# Adaptive Optimal Control of Heterogeneous CACC System With Uncertain Dynamics

Yuanheng Zhu, *Member, IEEE*, Dongbin Zhao, *Senior Member, IEEE*, and Zhiguang Zhong

***Abstract*— Cooperative adaptive cruise control (CACC), as an extension of adaptive cruise control, connects multiple vehicles in a platoon via wireless communication. In practice, different vehicles may have different dynamic parameters and their exact values are unknown/uncertain to designers. In this brief, we propose a new control structure that uses an estimate of dynamic parameters to transform the heterogeneous CACC problem into the regulation problem of error dynamics for each vehicle in the platoon. An adaptive optimal control is proposed to learn the optimal feedback based on online data. The position transfer function between adjacent vehicles is further analyzed in the frequency domain. By sum of squares programming, the minimum headway values that ensure the vehicle string stability are found. Experiments on numerical and complex systems validate our method.**

***Index Terms*— Adaptive optimal control, cooperative adaptive cruise control (CACC), heterogeneous platoon, string stability, sum-of-squares polynomial.**

## I. Introduction

OVER the past decades, explosion of vehicles has posed enormous challenges to current traffic infrastructure. Traffic jams are becoming a big problem in most cities and bigger towns. Among numerous approaches to enhance driving experience, adaptive cruise control (ACC) is a highly developed one [1]–[4]. It has been widely used in the luxury car market. ACC adjusts car velocity to follow the preceding vehicle. Commonly, a local distance sensor (e.g., radar) is mounted at the front bumper to measure the relative distance and relative velocity to its predecessor. ACC more concerns with safety and comfort and pays less attention to traffic throughput. For safety reasons, ACC requires a large intervehicle distance, which has a negative effect on traffic capacity. To shorten intervehicle distance, one solution is to share car information among vehicles to increase reaction rate. Motivated by that, cooperative ACC is proposed [5], [6]. Wireless communication network is set up to connect vehicles in the same traffic. Driving data are transmitted through the network. In addition to local measurements, cooperative ACC (CACC) takes other vehicles' behavior (e.g., acceleration and control input) into consideration. In this way, intervehicle distances are reduced and traffic throughput is increased. Effects of CACC on the relief of traffic congestion are studied in the literature [7], [8].

When multiple vehicles equipped with CACC form a platoon and drive along a longitudinal line, the string stability is critical. String stability refers to the attenuation of disturbance in upstream direction. If the platoon is string unstable, any disturbance from the leading vehicle will be amplified along vehicles. Then a halt or a collision may occur at the end of the platoon. The vehicle string stability can be evaluated by analyzing intervehicle transfer functions in the frequency domain [9]. Various CACC structures have been proposed and some of them have been validated on real vehicles [10], [11].

Unfortunately, existing CACC rarely considers the optimality. Their controllers are manually selected and may not be optimal in control efforts or cost criteria. Optimal control is to find a control policy that minimizes a cost function. If a system has linear dynamics and its cost selects a quadratic form, the problem is reduced by solving the famous algebraic Riccati equation (ARE). Levine and Athans [12] formulate the vehicle string as a blocked linear system and design an optimal linear feedback controller in a centralized fashion. Linear quadratic tracking theory is studied in [13] for a connected cruise control (CCC) system. In practice, vehicle dynamics is mostly unknown or uncertain, making it impossible to define the ARE. For this reason, model-free, or data-based optimal control methods are desired.

Adaptive dynamic programming (ADP) is capable of solving optimal control problems without dynamics information (see [14], [15] and references cited therein). Jiang and Jiang [16] present an adaptive optimal control for linear systems with completely unknown dynamics. They further extend the method to nonlinear systems in [17]. We solve nonlinear $H_\infty$ problems based on ADP in [18], [19] and apply ADP on crane systems in [20]. For cruise control, Wang *et al*. [21] use kernel-based least squares policy iteration to learn the optimal longitudinal controller with unknown dynamics and external disturbances. Gao *et al*. [22] apply a data-driven model-free approach for the adaptive optimal control of CCC. CCC system is composed of $n$ human-driven vehicles and one autonomous vehicle in the tail. The autonomous vehicle receives wireless information from multiple preceding vehicles [23]. When multiple CCC vehicles exist in the platoon and form a connected vehicle network, the traffic efficiency is proven to be improved in [24].

Manuscript received August 29, 2017; revised December 22, 2017; accepted February 25, 2018. Date of publication March 21, 2018; date of current version June 11, 2019. Manuscript received in final form February 27, 2018. This work was supported in part by National Natural Science Foundation of China under Grant 61603382, Grant 61573353, and Grant 61533017, and in part by Grant LY15F030008. Recommended by Associate Editor Y. Wang. *(Corresponding author: Dongbin Zhao.)*

Y. Zhu and D. Zhao are with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: yuanheng.zhu@ia.ac.cn; dongbin.zhao@ia.ac.cn).

Z. Zhong is with the College of Science and Technology, Ningbo University, Ningbo 315212, China.

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TCST.2018.2811376

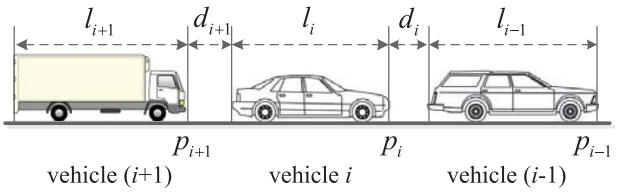


Fig. 1. Schematic of a vehicle platoon equipped with CACC.

The main contribution of this brief is to propose an adaptive optimal control framework for heterogeneous CACC system with uncertain dynamics. We consider the type of CACC topology where each vehicle connects only with its nearest preceding vehicle via wireless network. Vehicle dynamics are heterogeneous and uncertain. Based on a new CACC setup, the problem is reduced by regulating an error system with uncertain dynamics for each vehicle. An adaptive optimal control is proposed to find the optimal feedback controller based on data. Vehicle string stability is established by analyzing the position transfer function in the frequency domain. Based on the sum of squares (SOS) theory, the minimal headway time to ensure string stability is numerically solved. Two different systems test the performance.

## II. Heterogeneous CACC and Control Structure

In a CACC system, a string of vehicles drive as a platoon and are connected via a wireless network. Each vehicle follows its predecessor and maintains a safe distance. A local sensor is mounted at the front bumper to detect the relative distance and the relative velocity to its predecessor. For wireless communication, we assume each vehicle receives information only from its nearest preceding vehicle. This type of CACC reduces communication burden compared with other topologies. The leading vehicle transmits its own information to the follower but is not connected to the preceding traffic. It can be a human driver or an ACC-equipped car. Fig. 1 gives a schematic of a vehicle platoon equipped with CACC.

The longitude control of autonomous vehicle is to find a desired acceleration so that the vehicle can follow certain speeds. A low level close-loop ensures tracking of the desired acceleration through the actuation of the throttle/brake system. Let $p_i$, $v_i$, and $a_i$ be position, velocity, and acceleration of the $i$th vehicle. The longitudinal vehicle dynamics is modeled by [9], [10]

$$\begin{aligned} \dot{p}_i &= v_i \\ \dot{v}_i &= a_i \\ \dot{a}_i &= -\frac{1}{\tau_i}a_i + \frac{1}{\tau_i}u_i \end{aligned} \tag{1}$$

where the control input $u_i$ is the desired acceleration and parameter and $\tau_i$ represents the low-level control dynamics. For different vehicles, values of $\tau_i$ may be different. In that case, the platoon is heterogeneous. If every vehicle has the same $\tau_i$, the platoon is called homogeneous. Homogeneous CACC can be seen as a special case of heterogeneous CACC.

The aim of CACC is to pack vehicles as tight as possible but still with safe distances. The spacing policy we adopt here is the constant time headway spacing policy. The desired relative distance from the predecessor's rear bumper to the follower's front bumper is defined by

$$d_i^* = r_i + h_i v_i \tag{2}$$

where $r_i$ is the desired distance at standstill and $h_i$ is the headway time constant. The actual distance is detected by the local sensor and is equal to

$$d_i = p_{i-1} - p_i - l_{i-1}$$

where $p_{i-1}$ is the position of the predecessor and $l_{i-1}$ is the car length.

The difference between actual and desired intervehicle distances defines error

$$\begin{aligned} e_i &= d_i - d_i^* \\ &= p_{i-1} - p_i - h_i v_i - r_i - l_{i-1}. \end{aligned}$$

The control objective is to regulate $e_i$ to zero. The time derivatives of $e_i$ have

$$\dot{e}_i = v_{i-1} - v_i - h_i a_i \tag{3}$$

$$\ddot{e}_i = a_{i-1} - a_i - h_i \dot{a}_i \tag{4}$$

$$\dddot{e}_i = -\frac{1}{\tau_i}\ddot{e}_i + \dot{a}_{i-1} + \frac{1}{\tau_i}a_{i-1} - \frac{h_i}{\tau_i}\dot{u}_i - \frac{1}{\tau_i}u_i. \tag{5}$$

If the last four terms on the right-hand side of (5) are regarded as a new input $u = -(\dot{a}_{i-1} + (1/\tau_i)a_{i-1} - (h_i/\tau_i)\dot{u}_i - (1/\tau_i)u_i)$, the error dynamics becomes a single-input single-output system and the actual vehicle input can be recovered by

$$\dot{u}_i = -\frac{1}{h_i}u_i + \frac{\tau_i}{h_i}\dot{a}_{i-1} + \frac{1}{h_i}a_{i-1} + \frac{\tau_i}{h_i}u.$$

In real applications, the exact values of $\tau_i$ are unknown or uncertain, making it hard to compute the actual input $u_i$. To solve that, we use an estimation $\tau_0$ to replace $\tau_i$ and define a new control signal

$$u_{ai} = -\left(\dot{a}_{i-1} + \frac{1}{\tau_0}a_{i-1} - \frac{h_i}{\tau_0}\dot{u}_i - \frac{1}{\tau_0}u_i\right).$$

After inserting $u_{ai}$ into (5), it becomes

$$\dddot{e}_i = -\frac{1}{\tau_i}\ddot{e}_i - \frac{\tau_0}{\tau_i}u_{ai} + \left(1 - \frac{\tau_0}{\tau_i}\right)\dot{a}_{i-1}.$$

The error dynamics is rewritten as a disturbed linear system

$$\dot{x}_i = A_i x_i + b_i u_{ai} + \mathrm{c}_i \dot{a}_{i-1} \tag{6}$$

where $x_i$ consists of $x_i = [e_i, \dot{e}_i, \ddot{e}_i]^T$, $u_{ai}$ is control input, $\dot{a}_{i-1}$ is disturbance, and dynamics have

$$A_i = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & -\frac{1}{\tau_i} \end{bmatrix}, \quad b_i = \begin{bmatrix} 0 \\ 0 \\ -\frac{\tau_0}{\tau_i} \end{bmatrix}, \quad \mathrm{c}_i = \begin{bmatrix} 0 \\ 0 \\ 1 - \frac{\tau_0}{\tau_i} \end{bmatrix}.$$

The input gain vector and disturbance gain vector satisfy

$$\mathrm{c}_i = l + b_i$$

where $l = [0, 0, 1]^T$. The equality holds for every error dynamics in the platoon. Under $\tau_i > 0$ and $\tau_0 > 0$, $(A_i, b_i)$ is

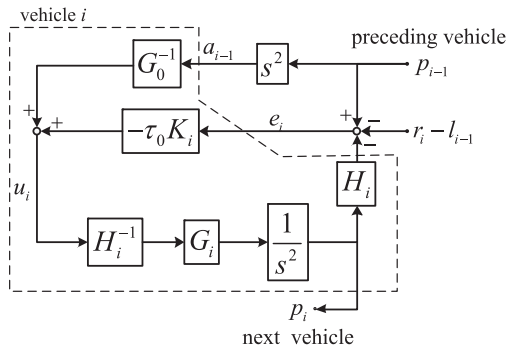


Fig. 2. Control structure of CACC system.

stabilizable. It is possible to find a feedback controller $u_{ai} = -k_i x_i$ such that $(A_i - b_i k_i)$ is Hurwitz. $k_i = [k_{i1}, k_{i2}, k_{i3}]$ is the feedback gain, whose values determine the error regulation performance.

The actual vehicle input now follows control dynamics

$$\dot{u}_i = -\frac{1}{h_i}u_i + \frac{\tau_0}{h_i}\dot{a}_{i-1} + \frac{1}{h_i}a_{i-1} + \frac{\tau_0}{h_i}u_{ai}. \tag{7}$$

Note that two input signals are included. One is the feedback control $u_{ai}$ that uses local data $e_i$. The other is the acceleration of the preceding vehicle $a_{i-1}$, which is transmitted via wireless network and is utilized in a feedforward setting.

Based on Laplace transforms, the CACC system can be described by transfer functions. For ease of notation, capital forms of signals represent their Laplace transforms. The vehicle dynamics has transfer function

$$G_i(s) = \frac{A_i(s)}{U_i(s)} = \frac{1}{\tau_i s + 1}.$$

The spacing policy transfer function $H_i(s)$ following (2) has

$$H_i(s) = \frac{D_i^*(s)}{U_i(s)} = h_i s + 1.$$

The feedback law $K_i(s) = -\frac{U_{ai}(s)}{E_i(s)}$ is equal to

$$K_i(s) = k_{i1} + k_{i2}s + k_{i3}s^2.$$

The transfer function for control dynamics (7) becomes

$$U_i(s) = H_i^{-1}(s)\ (G_0^{-1}(s)A_{i-1}(s) + \tau_0\ U_{ai}(s))$$

where $G_0(s) = (\tau_0\, s + 1)^{-1}$. The control structure for CACC vehicles is depicted in Fig. 2.

Note that CACC mainly works in normal traffic environment. Due to string stability (will be discussed below), vehicles will not accelerate or decelerate violently. Valid control signals within actuator limits are sufficient to track the attenuated acceleration or deceleration propagated along the platoon. Extreme cases like collision avoidance are out of the scope of CACC. In that case, other modules like autonomous emergency braking are expected to take control of the vehicle from CACC.

## III. Adaptive Optimal Control for CACC

Based on the CACC setup proposed in Section II, the problem is reduced by designing a feedback controller $u_{ai} = -k_i x_i$ for each error dynamics. In the field of optimal control, a controller can be evaluated by the cost function

$$J_i = \int_t^\infty \left(x_i^T Q_i x_i + u_{ai}^2\right) d\tau$$

where $Q_i$ is a positive semidefinite symmetric matrix. The optimal control is to find the optimal controller such that the cost function is minimized. Due to the linearity, if no disturbance exists in (6), i.e., $\dot{a}_{i-1} = 0$, the problem is reduced by solving the famous ARE

$$A_i^T P_i^* + P_i^* A_i + Q_i - P_i^* b_i b_i^T P_i^* = 0. \tag{8}$$

The optimal feedback gain $k_i^*$ is obtained by

$$k_i^* = b_i^T P_i^*.$$

As mentioned above, dynamic parameter $\tau_i$ is uncertain, so $A_i$ and $b_i$ are uncertain. An adaptive optimal control is proposed in the following to solve $P_i^*$ and $k_i^*$ without the knowledge of $A_i$ and $b_i$. For simplicity of expression, we omit the vehicle index in $A_i$, $b_i$, and $Q_i$, and use $A$, $b$, and $Q$ for any vehicle in the platoon. The proposed method will be applied to every CACC vehicle (except the leading one) to find their own optimal feedback $k_i^*$.

The new description of error dynamics becomes

$$\dot{x} = Ax + bu_a + \mathrm{c}w$$

where we use $w$ to represent the disturbance $\dot{a}_{i-1}$. With the new notation, the aim of optimal control is to find the optimal $P^*$ satisfying

$$A^T P^* + P^* A + Q - P^* b b^T P^* = 0$$

and the optimal feedback $k^* = b^T P^*$. Before presenting our adaptive optimal control, an important theorem is first introduced.

*Theorem 1 ( [25]):* Let $k^{(0)}$ be any stabilizing feedback gain. For $j = 0, 1, \ldots$, the Lyapunov equation

$$(A - bk^{(j)})^T P^{(j)} + P^{(j)}\ (A - bk^{(j)}) + Q + \ (k^{(j)})^T k^{(j)} = 0 \tag{9}$$

is iteratively solved with the symmetric positive definite solution $P^{(j)}$. The feedback is updated by

$$k^{(j+1)} = b^T P^{(j)}. \tag{10}$$

Then, the following properties hold for all $j$

1) $(A - bk^{(j)})$ is Hurwitz,
2) $P^* \le P^{(j+1)} \le P^{(j)}$,
3) $\lim_{j\to\infty} k^{(j)} = k^*$, $\lim_{j\to\infty} P^{(j)} = P^*$.

Theorem 1 transforms the nonlinear ARE (8) to a sequence of linear Lyapunov equations (9) and updating laws (10). After iteration, $P^{(j)}$ and $k^{(j)}$ uniformly converge to the optimal $P^*$ and $k^*$. Unfortunately, these equations still rely on $A$ and $b$. Inspired by the work of [16], we develop an adaptive optimal control that solves $P^{(j)}$ and $k^{(j+1)}$ based on data. The

difference between [16] and this brief is that we consider a disturbed system (6) while [16] supposes the system includes no disturbance.

Suppose the initial stabilizing $k^{(0)}$ is given. Apply the control $u_a = -k^{(0)}x$ on (6) and produce system data

$$\dot{x} = (A - bk^{(0)})x + \mathrm{c}w. \tag{11}$$

For every $j \geq 0$, on the basis of last iteration result $k^{(j)}$, we seek to solve $P^{(j)}$ and $k^{(j+1)}$ without using $A$ and $b$. If we differentiate $x^T P^{(j)} x$ along the solution of (11) and integrate it between the time interval $[t, t+T]$, we have

$$\begin{aligned} &x(t+T)^T P^{(j)} x(t+T) - x(t)^T P^{(j)} x(t) \\ &\quad = \int_t^{t+T} (x^T (A^T P^{(j)} + P^{(j)} A)x \\ &\qquad - 2x^T P^{(j)} b k^{(0)} x + 2x^T P^{(j)} \mathrm{c}w) d\tau. \end{aligned} \tag{12}$$

After inserting (9) and (10) and using the fact that $\mathrm{c} = l + b$, the following equation becomes

$$\begin{aligned} &x(t+T)^T P^{(j)} x(t+T) - x(t)^T P^{(j)} x(t) \\ &\quad = \int_t^{t+T} (2x^T P^{(j)} l w + 2x^T (k^{(j+1)})^T ((k^{(j)} - k^{(0)})x + w) \\ &\qquad - x^T (Q + (k^{(j)})^T k^{(j)})x) d\tau. \end{aligned} \tag{13}$$

Now the original model-based (9) and (10) are combined together in a model-free fashion. The equation is linear in $P^{(j)}$ and $k^{(j+1)}$. For ease of analysis, define two operators. One transforms the state vector $x = [x_1, x_2, \dots]^T \in \mathbb{R}^n$ into

$$\bar{x} = [x_1^2, x_1\ x_2, x_1\ x_3, \dots, x_2^2, x_2\ x_3, \dots]^T \in \mathbb{R}^{\bar{n}}$$

where $n$ indicates the dimension of $x$ and $\bar{n} = \frac{1}{2}n(n+1)$. The other rearranges the symmetrical matrix $P^{(j)}$ into a vector

$$\hat{P}^{(j)} = [p_{11}, 2p_{12}, 2p_{13}, \dots, p_{22}, 2p_{23}, \dots]^T \in \mathbb{R}^{\bar{n}}$$

where $p_{11}, p_{12}, \dots$ denote matrix elements. Besides, given the vector $\hat{P}^{(j)}$, the symmetrical $P^{(j)}$ is uniquely determined. After inserting $\bar{x}$ and $\hat{P}^{(j)}$ into (13) and moving all items to one side, it becomes

$$\begin{aligned} &\left( (\bar{x}(t+T) - \bar{x}(t))^T - \int_t^{t+T} w l^T \nabla \bar{x} d\tau \right) \hat{P}^{(j)} \\ &\quad - \int_t^{t+T} 2x^T (k^{(j+1)})^T ((k^{(j)} - k^{(0)})x + w) d\tau \\ &\qquad + \int_t^{t+T} x^T (Q + (k^{(j)})^T k^{(j)}) x d\tau = 0 \end{aligned} \tag{14}$$

where $\nabla \bar{x} = [\partial \bar{x}(1)/\partial x, \partial \bar{x}(2)/\partial x, \dots] \in \mathbb{R}^{n \times \bar{n}}$ is the partial derivative of $\bar{x}$ toward $x$. By the Kronecker product representation

$$\begin{aligned} x^T (k^{(j+1)})^T (k^{(j)} - k^{(0)})x &= (x^T \otimes x^T) \\ &\quad \times ((k^{(j)} - k^{(0)})^T \otimes I_n)(k^{(j+1)})^T \\ x^T (Q + (k^{(j)})^T k^{(j)})x &= (x^T \otimes x^T) \\ &\quad \times \mathrm{vec}\,(Q + (k^{(j)})^T k^{(j)}) \end{aligned}$$

where $I_n$ is $n \times n$ unit matrix, and vec () is the vectorizing operator that transforms a matrix into a vector by stacking its elements along column direction. Then (14) becomes

$$\begin{aligned} &\left( (\bar{x}(t+T) - \bar{x}(t))^T - \int_t^{t+T} w l^T \nabla \bar{x} d\tau \right) \hat{P}^{(j)} \\ &\quad - \int_t^{t+T} 2 \left( (x^T \otimes x^T)((k^{(j)} - k^{(0)})^T \otimes I_n) + w x^T \right) d\tau \ (k^{(j+1)})^T \\ &\quad + \int_t^{t+T} (x^T \otimes x^T) d\tau \,\mathrm{vec}\,(Q + (k^{(j)})^T k^{(j)}) = 0. \end{aligned} \tag{15}$$

Equation (15) is linear in $\hat{P}^{(j)}$ and $k^{(j+1)}$, so least-square method can solve it. Suppose along the system evolution in (11), a group of data are collected between intervals $\{[t_k, t_k + T]\}_{k=1}^N$, and the following matrices are defined

$$\delta_{\bar{x}} = \begin{bmatrix} (\bar{x}(t_1 + T) - \bar{x}(t_1))^T \\ \vdots \\ (\bar{x}(t_N + T) - \bar{x}(t_N))^T \end{bmatrix} \in \mathbb{R}^{N \times \bar{n}}$$

$$I_{w\nabla\bar{x}} = \begin{bmatrix} \int_{t_1}^{t_1+T} w l^T \nabla \bar{x} d\tau \\ \vdots \\ \int_{t_N}^{t_N+T} w l^T \nabla \bar{x} d\tau \end{bmatrix} \in \mathbb{R}^{N \times \bar{n}}$$

$$I_{xx} = \begin{bmatrix} \int_{t_1}^{t_1+T} (x^T \otimes x^T) d\tau \\ \vdots \\ \int_{t_N}^{t_N+T} (x^T \otimes x^T) d\tau \end{bmatrix} \in \mathbb{R}^{N \times n^2}$$

$$I_{wx} = \begin{bmatrix} \int_{t_1}^{t_1+T} w x^T d\tau \\ \vdots \\ \int_{t_N}^{t_N+T} w x^T d\tau \end{bmatrix} \in \mathbb{R}^{N \times n}.$$

Following (15), a group of linear equations are formulated and are expressed in the matrix form

$$\begin{aligned} &(\delta_{\bar{x}} - I_{w\nabla\bar{x}})\hat{P}^{(j)} \\ &\quad - 2 (I_{xx} ((k^{(j)} - k^{(0)})^T \otimes I_n) + I_{wx}) (k^{(j+1)})^T \\ &\quad + I_{xx} \mathrm{vec}\,(Q + (k^{(j)})^T k^{(j)}) = 0. \end{aligned}$$

Let

$$\begin{aligned} \Theta &= \left[ \delta_{\bar{x}} - I_{w\nabla\bar{x}}, -2I_{xx} \left( (k^{(j)} - k^{(0)})^T \otimes I_n \right) - 2I_{wx} \right] \\ \Xi &= I_{xx} \mathrm{vec}\,(Q + (k^{(j)})^T k^{(j)}) \end{aligned}$$

and it becomes

$$\Theta \begin{bmatrix} \hat{P}^{(j)} \\ (k^{(j+1)})^T \end{bmatrix} + \Xi = 0. \tag{16}$$

If $\Theta$ has full column rank, $P^{(j)}$ and $k^{(j+1)}$ are uniquely determined by

$$\begin{bmatrix} \hat{P}^{(j)} \\ (k^{(j+1)})^T \end{bmatrix} = -(\Theta^T \Theta)^{-1} \Theta^T \Xi. \tag{17}$$

Based on the new feedback $k^{(j+1)}$, the iterative process continues.

In (17), $\Theta$ needs to have the full column rank at every iteration. Next, lemma gives an alternative way to check the full-rank condition of $\Theta$. The proof is presented in the Appendix.

*Lemma 1:* If the following condition holds for the group of data collected from (11)

$$\text{rank } ([I_{xx}, I_{wx}]) = \frac{1}{2}n(n+1) + n$$

then $\Theta$ has full column rank for all $j \geq 0$.

Because $k^{(j)}$ is a stabilizing feedback, if there is no disturbance, i.e., $w = 0$, the system will eventually rest at equilibrium. In that case, it is hard to meet the full-rank condition. Therefore, the system needs to be persistently excited. Fortunately, in real applications, the platoon is continuously disturbed by the dynamic traffic. The disturbance $w$, i.e., $\dot{a}_{i-1}$, always exists for every vehicle, and the full-rank condition is easily satisfied.

*Theorem 2:* Given an initial stabilizing $k_0$. If the condition of Lemma 1 is satisfied, the sequence of $\{P^{(j)}\}$ and $\{k^{(j+1)}\}$ obtained by iterating (17) converge to the optimal solution $P^*$ and $k^*$.

*Proof:* At the $j$th iteration, under the stabilizing $k^{(j)}$, the solution to (17) is uniquely determined since $\Theta$ has full-column rank. From (14), the solutions $P^{(j)}$ and $k^{(j+1)}$ of (9) and (10) satisfy the linear equation (16). Hence, iterating (17) is equivalent to iterating (9) and (10) under the condition of Lemma 1. By Theorem 1, the convergence is obtained. ■

Before applying our learning method, the initial stabilizing $k_0$ needs to be known. Even though the exact value of $\tau_i$ is not known, the range of this parameter is commonly available in vehicle manuals. Finding a stabilizing $k_0$ for the error dynamics is equivalent to the robust synthesis problem of uncertain linear systems. Some existing literature has presented promising work on this topic [26].

## IV. String Stability

In Section III, a model-free adaptive optimal control is developed for CACC. The optimal feedback controller for individual vehicle is learned based on system data to regulate distance error. But for the safety of the whole platoon, the string stability has to be validated. When multiple vehicles move as a string, any disturbance from the leading vehicle is desired to be attenuated backwards. String stability is defined by the amplification of signals upstream the platoon, which can be quantified by the magnitude of transfer functions between the leading vehicle and its followers

$$SS^*_{\Lambda,i} = \frac{\Lambda_i}{\Lambda_1}, \quad \text{for } i > 1, \ \ \Lambda \in \{P, U, E\}$$

where $P, U,$ and $E$ represent Laplace transforms of position, control, and error signals, respectively. A necessary condition for string stability is

$$|SS^*_{\Lambda,i}\ (j\omega)| \leq 1, \quad \forall\omega, \ \text{for } i > 1, \ \ \Lambda \in \{P, U, E\}.$$

In the literature, a more conservative condition is widely used

$$|SS_{\Lambda,i}\ (j\omega)| \leq 1, \quad \forall\omega, \ \text{for } i > 1, \ \ \Lambda \in \{P, U, E\} \quad (18)$$

where $SS_{\Lambda,i}$ is the transfer function between the $i$th and $(i-1)$th vehicles with respect to position, control, and error signals, $SS_{\Lambda,i} = (\Lambda_i/\Lambda_{i-1})$. Since $SS^*_{\Lambda,i} = \prod_{k=2}^{i} SS_{\Lambda,k}$, if for any $i$ we have $|SS_{\Lambda,i}\ (j\omega)| \leq 1$, then $|SS^*_{\Lambda,i}\ (j\omega)| \leq 1$ is guaranteed.

As mentioned in [9], for heterogeneous CACC, string stability of position signal $\Lambda = P$ is mostly considered. Based on the fact that $A = sV = s^2\ P$, position stability implies velocity and acceleration stability. For ease of analysis, we ignore the standstill distance $r_i$ and the vehicle length $l_{i-1}$. From the control structure in Fig. 2, the position transfer function is described by

$$SS_{P,i}(s) = \frac{P_i(s)}{P_{i-1}(s)} = \frac{s^2\ G_0^{-1} - \tau_0\ K_i}{H_i\left(s^2\ G_i^{-1} - \tau_0\ K_i\right)}.$$

After inserting $s = j\omega$ from (18), we infer that a sufficient condition for string stability is the polynomial

$$\begin{aligned}\frac{\tau_i^2}{\tau_0^2}h_i^2\omega^6 &+ \left[-1 + \frac{\tau_i^2}{\tau_0^2} + 2\frac{\tau_i}{\tau_0}k_{i,2}h_i^2 + \left(\frac{1}{\tau_0} - k_{i,3}\right)^2 h_i^2\right]\omega^4 \\ &+ \left[-2k_{i,2} + 2\frac{\tau_i}{\tau_0}k_{i,2} + k_{i,2}^2h_i^2 + 2\left(\frac{1}{\tau_0} - k_{i,3}\right)k_{i,1}h_i^2\right]\omega^2 \\ &+ k_{i,1}^2h_i^2 \end{aligned} \quad (19)$$

being nonnegative $\forall\omega$.

Reviewing (19), it is a six-order polynomial in $\omega$. It is desired to find the minimal headway value $h_{i,\min} > 0$ such that the polynomial is nonnegative $\forall\omega$. Then, we can select $h_i > h_{i,\min}$ to ensure string stability. Unfortunately, determining positivity of a polynomial is NP-hard. With the aid of polynomial theory, the problem can be approximately solved by relaxing the nonnegative constraint to the SOS constraint [27].

A multivariate polynomial $f(x)$ is called SOS if there exist polynomials $g_1(x), \ldots, g_m(x)$ such that

$$f(x) = \sum_{k=1}^{m} g_k^2(x).$$

An SOS polynomial is naturally nonnegative, but the converse is not true. The advantage of SOS constraint is that the decomposition of a SOS polynomial is equivalent to the semidefinite programming and many useful toolboxes [28], [29] have been fully developed to solve that.

Based on the SOS theory, if the dynamic parameter $\tau_i$ and the estimation $\tau_0$ are given and the feedback gain $k_i$ is known, the minimal headway $h_{i,\min}$ to make string stable is approximately obtained by solving the optimization problem

$$\begin{aligned} &\min\ h_i^2 \\ &\text{s.t. (19) is SOS.} \end{aligned} \quad (20)$$

Fig. 3 illustrates an example of $h_{i,\min}$ at different $(\tau_i, \tau_0)$. The feedback is specified to $k_i = [-0.5, -0.5, 0]$. From the figure, when $\tau_0 \geq \tau_i$, $h_{i,\min}$ is very close to zero. The platoon allows small intervehicle distances, which contribute to high-traffic throughput. Otherwise when $\tau_0 < \tau_i$, $h_{i,\min}$ increases dramatically. Vehicles have to keep away from each other for safety, and road capacity is reduced. For this $k_i$, we would like to select a little bit larger $\tau_0$ to overestimate the uncertain $\tau_i$. Note that for some other $k_i$, a smaller $\tau_0$ less than $\tau_i$ may ensure a small $h_{i,\min}$.

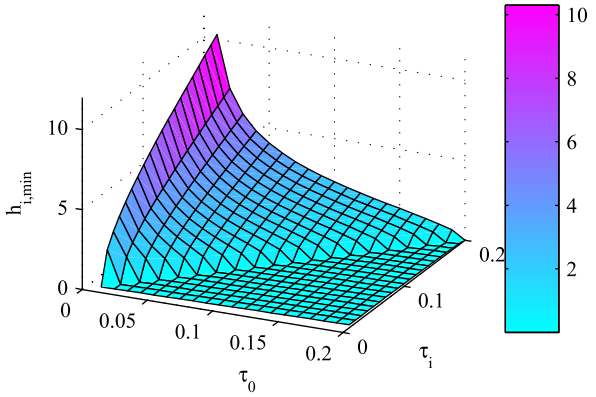


Fig. 3. Minimal values of $h_{i,\min}$ at different $(\tau_i, \tau_0)$ under controller $K_i = -0.5 - 0.5s$.

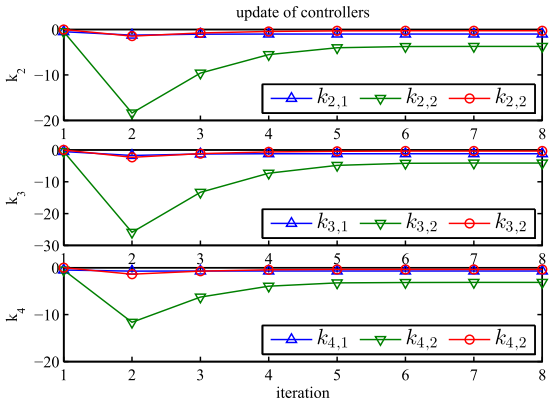


Fig. 4. Iterations of error feedback gains.

The selection of $\tau_0$ not only affects the value of $h_{i,\min}$ but also affects the error dynamics. When the difference between $\tau_0$ and $\tau_i$ becomes larger, the value of disturbance gain $c_i$ in (6) increases, and the error dynamics is more sensitive to the disturbance $\dot{a}_{i-1}$. If $\tau_0$ is exactly equal to $\tau_i$, the error dynamics is not disturbed by $\dot{a}_{i-1}$. Based on that, the difference between $\tau_0$ and $\tau_i$ should be as small as possible. In practice, we can adjust the low-level acceleration controllers of throttle and brake systems to make different $\tau_i$ close to each other.

## V. Experiments

### A. Numerical Experiments

To verify the effectiveness of our method, we first set up a numerical experiment with four vehicles based on (1). The dynamics of four vehicles choose

$$\tau_1 = 0.1, \tau_2 = 0.08, \tau_3 = 0.09, \tau_4 = 0.12.$$

These values are not known to our algorithm. To make string stable, the dynamics are overestimated by $\tau_0 = 0.15$. The initial stabilizing error feedbacks are uniformly set to $k_i^{(0)} = [-0.5, -0.5, 0]$, $i = 2, 3, 4$. According to Fig. 3, for every vehicle, we select the same headway time $h_i = 0.5$. The standstills are all set to $r_i = 2$.

For the optimal criteria, the error costs select $Q_2 = \text{diag}([1, 0, 0])$, $Q_3 = \text{diag}([1.5, 0, 0])$, and $Q_4 = \text{diag}([0.5, 0, 0])$. After inserting into the ARE (8), the optimal feedbacks are solved

$$\begin{aligned} k_2^* &= [-1.0000 \quad -3.7306 \quad -0.2921] \\ k_3^* &= [-1.2247 \quad -4.1498 \quad -0.3636] \\ k_4^* &= [-0.7071 \quad -3.1542 \quad -0.3683]. \end{aligned}$$

The whole platoon is initialized at rest and the distances between vehicles are set to standstill distances. We first use the initial feedback $k_i^{(0)}$ to collect system data. To ensure the full-rank condition of Theorem 2, the leading vehicle follows a noised control signal. Once the condition is satisfied, the adaptive optimal control starts to learn the optimal controllers based on collected data. The iteration process is depicted in Fig. 4. On a computer with Intel Core i5-4590T CPU and 4GB memory, the iterations converge within 10 $\mu$s. The final converged values are

$$\begin{aligned} k_2^\infty &= [-0.9999 \quad -3.7308 \quad -0.2921] \\ k_3^\infty &= [-1.2248 \quad -4.1496 \quad -0.3636] \\ k_4^\infty &= [-0.7071 \quad -3.1542 \quad -0.3683] \end{aligned}$$

which are nearly equal to the exact solutions. Then, the controllers are replaced by the converged feedbacks to continue driving.

Fig. 5 shows the error trajectories during the whole process. Vertical lines on the time axis indicate the instant when enough data are collected to meet the full-rank condition. Before the instant, initial $k_i^{(0)}$ controls the vehicle. After that, converged $k_i^\infty$ replaces $k_i^{(0)}$. For different vehicles, their error dynamics are different, so the time when the full-rank condition is satisfied is different. In the plot, error states are regulated around zero but still with oscillations. It is because of the continuously noised control input from the leading vehicle. The disturbance is propagated backwards, but the magnitudes are attenuated. For details, we plot the trajectories of velocities and accelerations of all vehicles in Figs. 6 and 7. In these plots, signal amplitudes are reduced along three vehicles. In fact, with the converged feedbacks, after solving the optimization problem (20), the minimal values $h_{i,\min}$ to ensure string stability are $h_{2,\min} = 0.10645$, $h_{3,\min} = 0.09790$, and $h_{4,\min} = 0.07202$. All of them are far less than our selection $h_i = 0.5$, so the platoon is theoretically string stable.

### B. TORCS Experiment

In order to validate the performance of CACC optimal control in a more complex environment, we set up a test experiment in TORCS. TORCS is an open source 3-D car racing simulator. It provides realistic experience with powerful physics engines and sophisticated 3-D graphics.[1] In the experiment, the platoon is composed of 16 vehicles with different dynamics. All vehicles use the same headway time $h_i = 0.5$ and the same standstill $r_i = 2$. In order to learn the optimal CACC controllers, the error costs are set equal to $Q_i = \text{diag}([1, 0, 0])$.

At the beginning, each vehicle uses an initial controller $k_i^{(0)} = [-0.2, -0.2, 0]$ to drive. Our method collects driving

[1] http://torcs.sourceforge.net/

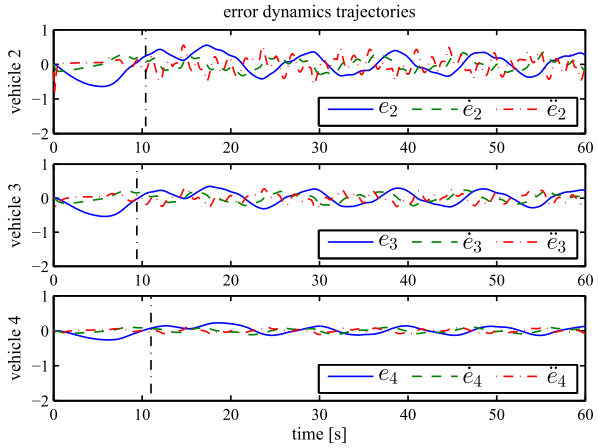


Fig. 5. Evolution of error dynamics.

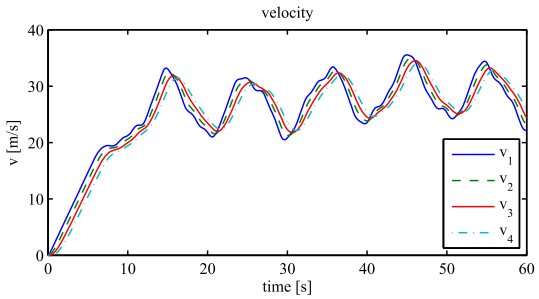


Fig. 6. Trajectories of vehicle velocities.

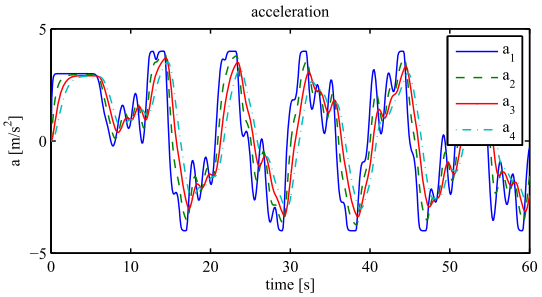


Fig. 7. Trajectories of vehicle accelerations.

data and learns the optimal CACC controllers for each vehicle. After convergence, the learned controllers take control of vehicles. The experiment results are plotted in Fig. 8(a). The $x$-coordinate is the time axis, and the $y$-coordinate is the intervehicle distance errors. Different vehicles are marked with different colors. "*Accel*," "*brake*," "*30*," and "*10*" in the plot indicate the leading vehicle is accelerating, braking, maintaining at 30 m/s and maintaining at 10 m/s, respectively. The vertical lines on the time axis indicate the time when the leading vehicle switches its behavior. For comparison, we repeat the experiment but with different controllers. The first comparison uses $k_i^{(0)}$, i.e., a nonoptimal CACC controller, and the results are plotted in Fig. 8(b). From the plot, the nonoptimal controller is not able to stabilize distance errors efficiently. In other words, after updating the controllers by our method, the CACC vehicles follow their predecessors more accurately. It helps to improve traffic safety. In the second comparison, we use the learned optimal controllers but turn off the intervehicle communication. Preceding accelerations

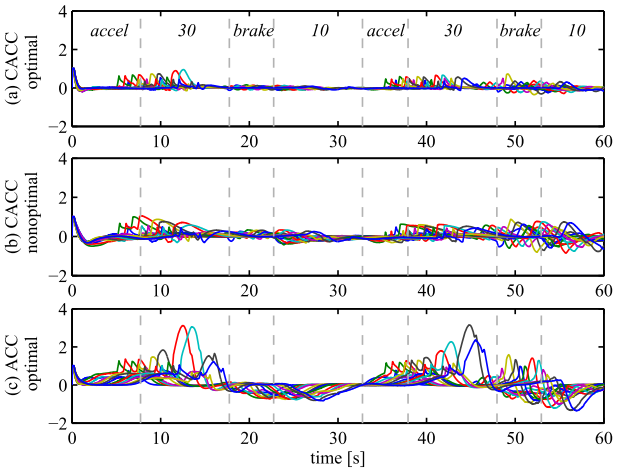


Fig. 8. Distance errors in TORCS experiments with different control.

are not transmitted to followers. The experiment results are depicted in Fig. 8(c). Brake or acceleration behaviors are transmitted quite slowly to the tail of the platoon. Distance errors are amplified along vehicles. In contrast to CACC, ACC fails in maintaining string stability. To solve the problem, a large headway time has to be adopted for ACC with the sacrifice of traffic throughput.

## VI. Conclusion

In this brief, we study the heterogeneous CACC platoon with uncertain vehicle dynamics. Wireless communicated intervehicle information is combined with local measurement. After introducing an estimate dynamic parameter, the new CACC setup reduces the problem to regulate an error system with uncertain dynamics for each vehicle. A model-free adaptive optimal control is developed to find the optimal error feedback. To ensure string stability, the position transfer function between adjacent vehicles is analyzed in the frequency domain. By the SOS theory, the minimum headway for string stability is numerically solved. Experiments on a complex system show the superiority of CACC over ACC in the aspect of maintaining string stability and improving traffic throughput. The optimal control makes the CACC platoon stabilize distance errors more efficiently.

## Appendix
## Proof of Lemma 1

For every $j$, the full-rank condition of $\Theta$ is equivalent to the linear equation

$$\Theta \begin{bmatrix} \hat{Y} \\ z \end{bmatrix} = 0 \tag{21}$$

having a trivial zero solution $\hat{Y} = 0 \in \mathbb{R}^{\bar{n}}$ and $z = 0 \in \mathbb{R}^n$. From $\hat{Y}$, a symmetrical matrix $Y \in \mathbb{R}^{n\times n}$ is uniquely determined. The first row of (21) becomes

$$x^T(t_1+T)Yx(t_1+T) - x^T(t_1)Yx(t_1) - \int_{t_1}^{t_1+T} 2x^T Yl\omega d\tau - \int_{t_1}^{t_1+T} 2(x^T z\ (k^{(j)} - k^{(0)})x + x^T z w)d\tau = 0. \tag{22}$$

Similar to (12), we have

$$x^T(t_1+T)Yx(t_1+T)-x^T(t_1)Yx(t_1) = \int_{t_1}^{t_1+T}(x^T\ (YA+A^TY)x-2x^TYbk^{(0)}x +2x^TY\ (l+b)w)d\tau.$$

After inserting into (22), it becomes

$$\int_{t_1}^{t_1+T}(x^TFx+2wx^Tg)d\tau=0$$

where $F=YA+A^TY-2zk^{(j)}-2(Yb-z)k^{(0)}$ and $g=Yb-z$. The equation holds for every time index $t_k$. Define a vector $\hat{F}\in\mathbb{R}^{\bar{n}}$ from $F$. The linear equation (21) is rewritten as

$$[I_{\bar{x}},2I_{wx}]\begin{bmatrix}\hat{F}\\ g\end{bmatrix}=0 \tag{23}$$

where $I_{\bar{x}}$ is defined as

$$I_{\bar{x}}=\left[\int_{t_1}^{t_1+T}\bar{x}d\tau,\int_{t_2}^{t_2+T}\bar{x}d\tau,\ldots,\int_{t_N}^{t_N+T}\bar{x}d\tau\right]^T.$$

From the requirement of Lemma 1 it is inferred that $[I_{\bar{x}},2I_{wx}]$ has full column rank, so (23) has the trivial zero solution. Based on the fact that $\hat{F}$ and $F$ uniquely determine each other, we have

$$YA+A^TY-2zk^{(j)}-2(Yb-z)k^{(0)}=0 \tag{24}$$

$$Yb-z=0. \tag{25}$$

After inserting (25) into (24), it becomes

$$Y\ (A-bk^{(j)})+\ (A-bk^{(j)})^TY=0.$$

Since $(A-bk^{(j)})$ is Hurwitz for all $j$, we have $Y=0$. By (25), $z=0$.

From the above analysis, (21) has a trivial zero solution. Therefore, $\Theta$ has full column rank.

## REFERENCES


[1] L. Xiao and F. Gao, "A comprehensive review of the development of adaptive cruise control systems," *Veh. Syst. Dyn.*, vol. 48, no. 10, pp. 1167–1192, 2010.

[2] S. Hassan Hosseinnia, I. Tejado, V. Milanes, J. Villagra, and B. M. Vinagre, "Experimental application of hybrid fractional-order adaptive cruise control at low speed," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 6, pp. 2329–2336, Nov. 2014.

[3] D. Zhao, Z. Hu, Z. Xia, C. Alippi, Y. Zhu, and D. Wang, "Full-range adaptive cruise control based on supervised adaptive dynamic programming," *Neurocomputing*, vol. 125, pp. 57–67, Feb. 2014.

[4] D. Zhao, Z. Xia, and Q. Zhang, "Model-free optimal control based intelligent cruise control with hardware-in-the-loop demonstration [research frontier]," *IEEE Comput. Intell. Mag.*, vol. 12, no. 2, pp. 56–69, May 2017.

[5] J. Ploeg, A. F. Serrarens, and G. J. Heijenk, "Connect & drive: Design and evaluation of cooperative adaptive cruise control for congestion reduction," *J. Modern Transp.*, vol. 19, no. 3, pp. 207–213, 2011.

[6] K. C. Dey *et al.*, "A review of communication, driver characteristics, and controls aspects of cooperative adaptive cruise control (CACC)," *IEEE Trans. Intell. Transp. Syst.*, vol. 17, no. 2, pp. 491–509, Feb. 2016.

[7] 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.

[8] W. J. Schakel, B. van Arem, and B. D. Netten, "Effects of cooperative adaptive cruise control on traffic flow stability," in *Proc. 13th Int. IEEE Conf. Intell. Transp. Syst.*, Sep. 2010, pp. 759–764.

[9] 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.

[10] S. Öncü, J. Ploeg, N. van de Wouw, and H. Nijmeijer, "Cooperative adaptive cruise control: Network-aware analysis of string stability," *IEEE Trans. Intell. Transp. Syst.*, vol. 15, no. 4, pp. 1527–1537, Aug. 2014.

[11] 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.

[12] W. Levine and M. Athans, "On the optimal error regulation of a string of moving vehicles," *IEEE Trans. Autom. Control*, vol. AC-11, no. 3, pp. 355–361, Jul. 1966.

[13] 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.

[14] F. L. Lewis and D. Liu, *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control*, vol. 17. Hoboken, NJ, USA: Wiley, 2013.

[15] Y. Zhu and D. Zhao, "Comprehensive comparison of online ADP algorithms for continuous-time optimal control," *Artif. Intell. Rev.*, vol. 49, no. 4, pp. 531–547, 2017, doi: 10.1007/s10462-017-9548-4.

[16] Y. Jiang and Z.-P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, no. 10, pp. 2699–2704, 2012.

[17] Y. Jiang and Z.-P. Jiang, "Robust adaptive dynamic programming and feedback stabilization of nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 5, pp. 882–893, May 2014.

[18] Y. Zhu, D. Zhao, and X. Li, "Iterative adaptive dynamic programming for solving unknown nonlinear zero-sum game based on online data," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 714–725, Mar. 2017.

[19] Y. Zhu, D. Zhao, X. Yang, and Q. Zhang, "Policy iteration for $H_\infty$ optimal control of polynomial nonlinear systems via sum of squares programming," *IEEE Trans. Cybern.*, vol. 48, pp. 500–509, Feb. 2017, doi: 10.1109/TCYB.2016.2643687.

[20] Y. Zhu, D. Zhao, H. He, and J. Ji, "Event-triggered optimal control for partially unknown constrained-input systems via adaptive dynamic programming," *IEEE Trans. Ind. Electron.*, vol. 64, no. 5, pp. 4101–4109, May 2017.

[21] J. Wang, X. Xu, D. Liu, Z. Sun, and Q. Chen, "Self-learning cruise control using kernel-based least squares policy iteration," *IEEE Trans. Control Syst. Technol.*, vol. 22, no. 3, pp. 1078–1087, May 2014.

[22] W. Gao, Z.-P. 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.

[23] L. Zhang, J. Sun, and G. Orosz, "Hierarchical design of connected cruise control in the presence of information delays and uncertain vehicle dynamics," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 1, pp. 139–150, Jan. 2018, doi: 10.1109/TCST.2017.2664721.

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

[25] D. L. Kleinman, "On an iterative technique for Riccati equation computations," *IEEE Trans. Autom. Control*, vol. AC-13, no. 1, pp. 114–115, Feb. 1968.

[26] W.-C. Huang, H.-F. Sun, and J.-P. Zeng, "Robust control synthesis of polynomial nonlinear systems using sum of squares technique," *Acta Autom. Sinica*, vol. 39, no. 6, pp. 799–805, 2013.

[27] P. A. Parrilo, "Structured semidefinite programs and semialgebraic geometry methods in robustness and optimization," Ph.D. dissertation, California Inst. Technol., Pasadena, CA, USA, 2000.

[28] A. Papachristodoulou, J. Anderson, G. Valmorbida, S. Prajna, P. Seiler, and P. A. Parrilo. (Oct. 2013). "SOSTOOLS version 3.00 sum of squares optimization toolbox for MATLAB." [Online]. Available: https://arxiv.org/abs/1310.4716

[29] D. Henrion and J.-B. Lasserre, "GloptiPoly: Global optimization over polynomials with MATLAB and SeDuMi," *ACM Trans. Math. Softw.*, vol. 29, no. 2, pp. 165–194, 2003.