IEEE Control Systems Letters paper presented at
2023 American Control Conference (ACC)
San Diego, CA, USA. May 31 - June 2, 2023

Learning Adaptive Optimal Controllers for Linear Time-Delay Systems*

Leilei Cui[1], Bo Pang[1] and Zhong-Ping Jiang[1], *Fellow, IEEE*

***Abstract*— This paper studies the learning-based optimal control for a class of infinite-dimensional linear time-delay systems. The aim is to fill the gap of adaptive dynamic programming (ADP) where adaptive optimal control of infinite-dimensional systems is not addressed. A key strategy is to combine the classical model-based linear quadratic (LQ) optimal control of time-delay systems with the state-of-art reinforcement learning (RL) technique. Both the model-based and data-driven policy iteration (PI) approaches are proposed to solve the corresponding algebraic Riccati equation (ARE) with guaranteed convergence. The proposed PI algorithm can be considered as a generalization of ADP to infinite-dimensional time-delay systems. The efficiency of the proposed algorithm is demonstrated by the practical application arising from autonomous driving in mixed traffic environments, where human drivers' reaction delay is considered.**

I. Introduction

By continuously interacting with environment and receiving rewards, RL is able to iteratively maximize the cumulative rewards (or minimize the costs) and learn an optimal control policy from scratch. Conventionally, RL algorithms are developed in the setting of Markov decision processes[1], where the dynamics of the environment is discrete-time, and the action and state spaces are finite or countable. Furthermore, the stability issue is often neglected by conventional RL. In reality, most physical systems are continuous-time and depicted by ordinary differential equations (ODEs), stochastic differential equations (SDEs), or delay differential equations (DDEs), of which the state and action spaces are continuous and infinite. Moreover, the stability of the system with the learned control policy is critical to ensure the safety, for example, autonomous vehicles and robots. These issues that are overlooked by the conventional RL algorithms invoked the development of ADP technique.

By integrating stability in classical control theory with the state-of-art RL technique, ADP is capable of learning a sequence of stabilizing control policies for both discrete and continuous-time systems, and these control polices converge to the optimal solution as the learning iteration tends to infinity [2], [3], [4]. So far, ADP techniques are developed for stabilization and output regulation of various linear/nonlinear/periodic systems [3], [5], [6], and are successfully applied to wheel-legged robots [7] and vehicle control [8]. For systems involving human-machine interaction and network control, the time delay induced by human reaction lags [9] and network communication [10] may degrade the system performance or even destabilize the system. Therefore, time-delay effects should be considered for the controller design. For time-delay systems, most existing ADP techniques are devoted to discrete-time systems [11], [12], [13], [14]. Due to the finite dimensionality of discrete-time time-delay systems, one can transfer the time-delay system to a delay-free system with an augmented state. However, these methods are not applicable to continuous-time time-delay systems with infinite dimensionality. For the continuous-time time-delay systems, one has to discretize the DDEs into ODEs with an augmented state, resulting in an approximate model [15]. The authors of [16] adopted ADP technique to design the learning-based controller for time-delay systems, but the resulting control policy is not optimal since the corresponding AREs are not solved by the proposed approach. Hence, the adaptive optimal control for continuous-time time-delay systems is still an open problem, that is worth further investigation.

For a delay-free linear system, the solution of the LQ optimal control is related to a matrix-valued ARE. In comparison, since the linear time-delay systems are infinite-dimensional, the corresponding ARE is a set of nonlinear partial differential equations (PDEs). Furthermore, the value function and control law for time-delay systems are functionals of a segment of the state trajectory. These facts hinder the development of adaptive optimal control for linear time-delay systems. In this paper, we generalize the celebrated Kleinman's PI algorithm [17] to time-delay systems. Given an initial admissible controller, the proposed model-based PI algorithm can approximate the optimal solution of the original nonlinear PDEs by iteratively solving a series of linear PDEs. Furthermore, it is theoretically demonstrated that the value functionals obtained at each iteration are monotonically decreasing, and both the value functional and control law converge to the optimal values as iteration tend to infinity. By combining the model-based PI algorithm with the state-of-art RL technique, a data-driven PI algorithm is proposed, which approximates the value functional and control law at each iteration using only input-state trajectories of the system. The proposed data-driven PI algorithm is applied to the connected and autonomous vehicles (CAVs) in mixed traffic environments to attenuate the stop-and-go waves, where human drivers' reaction delay influences the traffic flow.

Notations: In this paper, $\mathbb{R}_+$ denotes the set of nonnegative real numbers, and $\mathbb{N}_+$ denotes the set of positive integers. $|\cdot|$ denotes the Euclidean norm of a vector or Frobenius norm of a matrix. $\|\cdot\|_\infty$ denotes the supremum norm of a function.

*This work is supported partly by the National Science Foundation under Grants EPCN-1903781 and DMS-2009644.

[1]L. Cui, B. Pang, and Z. P. Jiang are with the Control and Networks Lab, Department of Electrical and Computer Engineering, Tandon School of Engineering, New York University, Brooklyn, NY 11201, USA (e-mail: l.cui@nyu.edu; bo.pang@nyu.edu; zjiang@nyu.edu).

$\frac{\mathrm{d}f}{\mathrm{d}\theta}(\cdot)$ denotes the derivative of the function f. $\mathscr{C}^0(X,Y)$ denotes the class of continuous functions from the linear space X to the linear space Y. $\mathscr{AC}([-\tau,0],\mathbb{R}^n)$ denotes the class of absolutely continuous functions. $\oplus$ denotes the direct sum. $L_i([-\tau,0],\mathbb{R}^n)$ denotes the space of measurable functions for which the ith power of the Euclidean norm is Lebesgue integrable, $\mathscr{M}_2=\mathbb{R}^n\oplus L_2([-\tau,0],\mathbb{R}^n)$, and $\mathscr{D}=\left\{[r^\top,f^\top(\cdot)]^\top\in\mathscr{M}_2: f\in\mathscr{AC},\frac{\mathrm{d}f}{\mathrm{d}\theta}(\cdot)\in L_2, \text{ and } f(0)=r\right\}$. $\mathscr{L}(X)$ and $\mathscr{L}(X,Y)$ denote the class of continuous bounded linear operators from X to X and from X to Y respectively. $\otimes$ denotes the Kronecker product. $\mathrm{vec}(A)=\left[a_1^\top,a_2^\top,...,a_n^\top\right]^\top$, where $A\in\mathbb{R}^{n\times n}$ and a_i is the ith column of A. For a symmetric matrix $P\in\mathbb{R}^{n\times n}$, $\mathrm{vecs}(P)=[p_{11},2p_{12},...,2p_{1n},p_{22},2p_{23},...,2p_{(n-1)n},p_{nn}]^\top$, $\mathrm{vecu}(P)=[2p_{12},...,2p_{1n},2p_{23},...,2p_{(n-1)n}]^\top$, and $\mathrm{diag}(P)=[p_{11},p_{22},...,p_{nn}]^\top$. For two arbitrary vectors $\nu,\mu\in\mathbb{R}^n$, $\mathrm{vecd}(\nu,\mu)=[\nu_1\mu_1,\cdots,\nu_n\mu_n]^\top$, $\mathrm{vecv}(\nu)=[\nu_1^2,\nu_1\nu_2,...,\nu_1\nu_n,\nu_2^2,...,\nu_{n-1}\nu_n,\nu_n^2]^\top$, $\mathrm{vecp}(\nu,\mu)=[\nu_1\mu_2,...,\nu_1\mu_n,\nu_2\mu_3,...,\nu_{n-1}\mu_n]^\top$. $A^\dagger$ denotes the Moore-Penrose inverse of matrix A. $[a]_i$ is the ith entry of the vector a.

II. Problem Formulation and Preliminaries

A. Problem Formulation

Consider a linear time-delay system

$$\dot{x}(t)=Ax(t)+A_dx(t-\tau)+Bu(t), \tag{1}$$

where $\tau\in\mathbb{R}_+$ denotes the delay of the system, $x(t)\in\mathbb{R}^n$, and $u(t)\in\mathbb{R}^m$. A, $A_d\in\mathbb{R}^{n\times n}$ and $B\in\mathbb{R}^{n\times m}$ are unknown constant matrices. Let $x_t(\theta)=x(t+\theta)$, $\forall\theta\in[-\tau,0]$ denote a segment of the state trajectory in the interval $[t-\tau,t]$. Due to the infinite dimensionality of system (1), the state of the system is $z(t)=[x^\top(t),x_t^\top(\cdot)]^\top\in\mathscr{M}_2$. Define the linear operators $\mathbf{A}\in\mathscr{L}(\mathscr{M}_2),\mathbf{B}\in\mathscr{L}(\mathbb{R}^m,\mathscr{M}_2)$ as $\mathbf{A}z(t)=\begin{bmatrix}Ax(t)+A_dx_t(-\tau)\\ \frac{\mathrm{d}x_t}{\mathrm{d}\theta}(\cdot)\end{bmatrix}$ and $\mathbf{B}u(t)=\begin{bmatrix}Bu(t)\\0\end{bmatrix}$. Then, according to [18, Theorem 2.4.6], (1) can be rewritten as

$$\dot{z}(t)=\mathbf{A}z(t)+\mathbf{B}u(t), \tag{2}$$

with the domain of $\mathbf{A}$ given by $\mathscr{D}$. Let $z_0=[x^\top(0),x_0^\top(\cdot)]^\top\in\mathscr{D}$ denote the initial state of the system (2). The performance index of (1) is

$$J(x_0,u)=\int_0^\infty x(t)^\top Qx(t)+u(t)^\top Ru(t)\mathrm{d}t \tag{3}$$

where $R^\top=R>0$, $Q^\top=Q\geq0$.

Definition 1. *For system (1), a control policy $u_c(x_t):\mathscr{D}\to\mathbb{R}^m$ is called admissible with respect to (3), if the linear time-delay system (1) with $u=u_c(x_t)$ is globally asymptotically stable (GAS) at the origin [19, Definition 1.1], and the performance index (3) is finite for all $z_0\in\mathscr{D}$.*

Assumption 1. *The system (1) with the output $y(t)=Q^{\frac{1}{2}}x(t)$ is exponentially stablizable and detectable, which are defined in [18, Definition 5.2.1] and can be checked by [18, Theorem 5.2.12].*

Remark 1. *Assumption 1 is a standard prerequisite for LQ optimal control of system (1) to ensure the existence of a stabilizing solution [18], [20].*

Given the aforementioned assumption, the problem to be studied in this paper can be formulated as follows.

Problem 1 (PI-based ADP). *Given an initial admissible controller $u_1(x_t)=-K_{0,1}x(t)-\int_{-\tau}^0K_{1,1}(\theta)x_t(\theta)\mathrm{d}\theta$, and without knowing the dynamics of the system (1), design a PI-based ADP algorithm to find approximations of the optimal controller which can minimize (3) using only the input-state data measured along the trajectories of the system.*

B. Optimality and Stability

For a linear system without time delay, i.e. $A_d=0$ in (1), one can calculate the optimal controller by solving the ARE as discovered by Kalman [21]. Correspondingly, for the linear time-delay system (1), the sufficient condition for a model-based solution to the optimal control problem is stated as follows.

Lemma 1 ([22], [23])**.** *For system (1) under Assumption 1,*

$$u^*(x_t)=-\underbrace{R^{-1}B^\top P_0^*}_{K_0^*}x(t)-\int_{-\tau}^0\underbrace{R^{-1}B^\top P_1^*(\theta)}_{K_1^*(\theta)}x_t(\theta)\mathrm{d}\theta \tag{4}$$

is the optimal controller minimizing the cost (3), and the corresponding minimal performance index is

$$\begin{aligned}V^*(x_0)=&x^\top(0)P_0^*x(0)+2x^\top(0)\int_{-\tau}^0P_1^*(\theta)x_0(\theta)\mathrm{d}\theta\\&+\int_{-\tau}^0\int_{-\tau}^0x_0^\top(\xi)P_2^*(\xi,\theta)x_0(\theta)\mathrm{d}\xi\mathrm{d}\theta,\end{aligned} \tag{5}$$

where $P_0^*=P_0^{*\top}\geq0$, $P_1^*(\theta)$, and $P_2^{*\top}(\theta,\xi)=P_2^*(\xi,\theta)$ for $\theta,\xi\in[-\tau,0]$ are the unique solution to the following PDEs

$$\begin{aligned}&A^\top P_0^*+P_0^*A-P_0^*BR^{-1}B^\top P_0^*+P_1^*(0)+P_1^{*\top}(0)+Q=0,\\&\frac{\mathrm{d}P_1^*(\theta)}{\mathrm{d}\theta}=(A^\top-P_0^*BR^{-1}B^\top)P_1^*(\theta)+P_2^*(0,\theta),\\&\partial_\xi P_2^*(\xi,\theta)+\partial_\theta P_2^*(\xi,\theta)=-P_1^{*\top}(\xi)BR^{-1}B^\top P_1^*(\theta),\\&P_1^*(-\tau)=P_0^*A_d,\qquad P_2^*(-\tau,\theta)=A_d^\top P_1^*(\theta).\end{aligned} \tag{6}$$

According to [18, Theorem 6.2.7], system (1) in closed-loop with u^* in (4) is exponentially stable at the origin.

III. Model-Based Policy Iteration

According to Lemma 1, if (6) can be solved, the optimal controller is obtained. However, due to the non-linearity with respect to P_0^*, P_1^* and P_2^*, it is difficult to solve (6) directly. Therefore, the model-based PI algorithm is proposed to simplify the process of solving (6).

Given an admissible controller $u_1(x_t)=-K_{0,1}x(t)-\int_{-\tau}^0K_{1,1}(\theta)x_t(\theta)\mathrm{d}\theta$, the model-based PI algorithm for the system (1) is proposed as follows.

1) Policy Evaluation: For $i\in\mathbb{N}_+$, and $\xi,\theta\in[-\tau,0]$, calculate $P_{0,i}=P_{0,i}^\top\geq0$, $P_{1,i}(\theta)$, and $P_{2,i}^\top(\theta,\xi)=P_{2,i}(\xi,\theta)$

by solving the following PDEs

$$\begin{aligned}
&A_i^\top P_{0,i} + P_{0,i}A_i + Q_i + P_{1,i}(0) + P_{1,i}^\top(0) = 0,\\
&\frac{\mathrm{d}P_{1,i}(\theta)}{\mathrm{d}\theta} = A_i^\top P_{1,i}(\theta) - P_{0,i}BK_{1,i}(\theta) + K_{0,i}^\top RK_{1,i}(\theta) + P_{2,i}(0,\theta),\\
&\partial_\xi P_{2,i}(\xi,\theta) + \partial_\theta P_{2,i}(\xi,\theta) = K_{1,i}^\top(\xi)RK_{1,i}(\theta) - 2K_{1,i}^\top(\xi)B^\top P_{1,i}(\theta),\\
&P_{1,i}(-\tau) = P_{0,i}A_d, \qquad P_{2,i}(-\tau,\theta) = A_d^\top P_{1,i}(\theta),
\end{aligned} \tag{7}$$

where $A_i = (A - BK_{0,i})$ and $Q_i = Q + K_{0,i}^\top RK_{0,i}$.

2) Policy Improvement: Update the policy u_{i+1} by

$$u_{i+1}(x_t) = -\underbrace{R^{-1}B^\top P_{0,i}}_{K_{0,i+1}}x(t) - \int_{-\tau}^{0}\underbrace{R^{-1}B^\top P_{1,i}(\theta)}_{K_{1,i+1}(\theta)}x_t(\theta)\mathrm{d}\theta. \tag{8}$$

The policy evaluation step calculates the value functional $V_i(x_0) = J(x_0,u_i)$, which is expressed as

$$\begin{aligned}
V_i(x_t) = {}& x^\top(t)P_{0,i}x(t) + 2x^\top(t)\int_{-\tau}^{0}P_{1,i}(\theta)x_t(\theta)\mathrm{d}\theta\\
&+ \int_{-\tau}^{0}\int_{-\tau}^{0}x_t^\top(\xi)P_{2,i}(\xi,\theta)x_t(\theta)\mathrm{d}\xi\mathrm{d}\theta.
\end{aligned} \tag{9}$$

By policy improvement, the value functional is monotonically decreasing ($V_{i+1}(x_0) \le V_i(x_0)$), and converges to the optimal value functional $V^*(x_0)$. Correspondingly, $P_{0,i}$, $P_{1,i}(\theta)$ and $P_{2,i}(\xi,\theta)$ converge to the optimal solutions P_0^*, $P_1^*(\theta)$ and $P_2^*(\xi,\theta)$, respectively. The convergence of the model-based PI algorithm is demonstrated in Theorem 1.

Theorem 1. *Given the admissible control $u_1(x_t)$, for $P_{0,i}$, $P_{1,i}(\theta)$, $P_{2,i}(\xi,\theta)$, and $u_{i+1}(x_t)$ obtained by solving (7) and (8), and for all $i \ge 1$, the following properties hold.*

1) *$V^*(x_0) \le V_{i+1}(x_0) \le V_i(x_0)$;*
2) *$u_{i+1}(x_t)$ is admissible;*
3) *$V_i(x_0)$ and $u_i(x_t)$ converge to $V^*(x_0)$ and $u^*(x_t)$ respectively.*

Proof. Along the trajectories of (1) driven by u, $\dot{V}_i(x_t)$ is

$$\dot{V}_i(x_t) = -x^\top Qx - u_i^\top Ru_i + 2u_{i+1}^\top Ru_i - 2u^\top Ru_{i+1}. \tag{10}$$

Suppose u_i is admissible, By (10), along the state trajectories of (1) driven by u_{i+1},

$$\begin{aligned}
\dot{V}_i(x_t) = {}& -x^\top Qx - u_{i+1}^\top Ru_{i+1}\\
&- (u_{i+1} - u_i)^\top R(u_{i+1} - u_i).
\end{aligned} \tag{11}$$

Integrating both sides of (11) from 0 to ∞ yields

$$\begin{aligned}
J(x_0,u_{i+1}) = {}& V_i(x_0) - V_i(x_\infty)\\
&- \int_0^\infty (u_{i+1} - u_i)^\top R(u_{i+1} - u_i)\mathrm{d}t \le V_i(x_0) < \infty.
\end{aligned} \tag{12}$$

The statement that u_{i+1} is admissible is obtained by the finite cost of u_{i+1}. Furthermore, since $J(x_0,u_{i+1}) = V_{i+1}(x_0)$, from (12), we have $V_{i+1}(x_0) \le V_i(x_0)$. Therefore, 1) and 2) can be proved by induction given that u_1 is admissible.

Since V_i is monotonically decreasing and lower bounded by V^*, its limit exists and satisfies the ARE (6). Hence, the proof of 3) is completed. Please see [24, Theorem 1] for the details. □

Notice that although (7) is linear with respect to $P_{0,i}$, $P_{1,i}$, and $P_{2,i}$, since (7) is PDEs, solving the analytical solution to (7) is still non-trivial. Besides, the accurate knowledge of system matrices A, A_d, and B is required to implement the model-based PI, and in practice due to the complex structure of the system, it is often hard to derive such an accurate model. Therefore, in the next section, a data-driven PI algorithm is proposed to approximate the optimal solution.

Remark 2. *When $A_d = 0$, (1) is degraded to the normal linear time-invariant systems. According to (7) and (8), we can see that $P_{1,i}(\theta) = 0$, $P_{2,i}(\xi,\theta) = 0$, and $K_{1,i}(\theta) = 0$. As a consequence, (7) and (8) are same as the model-based PI method in [17]. Therefore, the proposed model-based PI algorithm is a generalization of the celebrated Kleinman algorithm to linear time-delay systems.*

IV. Data-driven Policy Iteration

The purpose of this section is to propose a corresponding data-driven PI algorithm that does not require the accurate knowledge of system (1) to solve Problem 1. The input-state trajectories data of system (1) is required for the data-driven PI, that is the continuous-time trajectories of $x(t)$ and $u(t)$ sampled from system (1) within the interval $[t_1, t_{L+1}]$ is applied to train the control policy.

By (10), along the trajectories of system (1) driven by the behavior/exploratory policy u,

$$\dot{V}_i(x_t) = -x^\top Qx - u_i^\top Ru_i - 2u_{i+1}^\top Rv_i, \tag{13}$$

where $v_i = u - u_i$. Let $[t_k, t_{k+1}]$ denote the kth segment of the interval $[t_1, t_{L+1}]$. Integrating both sides of (13) from t_k to t_{k+1} yields

$$\begin{aligned}
&V_i(x_{t_{k+1}}) - V_i(x_{t_k}) =\\
&\qquad \int_{t_k}^{t_{k+1}} -x^\top Qx - u_i^\top Ru_i - 2u_{i+1}^\top Rv_i\mathrm{d}t.
\end{aligned} \tag{14}$$

Plugging the expressions of u_{i+1} in (8) and V_i in (9) into (14), one can obtain,

$$\begin{aligned}
&\Big[x^\top(t)P_{0,i}x(t) + 2x^\top(t)\int_{-\tau}^{0}P_{1,i}(\theta)x_t(\theta)\mathrm{d}\theta\\
&\quad + \int_{-\tau}^{0}\int_{-\tau}^{0}x_t^\top(\xi)P_{2,i}(\xi,\theta)x_t(\theta)\mathrm{d}\xi\mathrm{d}\theta\Big]_{t=t_k}^{t_{k+1}}\\
&\quad - 2\int_{t_k}^{t_{k+1}}\left(x^\top(t)K_{0,i+1}^\top + \int_{-\tau}^{0}x_t^\top(\theta)K_{1,i+1}^\top(\theta)\mathrm{d}\theta\right)Rv_i(t)\mathrm{d}t\\
&= -\int_{t_k}^{t_{k+1}} x(t)^\top Qx(t) + u_i(t)^\top Ru_i(t)\mathrm{d}t.
\end{aligned} \tag{15}$$

As seen in (7) and (8), $K_{1,i}(\theta)$ and $P_{1,i}(\theta)$ ($P_{2,i}(\xi,\theta)$) are continuous functions defined over the set $[-\tau,0]$ ($[-\tau,0]^2$). Next, we use the linear combinations of basis functions to approximate these continuous functions, such that only the weighting matrices of the basis functions should be determined for the function approximation. Let $\Phi(\theta)$, $\Lambda(\xi,\theta)$, and $\Psi(\xi,\theta)$ denote the N-dimensional vectors of linearly independent basis functions. To simplify the notation, we choose the same number of basis functions for Φ, Λ and Ψ. According to the approximation theory [25], the following

equations hold

$$\begin{aligned}
&\mathrm{vecs}(P_{0,i}) = W_{0,i},\ \mathrm{vec}(P_{1,i}(\theta)) = W_{1,i}^N\Phi(\theta) + e_{\Phi,i}^N(\theta),\\
&\mathrm{diag}(P_{2,i}(\xi,\theta)) = W_{2,i}^N\Psi(\xi,\theta) + e_{\Psi,i}^N(\xi,\theta),\\
&\mathrm{vecu}(P_{2,i}(\xi,\theta)) = W_{3,i}^N\Lambda(\xi,\theta) + e_{\Lambda,i}^N(\xi,\theta),\\
&\mathrm{vec}(K_{0,i}) = U_{0,i},\ \mathrm{vec}(K_{1,i}(\theta)) = U_{1,i}^N\Phi(\theta) + e_{K,i}^N(\theta),
\end{aligned} \tag{16}$$

where $W_{0,i} \in \mathbb{R}^{n_1}$, $n_1 = \frac{n(n+1)}{2}$, $W_{1,i}^N \in \mathbb{R}^{n^2\times N}$, $W_{2,i}^N \in \mathbb{R}^{n\times N}$, $W_{3,i}^N \in \mathbb{R}^{n_2\times N}$, $n_2 = \frac{n(n-1)}{2}$, $U_{0,i} \in \mathbb{R}^{nm}$, and $U_{1,i}^N \in \mathbb{R}^{nm\times N}$ are weighting matrices of the basis functions. $e_{\Phi,i}^N(\theta) \in \mathscr{C}^0([-\tau,0],\mathbb{R}^{n^2})$, $e_{\Psi,i}^N(\xi,\theta) \in \mathscr{C}^0([-\tau,0]^2,\mathbb{R}^n)$, $e_{\Lambda,i}^N(\xi,\theta) \in \mathscr{C}^0([-\tau,0]^2,\mathbb{R}^{n_2})$, and $e_{K,i}^N(\theta) \in \mathscr{C}^0([-\tau,0],\mathbb{R}^{mn})$ are approximation truncation errors. Therefore, by the uniform approximation theory, as $N\to\infty$, the truncation errors converge uniformly to zero, i.e. for any $\eta > 0$, there exists $N^* \in \mathbb{N}_+$, such that if $N > N^*$,

$$\begin{aligned}
&\|e_{\Phi,i}^N(\theta)\|_\infty \le \eta, \qquad \|e_{K,i}^N(\theta)\|_\infty \le \eta,\\
&\|e_{\Psi,i}^N(\xi,\theta)\|_\infty \le \eta, \quad \|e_{\Lambda,i}^N(\xi,\theta)\|_\infty \le \eta.
\end{aligned} \tag{17}$$

Therefore, the key idea of data-driven PI is that $W_{j,i}(j = 0,\cdots,3)$ and $U_{j,i}(j = 0,1)$ should be approximated by the data collected from system (1). Define Υ_i^N as the composite vector of the weighting matrices, i.e.

$$\begin{aligned}
\Upsilon_i^N = \Big[&W_{0,i}^\top, \mathrm{vec}^\top(W_{1,i}^N), \mathrm{vec}^\top(W_{2,i}^N), \mathrm{vec}^\top(W_{3,i}^N)\\
&U_{0,i+1}^\top, \mathrm{vec}^\top(U_{1,i+1}^N)\Big]^\top.
\end{aligned} \tag{18}$$

Let $\hat{\Upsilon}_i^N$ be the approximation of Υ_i^N. Then, if $\hat{\Upsilon}_i^N$ is obtained, the approximations $\hat{P}_{j,i}(j = 0,1,2)$, $\hat{K}_{0,i+1}$ and $\hat{K}_{1,i+1}(\theta)$ can be reconstructed according to (16) and (18). The details of the reconstruction is shown in [24, Equations (35) and (36)]. As a consequence, $\hat{u}_i(x_t)$, the approximation of $u_i(x_t)$, can be expressed as

$$\hat{u}_i(x_t) = -\hat{K}_{0,i}x(t) - \int_{-\tau}^0 \hat{K}_{1,i}(\theta)x_t(\theta)\mathrm{d}\theta. \tag{19}$$

Based on the approximations in (16), (15) is transferred to a linear equation with respect to $\hat{\Upsilon}_i^N$. Then, the unknown vector Υ_i^N is approximated by linear regression. In detail, let $\hat{v}_i = u - \hat{u}_i$ be the approximation of v_i with $\hat{u}_i$. $\tilde{u}_i = \hat{u}_i - u_i$ denotes the deviation between the policies of model-based PI and data-driven PI at the ith iteration. Define the data-constructed matrices

$$\begin{aligned}
&\Gamma_{\Phi xx}(t) = \int_{-\tau}^0 \Phi^\top(\theta)\otimes x_t^\top(\theta)\otimes x^\top(t)\mathrm{d}\theta,\\
&\Gamma_{\Psi xx}(t) = \int_{-\tau}^0\int_{-\tau}^0 \Psi^\top(\xi,\theta)\otimes \mathrm{vecd}^\top(x_t(\xi),x_t(\theta))\mathrm{d}\xi\mathrm{d}\theta,\\
&\Gamma_{\Lambda xx}(t) = \int_{-\tau}^0\int_{-\tau}^0 \Lambda^\top(\xi,\theta)\otimes \mathrm{vecp}^\top(x_t(\xi),x_t(\theta))\mathrm{d}\xi\mathrm{d}\theta,\\
&G_{x\hat{v}_i,k} = \int_{t_k}^{t_{k+1}} (x^\top(t)\otimes \hat{v}_i^\top(t))(I_n\otimes R)\mathrm{d}t,\\
&G_{\Phi x\hat{v}_i,k} = \int_{t_k}^{t_{k+1}}\int_{-\tau}^0 \Phi^\top(\theta)\otimes((x_t^\top(\theta)\otimes \hat{v}_i^\top(t))(I_n\otimes R))\mathrm{d}\theta\mathrm{d}t.
\end{aligned} \tag{20}$$

With the collected data, following variables are defined

$$\begin{aligned}
M_{i,k} = \Big[&\mathrm{vecv}^\top(x(t))|_{t_k}^{t_{k+1}}, 2\Gamma_{\Phi xx}|_{t_k}^{t_{k+1}}, \Gamma_{\Psi xx}(t)|_{t_k}^{t_{k+1}},\\
&\Gamma_{\Lambda xx}(t)|_{t_k}^{t_{k+1}}, -2G_{x\hat{v}_i,k}, -2G_{\Phi x\hat{v}_i,k}\Big],\\
Y_{i,k} = &-\int_{t_k}^{t_{k+1}} x^\top Qx + \hat{u}_i^\top R\hat{u}_i dt,\\
E_{i,k} = &[2\varepsilon_{1,i}(t) + \varepsilon_{2,i}(t) + \varepsilon_{3,i}(t)]_{t=t_k}^{t_{k+1}} - 2\psi_{i,k} - 2\rho_{i,k}^0\\
&- 2\rho_{i,k}^1 - \rho_{i,k}^2,\\
M_i = &\Big[M_{i,1}^\top,\cdots,M_{i,k}^\top,\cdots,M_{i,L}^\top\Big]^\top,\\
Y_i = &\Big[Y_{i,1},\cdots,Y_{i,k},\cdots,Y_{i,L}\Big]^\top,\\
E_i = &\Big[E_{i,1},\cdots,E_{i,k},\cdots,E_{i,L}\Big]^\top,
\end{aligned} \tag{21}$$

where $\varepsilon_{j,i}$ $(j = 1,2,3)$, $\psi_{i,k}$, $\rho_{i,k}^0$, $\rho_{i,k}^1$, and $\rho_{i,k}^2$ are induced by the truncation errors. Their detailed expressions are in [24, Equations (38) and (39)].

By the definitions of $M_{i,k}$, $Y_{i,k}$ and $E_{i,k}$ in (21), (15) is finally transferred as a linear equation with respect to Υ_i^N (See details in [24, Equation (38)]),

$$M_{i,k}\Upsilon_i^N + E_{i,k} = Y_{i,k}. \tag{22}$$

Combining equations of (22) from $k = 1$ to $k = L$, we have

$$M_i\Upsilon_i^N + E_i = Y_i. \tag{23}$$

Let $\hat{E}_i$ be defined such that

$$\hat{E}_i = Y_i - M_i\hat{\Upsilon}_i^N. \tag{24}$$

Assumption 2. *Given $N \in \mathbb{N}_+$, there exist $L^* \in \mathbb{N}_+$ and $\alpha > 0$, such that for all $L > L^*$ and $i \in \mathbb{N}_+$,*

$$\frac{1}{L}M_i^\top M_i \ge \alpha I. \tag{25}$$

Remark 3. *Assumption 2 is reminiscent of the persistent excitation (PE) condition [26], [27]. As in the literature of ADP-based data-driven control [3], [4], one can fulfill it by means of added exploration noise, such as sinusoidal signals and random noise.*

Under Assumption 2, the method of least squares is applied to minimize $\hat{E}_i^\top\hat{E}_i$, i.e. $\hat{E}_i^\top\hat{E}_i$ is minimized by

$$\hat{\Upsilon}_i^N = M_i^\dagger Y_i. \tag{26}$$

With the result of $\hat{\Upsilon}_i^N$ in (26), $\hat{P}_{j,i}(j = 0\cdots 2)$ and $\hat{K}_{j,i}(j = 0,1)$ can be reconstructed by (16) and (18).

The proposed algorithm is shown in Algorithm 1. From (21), M_i and Y_i are constructed by the input-state trajectory data of system (1). Hence, the system matrices are not involved in the computation of $\hat{\Upsilon}_i^N$. Furthermore, since the behavior policy u is different from the updated policy u_i, Algorithm 1 is called off-policy.

Remark 4. *Due to the property that $P_{2,i}^\top(\xi,\theta) = P_{2,i}(\theta,\xi)$, the diagonal elements of $P_{2,i}$ satisfy $diag(P_{2,i}(\xi,\theta)) = diag(P_{2,i}(\theta,\xi))$. Hence, the vector of basis functions Ψ should satisfy $\Psi(\xi,\theta) = \Psi(\theta,\xi)$ to approximate such functions.*

Algorithm 1 Data-driven Policy Iteration

1: Choose the vector of the basis functions $\Phi(\theta)$, $\Psi(\xi,\theta)$, and $\Lambda(\xi,\theta)$.
2: Choose $L\in\mathbb{N}_+$ and the sampling instance $t_k\in[t_1,t_{L+1}]$.
3: Choose input $u=u_1+e$, with e an exploration signal, to explore the system (1) and collect the input-state data $u(t),x(t),t\in[0,t_{L+1}]$. Set the threshold $\delta>0$ and $i=1$.
4: **repeat**
5: Given $\hat{u}_i$ and the data, construct M_i and Y_i by (21).
6: Get $\hat{\Upsilon}_i^N$ by solving (26).
7: Get $\hat{K}_{0,i+1}$ and $\hat{K}_{1,i+1}$ from $\hat{\Upsilon}_i^N$.
8: $\hat{u}_{i+1}(x_t)=-\hat{K}_{0,i+1}x(t)-\int_{-\tau}^{0}\hat{K}_{1,i+1}(\theta)x_t(\theta)\mathrm{d}\theta$
9: $i\leftarrow i+1$
10: **until** $|\hat{\Upsilon}_i^N-\hat{\Upsilon}_{i-1}^N|<\delta$.
11: Use $\hat{u}_i(x_t)$ as the control input.

Lemma 2. *Under Assumption 2, and given an admissible controller $u_1(x_t)=-K_{0,1}x(t)-\int_{-\tau}^{0}K_{1,1}(\theta)x_t(\theta)\mathrm{d}\theta$, for any $i\in\mathbb{N}_+$ and $\eta>0$, there exists some positive integer $N^*>0$, such that if $N>N^*$,*

$$|\hat{P}_{0,i}-P_{0,i}|\le\eta,\ \|\hat{P}_{1,i}-P_{1,i}\|_\infty\le\eta,\ \|\hat{P}_{2,i}-P_{2,i}\|_\infty\le\eta$$
$$|\hat{K}_{0,i+1}-K_{0,i+1}|\le\eta,\ \|\hat{K}_{1,i+1}-K_{1,i+1}\|_\infty\le\eta. \tag{27}$$

Proof. This lemma is proved by induction. When $i=1$, $\hat{u}_1=u_1$ and $\tilde{u}_1=0$. Then, under Assumption 2, we have $\lim_{N\to\infty}\tilde{\Upsilon}_1^{N\top}\tilde{\Upsilon}_1^N=0$. Hence, (27) holds for $i=1$. Suppose (27) holds for some $i-1\ge1$. Consequently, $\lim_{N\to\infty}\hat{u}_i=u_i$. Under Assumption 2, we have $\lim_{N\to\infty}\tilde{\Upsilon}_i^{N\top}\tilde{\Upsilon}_i^N=0$. Hence, (27) holds for i. See the detailed proof at [24, Lemma 3]. □

Theorem 2. *Given an admissible controller u_1, for any $\eta>0$, there exist integers $i^*>0$ and $N^{**}>0$, such that if $N>N^{**}$*

$$|\hat{P}_{0,i^*}-P_0^*|\le\eta,\ \|\hat{P}_{1,i^*}-P_1^*\|_\infty\le\eta,\ \|\hat{P}_{2,i^*}-P_2^*\|_\infty\le\eta,$$
$$|\hat{K}_{0,i^*+1}-K_0^*|\le\eta,\ \|\hat{K}_{1,i^*+1}-K_1^*\|_\infty\le\eta. \tag{28}$$

Proof. This theorem is from Lemma 2 and the triangle inequality of the norm. See the details in [24, Theorem 2]. □

By Theorem 2, we see that $\hat{K}_{j,i+1}(j=0,1)$ obtained by Algorithm 1 converges to $K_j^*(j=0,1)$ as the iteration step of the algorithm and the number of basis functions tend to infinity. Hence, the proposed data-driven PI solves Problem 1.

V. Application to Autonomous Driving

Algorithm 1 is applied to design a learning-based controller for a platoon of connected and autonomous vehicles (CAVs) in mixed traffic environments to mitigate the effect of human drivers' reaction delay and to attenuate the stop-and-go waves of traffic flow. This problem is studied by [28], where a model-based LQ control is applied to design the controller for CAVs.

A platoon consisting of two human-driven vehicles (HDVs) and one autonomous vehicle (AV) is shown in Fig. 1.

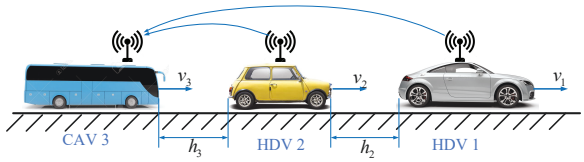

Fig. 1: A platoon consisting of two HDVs and an AV.

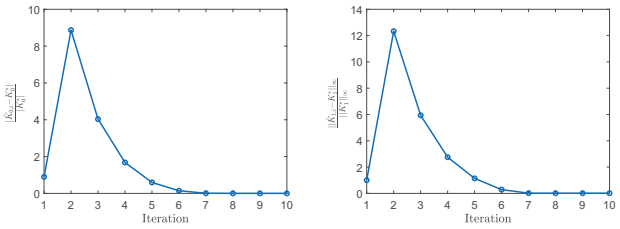

Fig. 2: Convergence of $\hat{K}_{0,i}$ and $\hat{K}_{1,i}(\theta)$ to K_0^* and $K_1^*(\theta)$.

The AV is at the last position. When there are sudden changes in the traffic (e.g. the preceding vehicle is decelerating suddenly), the HDV will make delayed reactions. That is why a platoon consisting of HDVs is a time-delay system. In Fig. 1, h_i denotes the bumper-to-bumper distance between the ith vehicle and $(i-1)$th vehicle, and v_i denotes the velocity of the ith vehicle. Define the relative headway as $\Delta h_i=h_i-h^*$ and the relative velocity as $\Delta v_i=v_i-v^*$, where (h^*,v^*) is the equilibrium of the platoon. The acceleration of the AV is the control input of the platoon. Assuming the velocity of the leading vehicle is constant, the system can be described as a linear time-delay system (1) with $x=[\Delta h_2,\Delta v_2,\Delta h_3,\Delta v_3]^\top$,

$$A=\begin{bmatrix}0&-1&0&0\\0&0&0&0\\0&1&0&-1\\0&0&0&0\end{bmatrix},\ A_d=\begin{bmatrix}0&0&0&0\\\alpha_2c^*&-(\alpha_2+\beta_2)&0&0\\0&0&0&0\\0&0&0&0\end{bmatrix},$$

and $B=\begin{bmatrix}0&0&0&1\end{bmatrix}^\top$, where α_2 and β_2 denote the human driver parameters and c^* is the derivative of the range policy [14], [28]. In the simulation, the human parameters are set as $\alpha_2=0.1$, $\beta_2=0.2$, and $c^*=1.5708$. The human reaction delay is $\tau=1.2s$. The weighting matrix in (3) is $Q=\mathrm{diag}([1,1,10,10])$, and $R=1$. The initial state of the platoon is $[x_0(\theta)]_i=30\sum_{j=1}^{10}\sin w_{i,j}\theta+[\chi]_i$ for $i=1,2,3,4$. $w_{i,j}$ and $[\chi]_i$ are randomly sampled from the uniform distributions over $[-10,10]$ and $[-30,30]$, respectively. The initial admissible controller is $\hat{u}_1(x_t)=-K_{0,1}x(t)-\int_{-\tau}^{0}K_{1,1}(\theta)x_t(\theta)\mathrm{d}\theta$, with $K_{0,1}=\begin{bmatrix}-0.0897&-0.2772&-0.3&0.5196\end{bmatrix}$ and $K_{1,1}(\theta)=0$. The added exploratory noise in Algorithm 1 is $e(t)=\sum_{i=1}^{200}\sin\omega_it$. ω_i is randomly sampled from an independent uniform distribution over $[-100,100]$. $u=u_1+e$ is applied to collect the input-state data from the system. The basis functions in (16) are $\Phi(\theta)=[1,\theta,\theta^2,\theta^3]^\top$, $\Psi(\xi,\theta)=[1,\xi+\theta,\xi^2+\theta^2,\xi\theta,\xi^3+\theta^3,\xi^2\theta+\xi\theta^2,\xi^3\theta+\xi\theta^3,\xi^2\theta^2,\xi^3\theta^2+\xi^2\theta^3,\xi^3\theta^3]^\top$, and $\Lambda(\xi,\theta)=[1,\theta,\theta^2,\theta^3]^\top\otimes[1,\xi,\xi^2,\xi^3]^\top$. The approach in [28] is adopted to calculate the optimal values of K_0^* and K_1^*, where the precise model is assumed known.

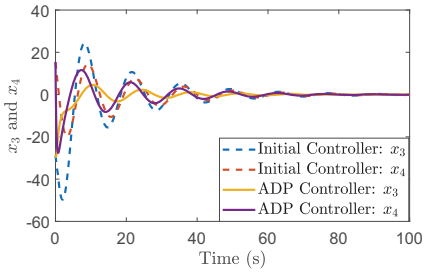

Fig. 3: Comparison between the initial controller and the ADP controller.

Fig. 2 shows the relative error of the obtained gain $\hat{K}_{0,i}$ and $\hat{K}_{1,i}(\theta)$ at each iteration. It is seen that the relative errors converge after the tenth iteration. This is consistent with Theorem 2. In particular, at the tenth iteration, the relative errors are $\frac{|\hat{K}_{0,10}-K_0^*|}{|K_0^*|} = 0.0008$ and $\frac{||\hat{K}_{1,10}-K_1^*||_\infty}{||K_1^*||_\infty} = 0.0137$. Hence, the optimal controller is well approximated by the proposed data-driven PI algorithm. The comparison between the learned controller at the tenth iteration and the initial controller is conducted. Since the AV is at the last position of the platoon, the movement of HDVs cannot be influenced by the AV. Hence, with different controllers, the evolution of Δh_2 and Δv_2 is same. With the initial controller $\hat{u}_1$ and the learned ADP controller $\hat{u}_{10}$, the AV's relative headway Δh_3 and relative velocity Δv_3 are shown in Fig. 3. It is seen that with the ADP controller, the state converges to the equilibrium more quickly. Hence, the stop-and-go waves of the AV is attenuated by the proposed data-driven PI algorithm. The values of the performance index are $J(x_0, \hat{u}_1) = 1.46 \cdot 10^5$ and $J(x_0, \hat{u}_{10}) = 4.73 \cdot 10^4$. The proposed data-driven PI algorithm minimizes the performance index and improves the performance of the AV in the platoon.

VI. Conclusions

In this paper, we have proposed innovative model-based and data-driven PI algorithms for linear time-delay systems. The proposed model-based PI can be considered as an extension of the celebrated Kleinman's PI [17] to linear time-delay systems. Based on the model-based PI, and only using input-state trajectories of the system, a data-driven PI algorithm has been proposed to approximate the value functional and control law at each iteration. It is rigorously shown that the value functional and control law generated at each iteration converge to the optimal solution. Furthermore, the proposed adaptive optimal control method is applied to CAVs in mixed traffic environments to attenuate the stop-and-go waves and mitigate the effect of human driver reaction delays.

References

[1] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An Introduction*. MA, USA: MIT Press, 2018.

[2] 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.

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

[4] F. L. Lewis and D. Liu, *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control*. NJ, USA: Wiley-IEEE Press, 2013.

[5] W. Gao and Z. P. Jiang, "Adaptive dynamic programming and adaptive optimal output regulation of linear systems," *IEEE Trans. Autom. Control*, vol. 61, no. 12, pp. 4164–4169, 2016.

[6] B. Pang and Z. P. Jiang, "Adaptive optimal control of linear periodic systems: an off-policy value iteration approach," *IEEE Trans. Autom. Control*, vol. 66, no. 2, pp. 888–894, 2021.

[7] L. Cui, S. Wang, J. Zhang, D. Zhang, J. Lai, Y. Zheng, Z. Zhang, and Z. P. Jiang, "Learning-based balance control of wheel-legged robots," *IEEE Robot. Autom. Lett.*, vol. 6, no. 4, pp. 7667–7674, 2021.

[8] W. Gao, Z. P. Jiang, and K. Ozbay, "Data-driven adaptive optimal control of connected vehicles," *IEEE Transactions on Intelligent Transportation Systems*, vol. 18, no. 5, pp. 1122–1133, 2017.

[9] D. McRuer, "Human dynamics in man-machine systems," *Automatica*, vol. 16, no. 3, pp. 237–253, 1980.

[10] H. Xu, S. Jagannathan, and F. Lewis, "Stochastic optimal control of unknown linear networked control system in the presence of random delays and packet losses," *Automatica*, vol. 48, no. 6, pp. 1017–1030, 2012.

[11] S. A. Asad Rizvi, Y. Wei, and Z. Lin, "Model-free optimal stabilization of unknown time delay systems using adaptive dynamic programming," in *Proc. IEEE Conf. Decis. Control.*, pp. 6536–6541, 2019.

[12] J. G. Rueda-Escobedo, E. Fridman, and J. Schiffer, "Data-driven control for linear discrete-time delay systems," *IEEE Transactions on Automatic Control*, vol. 67, no. 7, pp. 3321–3336, 2022.

[13] H. Zhang, R. Song, Q. Wei, and T. Zhang, "Optimal tracking control for a class of nonlinear discrete-time systems with time delays based on heuristic dynamic programming," *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 1851–1862, 2011.

[14] 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, 2022.

[15] M. Huang, Z. P. Jiang, M. Malisoff, and L. Cui, "Robust autonomous driving with human in the loop," in *Handbook of Reinforcement Learning and Control* (K. G. Vamvoudakis, Y. Wan, F. L. Lewis, and D. Cansever, eds.), pp. 62–77, New York, NY, USA: Springer, 2021.

[16] R. Moghadam, S. Jagannathan, V. Narayanan, and K. Raghavan, *Optimal Adaptive Control of Partially Uncertain Linear Continuous-Time Systems with State Delay*, pp. 243–272. Cham: Springer International Publishing, 2021.

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

[18] R. F. Curtain, *An Introduction to Infinite-Dimensional Linear Systems Theory*. New York, NY: Springer, 1995.

[19] K. Gu, V. L. Kharitonov, and J. Chen, *Stability of Time-Delay Systems*. Boston, MA: Birkhäuser, 2003.

[20] E. Fridman, *Introduction to Time-Delay Systems Analysis and Control*. Switzerland: Springer, 2014.

[21] R. E. Kalman, "Contributions to the theory of optimal control," *Boletin Sociedad Matematica Mexicana*, vol. 5, no. 2, pp. 102–119, 1960.

[22] D. Ross and I. Flügge-Lotz, "An optimal control problem for systems with differential-difference equation dynamics," *SIAM J. Control Optim.*, vol. 7, no. 4, pp. 609–623, 1969.

[23] K. Uchida, E. Shimemura, T. Kubo, and N. ABE, "The linear-quadratic optimal control approach to feedback control design for systems with delay," *Automatica*, vol. 24, no. 6, pp. 773–780, 1988.

[24] L. Cui, B. Pang, and Z. P. Jiang, "Learning-based adaptive optimal control of linear time-delay systems: A policy iteration approach," *arXiv preprint arXiv:2210.00204*, 2022.

[25] M. J. D. Powell, *Approximation Theory and Methods*. New York, NY: Cambridge University Press, 1981.

[26] Z. P. Jiang, C. Prieur, and A. Astolfi (Editors), *Trends in Nonlinear and Adaptive Control: A Tribute to Laurent Praly for His 65th Birthday,*. NY, USA: Springer Nature, 2021.

[27] K. J. Åström and B. Wittenmark, *Adaptive control, 2nd Edition*. MA, USA: Addison-Wesley, 1997.

[28] J. I. Ge and G. Orosz, "Optimal control of connected vehicle systems with communication delay and driver reaction time," *IEEE Trans. Intell. Transp. Syst.*, vol. 18, no. 8, pp. 2056–2070, 2017.