_F \\ &\quad + s \cos(\varphi_L + \beta_L)(\omega_L + v_{\beta,L}) - v_F \cos(\varphi_F + \beta_L) \\ &\quad - t_s \cos(\varphi_F + \beta_F) \dot{v}_F + d \sin(\varphi_F + \beta_F)(\omega_F + \dot{\beta}_F) \\ \dot{e}_2 &= v_L \sin(\varphi_L + \beta_L) - t_s \sin(\gamma) \cos(\varphi_L + \beta_L) \dot{v}_F \\ &\quad + s \sin(\varphi_L + \beta_L)(\omega_L + v_{\beta,L}) - v_F \sin(\varphi_F + \beta_F) \\ &\quad - t_s \sin(\varphi_F + \beta_F) \dot{v}_F - d \cos(\varphi_F + \beta_F)(\omega_F + \dot{\beta}_F) \\ \dot{e}_3 &= e_6 - \cos^2(\gamma) \kappa t_s \dot{v}_F + v_{\beta,L} - \dot{\beta}_F \\ \dot{e}_4 &= a_L - a_F \\ \dot{e}_5 &= a_{\varphi,L} - \dot{\omega}_F \\ \dot{e}_6 &= \dot{\beta}_{d,F} - \dot{\beta}_F. \end{aligned} \tag{B.3}$$

Then, by (B.3) and (13), the derivatives of z_1 and z_2 are

$$\begin{aligned} \dot{z}_1 &= (v_L + s\omega_L) \cos(e_3) - (v_L - e_4) \cos(\gamma) \\ &\quad - d \sin(\gamma)(\omega_L - e_5) + z_2(\omega_L - e_5) \\ &\quad + \left[\sin(\gamma) \sin(e_3) - \cos(\gamma) + \cos^2(\gamma) \kappa z_2\right] t_s a_F \\ &\quad + (z_2 - d \sin(\gamma)) \dot{\beta}_F + s \cos(e_3) v_{\beta,L} \\ \dot{z}_2 &= (v_L + s\omega_L) \sin(e_3) + (v_L - e_4) \sin(\gamma) \\ &\quad - d \cos(\gamma)(\omega_L - e_5) - z_1(\omega_L - e_5) \\ &\quad - \left[\sin(\gamma) \cos(e_3) + \sin(\gamma) + \cos^2(\gamma) \kappa z_1\right] t_s a_F \\ &\quad - (z_1 + d \cos(\gamma)) \dot{\beta}_F + s \sin(e_3) v_{\beta,L}. \end{aligned} \tag{B.4}$$

It follows from (8) that:

$$\beta_{L,d} = \frac{\theta_1 \lambda_3 - \lambda_1 \theta_3}{\theta_2 \lambda_3 - \lambda_2 \theta_3} \frac{\omega_L}{v_L} - \frac{\lambda_3}{\theta_2 \lambda_3 - \lambda_2 \theta_3} \omega_L v_L. \tag{B.5}$$

Combining (B.3), (B.4), (B.5), and (14), the error system is finally derived as (B.6), as shown at the bottom of the next page.

Appendix C
Proof of Proposition 1

Define

$$\hat{\varepsilon} = \Lambda \left[\mathrm{vec}\left(\hat{\sigma}_f^T\right)^T, \mathrm{vec}\left(\hat{\sigma}_g^T\right)^T\right]^T - \Xi. \tag{C.1}$$

As $\mathrm{vec}(\hat{\sigma}_f^T)^T$ and $\mathrm{vec}(\hat{\sigma}_g^T)^T$ in (28) are the least-square solutions to (25), we have

$$\begin{aligned} \varepsilon^T \varepsilon &= \left(\Lambda \left[\mathrm{vec}\left(\hat{\sigma}_f^T\right)^T, \mathrm{vec}\left(\hat{\sigma}_g^T\right)^T\right]^T - \Xi\right)^T \\ &\quad \times (\Lambda \left[\mathrm{vec}\left(\hat{\sigma}_f^T\right)^T, \mathrm{vec}\left(\hat{\sigma}_g^T\right)^T\right]^T - \Xi) \geq \hat{\varepsilon}^T \hat{\varepsilon}. \end{aligned} \tag{C.2}$$

Subtracting (C.1) from (25) yields

$$\hat{\varepsilon} - \varepsilon = \Lambda \left[\mathrm{vec}(\hat{\sigma}_f^T)^T - \mathrm{vec}(\sigma_f^T)^T, \mathrm{vec}(\hat{\sigma}_g^T)^T - \mathrm{vec}(\sigma_g^T)^T\right]^T. \tag{C.3}$$

As a consequence

$$\begin{aligned} &(\hat{\varepsilon} - \varepsilon)^T (\hat{\varepsilon} - \varepsilon) \\ &= \left[\mathrm{vec}\left(\hat{\sigma}_f^T\right)^T - \mathrm{vec}\left(\sigma_f^T\right)^T, \mathrm{vec}\left(\hat{\sigma}_g^T\right)^T - \mathrm{vec}\left(\sigma_g^T\right)^T\right] \Lambda^T \Lambda \\ &\quad \times \left[\mathrm{vec}\left(\hat{\sigma}_f^T\right)^T - \mathrm{vec}\left(\sigma_f^T\right)^T, \mathrm{vec}\left(\hat{\sigma}_g^T\right)^T - \mathrm{vec}\left(\sigma_g^T\right)^T\right]^T. \end{aligned} \tag{C.4}$$

Combining (C.2) and (C.4), we have

$$\begin{aligned} &\left[\mathrm{vec}\left(\hat{\sigma}_f^T\right)^T - \mathrm{vec}\left(\sigma_f^T\right)^T, \mathrm{vec}\left(\hat{\sigma}_g^T\right)^T - \mathrm{vec}\left(\sigma_g^T\right)^T\right] \Lambda^T \Lambda \\ &\quad \times \left[\mathrm{vec}\left(\hat{\sigma}_f^T\right)^T - \mathrm{vec}\left(\sigma_f^T\right)^T, \mathrm{vec}\left(\hat{\sigma}_g^T\right)^T - \mathrm{vec}\left(\sigma_g^T\right)^T\right]^T \\ &\leq 2\varepsilon^T \varepsilon + 2\hat{\varepsilon}^T \hat{\varepsilon} \leq 4\varepsilon^T \varepsilon. \end{aligned} \tag{C.5}$$

It follows from Assumption 1 that:

$$\begin{aligned} &\left[\mathrm{vec}\left(\hat{\sigma}_f^T\right)^T - \mathrm{vec}\left(\sigma_f^T\right)^T, \mathrm{vec}\left(\hat{\sigma}_g^T\right)^T - \mathrm{vec}\left(\sigma_g^T\right)^T\right] \\ &\quad \times \left[\mathrm{vec}\left(\hat{\sigma}_f^T\right)^T - \mathrm{vec}\left(\sigma_f^T\right)^T, \mathrm{vec}\left(\hat{\sigma}_g^T\right)^T - \mathrm{vec}\left(\sigma_g^T\right)^T\right]^T \\ &\leq \frac{4}{\eta} \max_{1 \leq k \leq K} [\varepsilon]_k^2. \end{aligned} \tag{C.6}$$

Since $\lim_{N_f\to\infty} e_f(v_F,\omega_F) = 0$ and $\lim_{N_g\to\infty} e_g(v_F) = 0$ uniformly over $[v_F,\omega_F] \in \mathcal{V}$, it follows from (26c) that $\lim_{N_f,N_g\to\infty}[\varepsilon]_k = 0$. Consequently, $\lim_{N_f,N_g\to\infty}(\max_{1\le k\le K}[\varepsilon]_k^2) = 0$, and it follows from (C.6) that:

$$\lim_{N_f,N_g\to\infty}\left\|\left[\operatorname{vec}\left(\hat{\sigma}_f^T\right)^T - \operatorname{vec}\left(\sigma_f^T\right)^T, \operatorname{vec}\left(\hat{\sigma}_g^T\right)^T - \operatorname{vec}\left(\sigma_g^T\right)^T\right]\right\|$$
$$= 0.$$

Therefore, by (23), we conclude that Proposition 1 holds.

Appendix D
Proof of Proposition 3

Define

$$\tau_i = \Theta_i\begin{bmatrix}\hat{\rho}_i \\ \operatorname{vec}(\hat{\mu}_{i+1})\end{bmatrix} - \Gamma_i. \tag{D.1}$$

Subtracting (D.1) from (41), we have

$$\Theta_i\begin{bmatrix}\rho_i - \hat{\rho}_i \\ \operatorname{vec}(\mu_{i+1} - \hat{\mu}_{i+1})\end{bmatrix} = \Delta_i + \xi_i - \tau_i. \tag{D.2}$$

Since $\begin{bmatrix}\hat{\rho}_i \\ \operatorname{vec}(\hat{\mu}_{i+1})\end{bmatrix}$ is the least-square solution that minimizes $\tau_i^T\tau_i$, we have

$$\tau_i^T\tau_i \le (\Delta_i + \xi_i)^T(\Delta_i + \xi_i). \tag{D.3}$$

Combining (D.2) with (D.3), we have

$$\begin{bmatrix}\rho_i - \hat{\rho}_i \\ \operatorname{vec}(\mu_{i+1} - \hat{\mu}_{i+1})\end{bmatrix}^T \Theta_i^T\Theta_i\begin{bmatrix}\rho_i - \hat{\rho}_i \\ \operatorname{vec}(\mu_{i+1} - \hat{\mu}_{i+1})\end{bmatrix}$$
$$= (\Delta_i + \xi_i - \tau_i)^T(\Delta_i + \xi_i - \tau_i) \le 4(\Delta_i + \xi_i)^T(\Delta_i + \xi_i). \tag{D.4}$$

Under Assumption 2, (D.4) is derived as

$$\begin{aligned}\begin{bmatrix}\rho_i - \hat{\rho}_i \\ \operatorname{vec}(\mu_{i+1} - \hat{\mu}_{i+1})\end{bmatrix}^T&\begin{bmatrix}\rho_i - \hat{\rho}_i \\ \operatorname{vec}(\mu_{i+1} - \hat{\mu}_{i+1})\end{bmatrix} \\ &\le \frac{4}{P\eta}(\Delta_i + \xi_i)^T(\Delta_i + \xi_i) \\ &\le \frac{4}{\eta}\max_{1\le p\le P}[\Delta_i + \xi_i]_p^2.\end{aligned} \tag{D.5}$$

Next, we will prove the proposition by induction. When $i = 1$, $\alpha_1 = \hat{\alpha}_1$. Following Proposition 1, $\lim_{N_f,N_g\to\infty}\hat{u}_d = u_d$. By the definition of $\Delta_{i,p}$ in (37), we have $\lim_{N_f,N_g\to\infty}\Delta_1 = 0$. In addition, since $\lim_{N_V\to\infty} e_{V,1}(\chi_L, e) = 0$ and $\lim_{N_\alpha\to\infty} e_{\alpha,2}(\chi_L, e) = 0$, it follows from (40) that $\lim_{N_V,N_\alpha\to\infty}\xi_1 = 0$. Hence, $\lim_{N_f,N_g,N_V,N_\alpha\to\infty}\begin{bmatrix}\rho_i - \hat{\rho}_i \\ \operatorname{vec}(\mu_{i+1} - \hat{\mu}_{i+1})\end{bmatrix} = 0$ follows from (D.5). Considering (38), it is obtained that (46) holds for $i = 1$.

By induction, suppose (46) holds for some $i > 1$. Then, it follows from (37) and $\lim_{N_f,N_g,N_V,N_\alpha\to\infty}\hat{\alpha}_{i+1}(\chi_L, e) = \alpha_{i+1}(\chi_L, e)$ that $\lim_{N_f,N_g,N_V,N_\alpha\to\infty}\Delta_{i+1} = 0$. In addition, since $\lim_{N_V\to\infty} e_{V,i+1}(\chi_L, e) = 0$ and $\lim_{N_\alpha\to\infty} e_{\alpha,i+2}(\chi_L, e) = 0$, it follows from (40) that $\lim_{N_V,N_\alpha\to\infty}\xi_{i+1} = 0$. Hence, $\lim_{N_f,N_g,N_V,N_\alpha\to\infty}\begin{bmatrix}\rho_{i+1} - \hat{\rho}_{i+1} \\ \operatorname{vec}(\mu_{i+2} - \hat{\mu}_{i+2})\end{bmatrix} = 0$ follows from (D.5). Considering (38), it is obtained that (46) holds for $i + 1$. The proof is thus completed.

Acknowledgment

The U.S. Government assumes no liability for the contents or use thereof.

$$\begin{aligned}
\dot{z}_1 &= (v_L + s\omega_L)\cos(e_3) - (v_L - e_4)\cos(\gamma) - d\sin(\gamma)(\omega_L - e_5) + z_2(\omega_L - e_5) \\
&\quad + (z_2 - d\sin(\gamma))\left[e_5 + \theta_1\frac{-v_L^2 e_5 + 2v_L\omega_L e_4 - e_4^2\omega_L}{v_L^2(v_L - e_4)^2} + \theta_2\frac{\beta_{F,d}e_4 - v_L e_6}{v_L(v_L - e_4)} + \frac{\theta_3 u_{2,d}e_4}{v_L(v_L - e_4)}\right] \\
&\quad + \left[\sin(\gamma)\sin(e_3) - \cos(\gamma) + \cos^2(\gamma)\kappa z_2\right]t_s u_{1,e} + (z_2 - d\sin(\gamma))\frac{\theta_3}{(v_L - e_4)}u_{2,e} + s\cos(e_3)v_{\beta,L} \\
\dot{z}_2 &= (v_L + s\omega_L)\sin(e_3) + (v_L - e_4)\sin(\gamma) - d\cos(\gamma)(\omega_L - e_5) - z_1(\omega_L - e_5) \\
&\quad - (z_1 + d\cos(\gamma))\left[e_5 + \theta_1\frac{-v_L^2 e_5 + 2v_L\omega_L e_4 - e_4^2\omega_L}{v_L^2(v_L - e_4)^2} + \theta_2\frac{\beta_{F,d}e_4 - v_L e_6}{v_L(v_L - e_4)} + \frac{\theta_3 u_{2,d}e_4}{v_L(v_L - e_4)}\right] \\
&\quad - \left[\sin(\gamma)\cos(e_3) + \sin(\gamma) + \cos^2(\gamma)\kappa z_1\right]t_s u_{1,e} - (z_1 + d\cos(\gamma))\frac{\theta_3}{v_L - e_4}u_{2,e} + s\sin(e_3)v_{\beta,L} \\
\dot{e}_3 &= -\theta_1\frac{-v_L^2 e_5 + 2v_L\omega_L e_4 - e_4^2\omega_L}{v_L^2(v_L - e_4)^2} - \theta_2\frac{\beta_{F,d}e_4 - v_L e_6}{v_L(v_L - e_4)} - \frac{\theta_3 u_{2,d}e_4}{v_L(v_L - e_4)} - \cos^2(\gamma)\kappa t_s u_{1,e} - \frac{\theta_3}{v_L - e_4}u_{2,e} + v_{\beta,L} \\
\dot{e}_4 &= -u_{1,e} + a_L \\
\dot{e}_5 &= -\lambda_1\frac{\omega_L e_4 - v_L e_5}{v_L(v_L - e_4)} + \lambda_2 e_6 - \lambda_3 u_{2,e} + a_{\varphi,L} \\
\dot{e}_6 &= -e_5 - \theta_1\frac{-v_L^2 e_5 + 2v_L\omega_L e_4 - e_4^2\omega_L}{v_L^2(v_L - e_4)^2} - \theta_2\frac{\beta_{F,d}e_4 - v_L e_6}{v_L(v_L - e_4)} - \frac{\theta_3 u_{2,d}e_4}{v_L(v_L - e_4)} - \frac{\theta_3}{v_L - e_4}u_{2,e} \\
&\quad - \frac{\theta_1\lambda_3 - \lambda_1\theta_3}{\theta_2\lambda_3 - \lambda_2\theta_3}\frac{v_L a_{\varphi,L} - \omega_L a_L}{v_L^2} - \frac{\lambda_3}{\theta_2\lambda_3 - \lambda_2\theta_3}(v_L a_{\varphi,L} + \omega_L a_L)
\end{aligned} \tag{B.6}$$

References

[1] S. E. Shladover, "Reasons for operating AHS vehicles in platoons," in *Automated Highway Systems*, P. A. Ioannou, Ed., Berlin, Germany: Springer, 1997, pp. 11–27.

[2] B. van Arem, C. J. G. van Driel, and R. Visser, "The impact of cooperative adaptive cruise control on traffic-flow characteristics," *IEEE Trans. Intell. Transp. Syst.*, vol. 7, no. 4, pp. 429–436, Dec. 2006.

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

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

[5] G. Orosz, "Connected cruise control: Modeling, delay effects, and nonlinear behavior," *Vehicle Syst. Dyn.*, vol. 54, pp. 1–30, Jun. 2016.

[6] M. Huang, Z.-P. Jiang, and K. Ozbay, "Learning-based adaptive optimal control for connected vehicles in mixed traffic: Robustness to driver reaction time," *IEEE Trans. Cybern.*, vol. 52, no. 6, pp. 5267–5277, Jun. 2022.

[7] R. Rajamani, H.-S. Tan, B. K. Law, and W.-B. Zhang, "Demonstration of integrated longitudinal and lateral control for the operation of automated vehicles in platoons," *IEEE Trans. Control Syst. Technol.*, vol. 8, no. 4, pp. 695–708, Jul. 2000.

[8] V. Turri, A. Carvalho, H. E. Tseng, K. H. Johansson, and F. Borrelli, "Linear model predictive control for lane keeping and obstacle avoidance on low curvature roads," in *Proc. 16th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2013, pp. 378–383.

[9] Y. Wang, Q. Shao, J. Zhou, H. Zheng, and H. Chen, "Longitudinal and lateral control of autonomous vehicles in multi-vehicle driving environments," *IET Intell. Transp. Syst.*, vol. 14, no. 8, pp. 924–935, Aug. 2020.

[10] K. Chen et al., "A hierarchical hybrid system of integrated longitudinal and lateral control for intelligent vehicles," *ISA Trans.*, vol. 106, pp. 200–212, Nov. 2020.

[11] J. Nilsson, M. Brännström, J. Fredriksson, and E. Coelingh, "Longitudinal and lateral control for automated yielding maneuvers," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 5, pp. 1404–1414, May 2016.

[12] R. Kianfar, M. Ali, P. Falcone, and J. Fredriksson, "Combined longitudinal and lateral control design for string stable vehicle platooning within a designated lane," in *Proc. 17th Int. IEEE Conf. Intell. Transp. Syst. (ITSC)*, Oct. 2014, pp. 1003–1008.

[13] L. Xu, Y. Wang, H. Sun, J. Xin, and N. Zheng, "Integrated longitudinal and lateral control for Kuafu-II autonomous vehicle," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 7, pp. 2032–2041, Jul. 2016.

[14] M. Huang, W. Gao, and Z.-P. Jiang, "Connected cruise control with delayed feedback and disturbance: An adaptive dynamic programming approach," *Int. J. Adapt. Control Signal Process.*, vol. 33, no. 2, pp. 356–370, Feb. 2019.

[15] X. Li, "Trade-off between safety, mobility and stability in automated vehicle following control: An analytical method," *Transp. Res. B, Methodol.*, vol. 166, pp. 1–18, Dec. 2022.

[16] 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, Methodol.*, vol. 128, pp. 69–86, Oct. 2019.

[17] Y. Zhou, S. Ahn, M. Wang, and S. Hoogendoorn, "Stabilizing mixed vehicular platoons with connected automated vehicles: An H-infinity approach," *Transp. Res. B, Methodol.*, vol. 132, pp. 152–170, Feb. 2020.

[18] M. Huang, M. Zhao, P. Parikh, Y. Wang, K. Ozbay, and Z.-P. Jiang, "Reinforcement learning for vision-based lateral control of a self-driving car," in *Proc. IEEE 15th Int. Conf. Control Autom. (ICCA)*, Jul. 2019, pp. 1126–1131.

[19] Y. S. Son, W. Kim, S. Lee, and C. C. Chung, "Robust multirate control scheme with predictive virtual lanes for lane-keeping system of autonomous highway driving," *IEEE Trans. Veh. Technol.*, vol. 64, no. 8, pp. 3378–3391, Aug. 2015.

[20] Ö. Tunçer, L. Güvenç, F. Coskun, and E. Karsligil, "Vision based lane keeping assistance control triggered by a driver inattention monitor," in *Proc. IEEE Int. Conf. Syst., Man Cybern.*, Oct. 2010, pp. 289–297.

[21] S. Di Cairano, H. E. Tseng, D. Bernardini, and A. Bemporad, "Vehicle yaw stability control by coordinated active front steering and differential braking in the tire sideslip angles domain," *IEEE Trans. Control Syst. Technol.*, vol. 21, no. 4, pp. 1236–1248, Jul. 2013.

[22] R. T. O'Brien, P. A. Iglesias, and T. J. Urban, "Vehicle lateral control for automated highway systems," *IEEE Trans. Control Syst. Technol.*, vol. 4, no. 3, pp. 266–273, May 1996.

[23] M. Mirzaei, "A new strategy for minimum usage of external yaw moment in vehicle dynamic control system," *Transp. Res. C, Emerg. Technol.*, vol. 18, pp. 213–224, Apr. 2010.

[24] H. Du, N. Zhang, and F. Naghdy, "Velocity-dependent robust control for improving vehicle lateral dynamics," *Transp. Res. C, Emerg. Technol.*, vol. 19, no. 3, pp. 454–468, Jun. 2011.

[25] E. Lim and J. Hedrick, "Lateral and longitudinal vehicle control coupling for automated vehicle operation," in *Proc. Amer. Control Conf.*, vol. 5, 1999, pp. 3676–3680.

[26] V. Shia et al., "Semiautonomous vehicular control using driver modeling," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 6, pp. 2696–2709, Dec. 2014.

[27] H. Zhou, F. Jia, H. Jing, Z. Liu, and L. Güvenç, "Coordinated longitudinal and lateral motion control for four wheel independent motor-drive electric vehicle," *IEEE Trans. Veh. Technol.*, vol. 67, no. 5, pp. 3782–3790, May 2018.

[28] C. Wei et al., "Risk-based autonomous vehicle motion control with considering human driver's behaviour," *Transp. Res. C, Emerg. Technol.*, vol. 107, pp. 1–14, Oct. 2019.

[29] R. Zhang, Y. Ma, Z. Li, R. Malekian, and M. A. Sotelo, "Energy dissipation based longitudinal and lateral coupling control for intelligent vehicles," *IEEE Intell. Transp. Syst. Mag.*, vol. 10, no. 2, pp. 121–133, Summer. 2018.

[30] A. Chebly, R. Talj, and A. Charara, "Coupled longitudinal and lateral control for an autonomous vehicle dynamics modeled using a robotics formalism," *IFAC-PapersOnLine*, vol. 50, no. 1, pp. 12526–12532, Jul. 2017.

[31] A. Chebly, R. Talj, and A. Charara, "Coupled longitudinal/lateral controllers for autonomous vehicles navigation, with experimental validation," *Control Eng. Pract.*, vol. 88, pp. 79–96, Jul. 2019.

[32] A. Bayuwindra, E. Lefeber, J. Ploeg, and H. Nijmeijer, "Extended look-ahead tracking controller with orientation-error observer for vehicle platooning," *IEEE Trans. Intell. Transp. Syst.*, vol. 21, no. 11, pp. 4808–4821, Nov. 2020.

[33] A. Bayuwindra, J. Ploeg, E. Lefeber, and H. Nijmeijer, "Combined longitudinal and lateral control of car-like vehicle platooning with extended look-ahead," *IEEE Trans. Control Syst. Technol.*, vol. 28, no. 3, pp. 790–803, May 2020.

[34] Y. Wang, N. Bian, L. Zhang, and H. Chen, "Coordinated lateral and longitudinal vehicle-following control of connected and automated vehicles considering nonlinear dynamics," *IEEE Control Syst. Lett.*, vol. 4, no. 4, pp. 1054–1059, Oct. 2020.

[35] S. Aradi, "Survey of deep reinforcement learning for motion planning of autonomous vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 23, no. 2, pp. 740–759, Feb. 2022.

[36] Y. Jiang and Z. P. Jiang, *Robust Adaptive Dynamic Programming*. Hoboken, NJ, USA: Wiley-IEEE Press, 2017.

[37] Z.-P. Jiang, T. Bian, and W. Gao, "Learning-based control: A tutorial and some recent results," *Found. Trends Syst. Control*, vol. 8, no. 3, pp. 176–284, 2020.

[38] L. Cui, T. Başar, and Z.-P. Jiang, "Robust reinforcement learning for risk-sensitive linear quadratic Gaussian control," *IEEE Trans. Autom. Control*, vol. 69, no. 11, pp. 7678–7693, Nov. 2024, doi: 10.1109/TAC.2024.3397928.

[39] T. Liu, L. Cui, B. Pang, and Z.-P. Jiang, "A unified framework for data-driven optimal control of connected vehicles in mixed traffic," *IEEE Trans. Intell. Vehicles*, vol. 8, no. 8, pp. 4131–4145, Aug. 2023.

[40] W. Gao, Z. Jiang, and K. Ozbay, "Data-driven adaptive optimal control of connected vehicles," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 5, pp. 1122–1133, May 2017.

[41] W. Gao, J. Gao, K. Ozbay, and Z.-P. Jiang, "Reinforcement-learning-based cooperative adaptive cruise control of buses in the Lincoln tunnel corridor with time-varying topology," *IEEE Trans. Intell. Transp. Syst.*, vol. 20, no. 10, pp. 3796–3805, Oct. 2019.

[42] W. Sun, X. Wang, and C. Zhang, "A model-free control strategy for vehicle lateral stability with adaptive dynamic programming," *IEEE Trans. Ind. Electron.*, vol. 67, no. 12, pp. 10693–10701, Dec. 2020.

[43] W. Gao and Z.-P. Jiang, "Learning-based adaptive optimal tracking control of strict-feedback nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2614–2624, Jun. 2018.

[44] J. Huang, *Nonlinear Output Regulation: Theory and Applications*. Philadelphia, PA, USA: Society for Industrial and Applied Mathematics, 2004.

[45] D. Liberzon, *Calculus of Variations and Optimal Control Theory: A Concise Introduction*. Princeton, NJ, USA: Princeton Univ. Press, 2011.

[46] P. A. Lopez et al., “Microscopic traffic simulation using SUMO,” in *Proc. 21st Int. Conf. Intell. Transp. Syst. (ITSC)*, Nov. 2018, pp. 2575–2582.

[47] M. Althoff, M. Koschi, and S. Manzinger, “CommonRoad: Composable benchmarks for motion planning on roads,” in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2017, pp. 719–726.

[48] R. Rajamani, *Vehicle Dynamics and Control*, 2nd ed., New York, NY, USA: Springer, 2012.

[49] G. J. L. Naus, R. P. A. Vugts, J. Ploeg, M. J. G. van de Molengraft, and M. Steinbuch, “String-stable CACC design and experimental validation: A frequency-domain approach,” *IEEE Trans. Veh. Technol.*, vol. 59, no. 9, pp. 4268–4279, Nov. 2010.

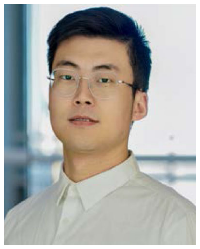
[50] Z. P. Jiang and H. Nijmeijer, “Tracking control of mobile robots: A case study in backstepping,” *Automatica*, vol. 33, no. 7, pp. 1393–1399, Jul. 1997.

[51] M. J. D. Powell, *Approximation Theory and Methods*. Cambridge, U.K.: Cambridge Univ. Press, 1981.

[52] K. J. Aström and B. Wittenmark, *Adaptive Control*, 2nd ed., Reading, MA, USA: Addison-Wesley, 1997.

[53] H. K. Khalil, *Nonlinear Systems*, 3rd ed., Upper Saddle River, NJ, USA: Prentice-Hall, 2002.

[54] C. C. Pugh, *Real Mathematical Analysis*, 2nd ed., Cham, Switzerland: Springer, 2015.



Leilei Cui (Member, IEEE) received the B.Sc. degree in automation from Northwestern Polytechnical University, Xi’an, China, in 2016, the M.Sc. degree in control science and engineering from Shanghai Jiao Tong University, Shanghai, China, in 2019, and the Ph.D. degree in electrical engineering from New York University, Brooklyn, NY, USA, in 2024.

He is now a Post-Doctoral Associate at the Massachusetts Institute of Technology, Cambridge, MA, USA. His research interests include optimal control, reinforcement learning, and adaptive dynamic programming.

Sayan Chakraborty (Graduate Student Member, IEEE) received the B.Tech. degree in electrical engineering from the National Institute of Technology, Silchar, India, in 2017, and the M.Tech. degree in electrical engineering with a specialization in systems and control from the Indian Institute of Technology Hyderabad, Kandi, India, in 2021. He is currently pursuing the Ph.D. degree with the Department of Electrical and Computer Engineering, New York University, Brooklyn, NY, USA, under the guidance of Professor Zhong-Ping Jiang.

His research interests include data-driven control, adaptive dynamic programming, and their applications to autonomous vehicles.

Kaan Ozbay (Member, IEEE) joined as a tenured Full Professor with the Civil and Urban Engineering, Tandon School of Engineering, and the Center for Urban Science and Progress (CUSP), New York University (NYU), Brooklyn, NY, USA, in 2013. Prior to that he was a tenured Full Professor at the Department of Civil and Environmental Engineering, Rutgers University, New Brunswick, NJ, USA, where he joined as an Assistant Professor, in July 1996. In 2008, he was a Visiting Scholar at the Operations Research and Financial Engineering (ORFE) Department, Princeton University, Princeton, NJ, USA. He is currently the Founding Director of the C2SMARTER Tier-1 University Transportation Center, Tandon School of Engineering, NYU. Since 1994, he has been the Principal Investigator and Co-Principal Investigator of 125 research projects funded at a level of more than U.S. $35M by USDOT, National Science Foundation, NCHRP, NJDOT, NY State DOT, NYC DOT, New Jersey Highway Authority, FHWA, VDOT, Department of Homeland Security, among others. He has co-authored four books and published approximately 500 refereed papers in scholarly journals and conference proceedings. His research interests include transportation cover a wide range of topics including data-driven AI/ML applications in smart cities, development and calibration of large-scale complex transportation simulation models for the study of traffic incident and emergency management and traffic safety, feedback-based real-time traffic control, application of operations research techniques in transportation system network, and humanitarian inventory control.

Dr. Ozbay was a recipient of several awards including, the prestigious National Science Foundation (NSF) CAREER Award, the IBM Faculty Award in addition to several best paper and excellence in research awards such as the “Franz Edelman Finalist Award for Achievement in Operations Research and Management Science.” He is an Associate Editor of *ITS Journal* and serves as the Associate Editor for *Networks and Spatial Economic Journal* and *Transportmetrica B: Transportation Dynamics Journal*.

Zhong-Ping Jiang (Fellow, IEEE) received the M.Sc. degree in statistics from the University of Paris XI, Paris, France, in 1989, and the Ph.D. degree in automatic control and mathematics from ParisTech-Mines, Paris, in 1993, under the direction of Prof. Laurent Praly.

Currently, he is an Institute Professor at the Department of Electrical and Computer Engineering and an Affiliate Professor at the Department of Civil and Urban Engineering, Tandon School of Engineering, New York University, Brooklyn, NY, USA. His main research interests include stability theory, robust/adaptive/distributed nonlinear control, robust adaptive dynamic programming, reinforcement learning, and their applications to information, mechanical, and biological systems. In these fields, he has written six books and has authored/co-authored about 600 peer-reviewed journal and conference papers.

Prof. Jiang is a Foreign Member of the Academia Europaea (Academy of Europe), an Ordinary Member of the European Academy of Sciences and Arts, and a fellow of the IFAC, CAA, AAIA, and AAAS. He was a recipient of the prestigious Queen Elizabeth II Fellowship Award from the Australian Research Council, the CAREER Award from the U.S. National Science Foundation, the JSPS Invitation Fellowship from the Japan Society for the Promotion of Science, the Distinguished Overseas Chinese Scholar Award from the NSF of China, and several best paper awards. In 2022, he received the Excellence in Research Award from the NYU Tandon School of Engineering. He has served as the deputy editor-in-chief, a senior editor, and an associate editor for numerous journals, and is among the Clarivate Analytics Highly Cited Researchers and Stanford’s Top 2% Most Highly Cited Scientists.