Available online at www.sciencedirect.com

ScienceDirect

IFAC PapersOnline CONFERENCE PAPER ARCHIVE

IFAC PapersOnLine 56-2 (2023) 6964–6969

A Value Iteration Approach to Adaptive Optimal Control of Linear Time-Delay Systems

Leilei Cui * **Bo Pang** * **Zhong-Ping Jiang** *

* *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,bo.pang,zjiang@nyu.edu).*

Abstract: This paper studies the adaptive optimal control for linear time-delay systems described by delay differential equations (DDEs). A key strategy is to exploit the value iteration (VI) approach to solve the linear quadratic optimal control problem for time-delay systems. However, previous learning-based control methods are all exclusively devoted to discrete-time time-delay systems. In this article, we aim to fill in the gap by developing a learning-based VI approach to solve the infinite-dimensional algebraic Riccati equation (ARE) for continuous-time time-delay systems. One nice feature of the proposed VI approach is that an initial admissible controller is not required to start the algorithm. The efficacy of the proposed methodology is demonstrated by the example of autonomous driving.

Keywords: Data-based control, time-delay systems

1. INTRODUCTION

In the past decades, time-delay systems attracted numerous interests from researchers. It is well known that the optimal control can ensure the stability and performance of the closed-loop system under some mild conditions. Therefore, the optimal control problem for time-delay systems is fundamentally important, yet challenging, in control fields for a long time. The author in Krasovskii (1962) first studied the linear quadratic (LQ) optimal control problem for continuous-time linear systems with state delay. Following this original work, in Ross and Flügge-Lotz (1969), the sufficient condition for the optimal control was derived as a set of partial differential equations (PDEs). These PDEs can be considered as the extension of the ARE for delay-free systems to time-delay systems. However, the precise system model is needed for solving the ARE. In reality, such an accurate system model is hard to obtain. Hence, it is significant to propose a learning-based controller design approach for time-delay systems. Adaptive dynamic programming (ADP) is a potential method for this problem.

By integrating the reinforcement learning (RL) technique with the classical control theory, ADP was developed to learn a stabilizing and optimal control policy using finite samples of input-state data (Jiang et al. (2020); Lewis and Liu (2013)). Recently, based on the ADP technique, substantial progress has been made on the learning-based control for various important classes of linear/nonlinear/periodic dynamical systems for optimal state stabilization and output regulation (Jiang and Jiang (2012); Gao and Jiang (2016); Pang and Jiang (2021); Cui and Jiang (2022)). ADP has been successfully applied in various engineering fields, for example, wheel-legged robots (Cui et al. (2021)), autonomous driving (Chakraborty et al. (2022)), and human motor control (Pang et al. (2022)). Unlike finite-dimensional systems, the optimal controller for time-delay systems is a functional of the state, which poses a major challenge for the learning-based adaptive optimal control of time-delay systems. In the most of the relevant literature, e.g. Asad Rizvi et al. (2019); Liu et al. (2016); Huang et al. (2022); Rueda-Escobedo et al. (2022), the learning-based control problem for discrete-time time-delay systems is solved. Since the discrete-time time-delay systems are finite dimensional, these methods cannot be directly applied to continuous-time systems with time delays.

In this paper, a novel VI-based ADP algorithm is proposed to find a near-optimal controller for linear time-delay systems without the precise knowledge of system dynamics. It is first shown that the solution of the finite-horizon LQ optimal control problem (as a differential Riccati equation (DRE)) asymptotically converges to the solution of the infinite-horizon LQ optimal control problem. By combining the convergence property of DRE with the RL technique, a learning-based VI approach is proposed to find a near-optimal controller using the input-state data collected along the trajectories of the system.

The rest content of this paper is organized as follows. Section II introduces the preliminaries for the LQ optimal control of time-delay systems. Section III proposes a model-based VI approach to find a near-optimal controller. Section IV develops a learning-based VI approach based on ADP technique. Section V demonstrates the efficacy of the proposed learning-based VI approach by numerical simulations. The paper is concluded in Section VI.

Notations: In this paper, $\mathbb{R}$ denotes the set of real numbers. $|\cdot|$ denotes the Euclidean norm of a vector or Frobe-

⋆ This work has been supported in part by the NSF grants EPCN-1903781 and ECCS-2210320.

10.1016/j.ifacol.2023.10.524

nius norm of a matrix, and $\|\cdot\|_\infty$ denotes the supreme norm of a function. $\frac{\mathrm{d}f}{\mathrm{d}\theta}(\cdot)$ denotes the function which is the derivative of the function f. $\oplus$ 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, and $\mathcal{M}_2 = \mathbb{R}^n \oplus L_2([-\tau,0],\mathbb{R}^n)$. $\mathcal{L}(X)$ denotes the class of continuous bounded linear operators from X to X. $\langle\cdot,\cdot\rangle$ denotes the inner product in $\mathcal{M}_2$, i.e. $\langle z_1, z_2\rangle = r_1^\top r_2 + \int_{-\tau}^0 f_1^\top(\theta) f_2(\theta)\mathrm{d}\theta$, where $z_i = [r_i, f_i(\cdot)]^\top$ for $i = 1, 2$. $\mathrm{vec}(A) = \left[a_1^\top, a_2^\top, ..., a_n^\top\right]^\top$, where a_i is the ith column of A. $\mathrm{vec}^{-1}(\cdot)$ is the inverse operator of $\mathrm{vec}(\cdot)$. For a symmetric matrix $P \in \mathbb{R}^{n\times n}$, $\mathrm{vecs}(P) = [p_{11}, 2p_{12}, ..., 2p_{1n}, p_{22}, 2p_{23}, ..., 2p_{(n-1)n}, p_{nn}]^\top$, $\mathrm{vecu}(P) = [2p_{12}, ..., 2p_{1n}, 2p_{23}, ..., 2p_{(n-1)n}]^\top$, $\mathrm{diag}(P) = [p_{11}, p_{22}, ..., p_{nn}]^\top$. For the vectors $\nu, \mu \in \mathbb{R}^n$, $\mathrm{vecd}(\nu,\mu) = [\nu_1\mu_1, .., \nu_n\mu_n]^\top$, $\mathrm{vecv}(\nu) = [\nu_1^2, ..., \nu_1\nu_n, ..., \nu_{n-1}\nu_n, \nu_n^2]^\top$, and $\mathrm{vecp}(\nu,\mu) = [\nu_1\mu_2, ..., \nu_1\mu_n, \nu_2\mu_3, ..., \nu_{n-1}\mu_n]^\top$. $[a]_{i,j}$ denotes the sub-vector of the vector a comprised of the entries between the ith and jth entries. $A^\dagger$ denotes the Moore-Penrose inverse of A.

2. PROBLEM FORMULATION AND PRELIMINARIES

2.1 Problem Formulation

This paper considers the following continuous-time linear time-delay system:

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

where $\tau \geq 0$ is the delay of the system, which is constant and known, $x(t) \in \mathbb{R}^n$, 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. A segment of the trajectory for $x(t)$ within the interval $[t-\tau, t]$ is denoted as $x_t(\theta) = x(t+\theta)$, $\forall\theta \in [-\tau, 0]$. Since system (1) is infinite dimensional, the system's state is $z(t) = [x^\top(t), x_t^\top(\cdot)]^\top \in \mathcal{M}_2$. The quadratic performance index adopted for system (1) is

$$\begin{aligned}\min_u J(x_0, u) &= \int_0^\infty x(t)^\top Q x(t) + u(t)^\top R u(t)\mathrm{d}t \\ &= \int_0^\infty \langle z(t), \mathbf{Q} z(t)\rangle + u(t)^\top R u(t)\mathrm{d}t,\end{aligned} \tag{2}$$

where $R^\top = R > 0$, $Q^\top = Q \geq 0$, and $\mathbf{Q} = \begin{bmatrix} Q & \\ & \mathbf{0}\end{bmatrix} \in \mathcal{L}(\mathcal{M}_2)$ is symmetric (Eidelman et al., 2004, Chapter 6), and non-negative (Eidelman et al., 2004, Definition 6.3.1).

It is assumed that system (1) with the output $y(t) = Q^{\frac{1}{2}}x(t)$ is exponentially stabilizable and detectable (Curtain, 1995, Definition 5.2.1). The problem to be studied in this paper is formulated as follows.

Problem (*VI-based ADP*) Without the knowledge of the system dynamics in (1), design a VI-based ADP algorithm to find the approximation of the optimal controller using only the input-state data measured along the trajectories of the system.

2.2 Optimality and Stability

For system (1), the following lemma gives the expression of the optimal controller for (2).

Lemma 1. (Uchida et al. (1988)). Consider system (1), the optimal controller for (2) is

$$u^*(x_t) = -\underbrace{R^{-1}B^\top P_0^*}_{K_0^*} x(t) - \int_{-\tau}^0 \underbrace{R^{-1}B^\top P_1^*(\theta)}_{K_1^*(\theta)} x_t(\theta)\mathrm{d}\theta, \tag{3}$$

and the corresponding minimal performance index is

$$\begin{aligned}V^*(x_0) = x_0^\top(0)P_0^* x_0(0) + 2x_0^\top(0)\int_{-\tau}^0 P_1^*(\theta)x_0(\theta)\mathrm{d}\theta \\ + \int_{-\tau}^0\int_{-\tau}^0 x_0^\top(\xi)P_2^*(\xi,\theta)x_0(\theta)\mathrm{d}\xi\mathrm{d}\theta,\end{aligned} \tag{4}$$

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

$$\begin{aligned}&A^\top P_0^* + P_0^* A - P_0^* B R^{-1} B^\top P_0^* \\ &\quad + P_1^*(0) + P_1^{*\top}(0) + Q = 0, \\ &\frac{\mathrm{d}P_1^*(\theta)}{\mathrm{d}\theta} = (A^\top - P_0^* B R^{-1} B^\top)P_1^*(\theta) + P_2^*(0,\theta), \\ &\partial_\xi P_2^*(\xi,\theta) + \partial_\theta P_2^*(\xi,\theta) = -P_1^{*\top}(\xi)BR^{-1}B^\top P_1^*(\theta), \\ &P_1^*(-\tau) = P_0^* A_d, \quad P_2^*(-\tau,\theta) = A_d^\top P_1^*(\theta).\end{aligned} \tag{5}$$

We can consider (5) as the ARE for time-delay systems. By (Curtain, 1995, Theorem 6.2.7), the closed-loop system with u^* in (3) is exponentially stable at the origin.

3. CONTINUOUS-TIME VALUE ITERATION

VI-based ADP is derived from the DRE, which is related to the finite-horizon optimal control problem:

$$\begin{aligned}&\min_u \mathcal{J}(t_0, T, x_{t_0}, u) = \int_{t_0}^T \langle z(t), \mathbf{Q} z(t)\rangle + u^\top(t) R u(t)\mathrm{d}t \\ &\text{subject to (1)},\end{aligned} \tag{6}$$

where x_{t_0} is the initial segment of $x(t)$, t_0 is the initial time, and T is the terminal time. The following lemma gives the solution to (6).

Lemma 2. For problem (6), the minimal performance index $V(x_{t_0}, t_0) = \min_u \mathcal{J}(t_0, T, x_{t_0}, u)$ can be expressed as

$$V(x_{t_0}, t_0) = \langle z_0, \mathbf{P}(t_0) z_0\rangle, \tag{7}$$

where $z_0 = [x^\top(t_0), x_{t_0}^\top(\cdot)]^\top$ is the initial state, and $\mathbf{P}(s)z_0$ is expressed as

$$\mathbf{P}(s)z_0 = \begin{bmatrix} P_0(s)x(t_0) + \int_{-\tau}^0 P_1(s,\theta)x_{t_0}(\theta)\mathrm{d}\theta \\ \int_{-\tau}^0 P_2(s,\cdot,\theta)x_{t_0}(\theta)\mathrm{d}\theta + P_1^\top(s,\cdot)x(t_0)\end{bmatrix}. \tag{8}$$

Here, $P_0(s) = P_0^\top(s)$, $P_1(s,\theta)$ and $P_2(s,\xi,\theta) = P_2^\top(s,\theta,\xi)$ can be obtained by solving the following PDEs backwards

$$\begin{aligned}&\frac{\mathrm{d}}{\mathrm{d}s}P_0(s) = -A^\top P_0(s) - P_0(s)A - Q - P_1(s,0) \\ &\quad - P_1^\top(s,0) + P_0(s)BR^{-1}B^\top P_0(s), &\text{(9a)}\\ &\partial_s P_1(s,\theta) = \partial_\theta P_1(s,\theta) - P_2(s,0,\theta) \\ &\quad - (A^\top - P_0(s)BR^{-1}B^\top)P_1(s,\theta), &\text{(9b)}\\ &\partial_s P_2(s,\xi,\theta) = \partial_\xi P_2(s,\xi,\theta) + \partial_\theta P_2(s,\xi,\theta) \\ &\quad + P_1^\top(s,\xi)BR^{-1}B^\top P_1(s,\theta), &\text{(9c)}\\ &P_1(s,-\tau) = P_0(s)A_d, \quad P_2(s,-\tau,\theta) = A_d^\top P_1(s,\theta), &\text{(9d)}\\ &P_0(T) = 0, \quad P_1(T,\theta) = 0, \quad P_2(T,\xi,\theta) = 0. &\text{(9e)}\end{aligned}$$

Theorem 3. The solution of (9) satisfies

$$\lim_{s\to-\infty}|P_0(s)-P_0^*|=0, \tag{10a}$$
$$\lim_{s\to-\infty}\|P_1(s,\theta)-P_1^*(\theta)\|_\infty=0, \tag{10b}$$
$$\lim_{s\to-\infty}\|P_2(s,\xi,\theta)-P_2^*(\xi,\theta)\|_\infty=0. \tag{10c}$$

Theorem 3 implies that the PDEs (5) can be solved by solving (9) backwards from the terminal time T to $-\infty$. However, in (9), the system matrices A, A_d, and B are required and it is non-trivial to solve such complicated PDEs. In the next section, in the absence of the accurate model of the system, a VI-based ADP algorithm will be proposed to solve (9) using the input-state data measured along the system's trajectories.

4. DATA-DRIVEN VALUE ITERATION

In this section, we suppose only that the continuous-time trajectories of $x(t)$ and $u(t)$ within the time interval $[t_1, t_{L+1}]$ are available for the optimal controller design.

Recall that $P_0(s)$, $P_1(s,\theta)$, and $P_2(s,\xi,\theta)$ are the solutions to (9) and $\mathbf{P}(s)z$ is defined in (8). According to (7) and (8), $V(x_t,s)$ can be expressed as

$$\begin{aligned}V(x_t,s)&=x^\top(t)P_0(s)x(t)+2x^\top(t)\int_{-\tau}^{0}P_1(s,\theta)x_t(\theta)\mathrm{d}\theta\\&+\int_{-\tau}^{0}\int_{-\tau}^{0}x_t^\top(\xi)P_2(s,\xi,\theta)x_t(\theta)\mathrm{d}\xi\mathrm{d}\theta.\end{aligned} \tag{11}$$

Along the trajectories of system (1) driven by the control input u, considering the partial integration and the formula $\partial_t x(t+\theta)=\partial_\theta x(t+\theta)$, we have

$$\begin{aligned}\frac{\mathrm{d}}{\mathrm{d}t}V(x_t,s)&=x^\top(t)[A^\top P_0(s)+P_0(s)A\\&\quad+P_1^\top(s,0)+P_1(s,0)]x(t)\\&+2x^\top(t-\tau)[A_d^\top P_0(s)-P_1^\top(s,-\tau)]x(t)\\&+2x^\top(t)\int_{-\tau}^{0}[A^\top P_1(s,\theta)-\partial_\theta P_1+P_2(s,0,\theta)]x_t(\theta)\mathrm{d}\theta\\&+2x^\top(t-\tau)\int_{-\tau}^{0}[A_d^\top P_1(s,\theta)-P_2(s,-\tau,\theta)]x_t(\theta)\mathrm{d}\theta\\&-\int_{-\tau}^{0}\int_{-\tau}^{0}x_t^\top(\xi)[\partial_\xi P_2(s,\xi,\theta)+\partial_\theta P_2(s,\xi,\theta)]x_t(\theta)\mathrm{d}\xi\mathrm{d}\theta\\&+2u^\top(t)B^\top P_0(s)x(t)+2u^\top(t)B^\top\int_{-\tau}^{0}P_1(s,\theta)x_t(\theta)\mathrm{d}\theta.\end{aligned} \tag{12}$$

Define the following matrix-valued functions

$$\begin{aligned}H_0(s)&=A^\top P_0(s)+P_0(s)A+P_1^\top(s,0)+P_1(s,0),\\H_1(s,\theta)&=A^\top P_1(s,\theta)+P_2(s,0,\theta)-\partial_\theta P_1(s,\theta),\\H_2(s,\xi,\theta)&=\partial_\xi P_2(s,\xi,\theta)+\partial_\theta P_2(s,\xi,\theta),\\K_0(s)&=R^{-1}B^\top P_0(s),\\K_1(s,\theta)&=R^{-1}B^\top P_1(s,\theta).\end{aligned} \tag{13}$$

Then, from Theorem 3, it is seen that as $s\to-\infty$, $H_0(s)$, $H_1(s,\theta)$, $H_2(s,\theta,\xi)$, $K_0(s)$, and $K_1(s,\theta)$ can well approximate H_0^*, $H_1^*(\theta)$, $H_2^*(\xi,\theta)$, K_0^*, and $K_1^*(\theta)$, where the superscript $*$ denotes that in (13) P_j is replaced by P_j^* for $j=0,1,2$. Since for each fixed algorithmic time $s\in(-\infty,T]$, $H_1(s,\theta)$ and $K_1(s,\theta)$ ($H_2(s,\xi,\theta)$) are continuous functions defined on the interval $[-\tau,0]$ ($[-\tau,0]^2$), we use the linear combinations of the basis functions to approximate these continuous functions. Let $\Phi(\theta)$, $\Lambda(\xi,\theta)$, and $\Psi(\xi,\theta)$ denote the N-dimensional linearly independent basis functions. Without losing the generality, it is supposed that the dimensions of Φ, Λ, and Ψ are same. Then, by the uniform approximation theory (Powell (1981)), for each fixed algorithmic time $s\in(-\infty,T]$, we have

$$\begin{aligned}&\mathrm{vecs}(H_0)=W_0(s),\\&\mathrm{vec}(H_1)=W_1^N(s)\Phi(\theta)+e_{H\Phi}^N(s,\theta),\\&\mathrm{diag}(H_2)=W_2^N(s)\Psi(\xi,\theta)+e_{H\Psi}^N(s,\xi,\theta),\\&\mathrm{vecu}(H_2)=W_3^N(s)\Lambda(\xi,\theta)+e_{H\Lambda}^N(s,\xi,\theta),\\&\mathrm{vec}(K_0)=U_0(s),\\&\mathrm{vec}(K_1)=U_1^N(s)\Phi(\theta)+e_{K\Phi}^N(s,\theta),\end{aligned} \tag{14}$$

where $W_0(s)\in\mathbb{R}^{n_1}$, $n_1=\frac{n(n+1)}{2}$, $W_1^N(s)\in\mathbb{R}^{n^2\times N}$, $W_2^N(s)\in\mathbb{R}^{n\times N}$, $W_3^N(s)\in\mathbb{R}^{n_2\times N}$, $n_2=\frac{n(n-1)}{2}$, $U_0(s)\in\mathbb{R}^{nm}$, and $U_1^N(s)\in\mathbb{R}^{nm\times N}$ are weighting matrices. $e_{H\Phi}^N(s,\theta)$ and $e_{K\Phi}^N(s,\theta)$ ($e_{H\Psi}^N(s,\xi,\theta)$ and $e_{H\Lambda}^N(s,\xi,\theta)$) are truncation errors, and they converge to zero uniformly in $\theta\in[-\tau,0]$ ($\xi,\theta\in[-\tau,0]$), and pointwisely in $s\in(-\infty,T]$, as the number of basis functions N tends to infinity.

By plugging (9d) and (13) into (12), integrating (12) from t_k to t_{k+1}, and vectorizing the equation, we have

$$\begin{aligned}&V(x_{t_{k+1}},s)-V(x_{t_k},s)\\&=\int_{t_k}^{t_{k+1}}\mathrm{vecv}^\top(x(t))\mathrm{d}t\mathrm{vecs}(H_0(s))\\&+2\int_{t_k}^{t_{k+1}}\int_{-\tau}^{0}x_t^\top(\theta)\otimes x^\top(t)\mathrm{vec}(H_1(s,\theta))\mathrm{d}\theta\mathrm{d}t\\&-\int_{t_k}^{t_{k+1}}\int_{-\tau}^{0}\int_{-\tau}^{0}\mathrm{vecd}^\top(x_t(\xi),x_t(\theta))\\&\qquad\mathrm{diag}(H_2(s,\xi,\theta))\mathrm{d}\xi\mathrm{d}\theta\mathrm{d}t\\&-\int_{t_k}^{t_{k+1}}\int_{-\tau}^{0}\int_{-\tau}^{0}\mathrm{vecp}^\top(x_t(\xi),x_t(\theta))\\&\qquad\mathrm{vecu}(H_2(s,\xi,\theta))\mathrm{d}\xi\mathrm{d}\theta\mathrm{d}t\\&+2\int_{t_k}^{t_{k+1}}x^\top(t)\otimes(u^\top(t)R)\mathrm{d}t\mathrm{vec}(K_0(s))\\&+2\int_{t_k}^{t_{k+1}}\int_{-\tau}^{0}x_t^\top(\theta)\otimes(u^\top(t)R)\mathrm{vec}(K_1(s,\theta))\mathrm{d}\theta\mathrm{d}t,\end{aligned} \tag{15}$$

where $t_1<t_2<\cdots<t_k<\cdots<t_{L+1}$ is the boundary of each integral window. The following variables are defined to simplify the notations

$$\Gamma_{\Phi xx}(t)=\int_{-\tau}^{0}\Phi^\top(\theta)\otimes x_t^\top(\theta)\otimes x^\top(t)\mathrm{d}\theta \tag{16}$$
$$\Gamma_{\Psi xx}(t)=\int_{-\tau}^{0}\int_{-\tau}^{0}\Psi^\top(\xi,\theta)\otimes\mathrm{vecd}^\top(x_t(\xi),x_t(\theta))\mathrm{d}\xi\mathrm{d}\theta$$
$$\Gamma_{\Lambda xx}(t)=\int_{-\tau}^{0}\int_{-\tau}^{0}\Lambda^\top(\xi,\theta)\otimes\mathrm{vecp}^\top(x_t(\xi)x_t(\theta))\mathrm{d}\xi\mathrm{d}\theta$$
$$\Gamma_{\Phi\Phi xx}(t)=\int_{-\tau}^{0}\int_{-\tau}^{0}\Phi^\top(\theta)\otimes\Phi^\top(\xi)\otimes x_t^\top(\theta)\otimes x_t^\top(\xi)\mathrm{d}\xi\mathrm{d}\theta$$

In addition, define the following variables as the integration of the sampled state and input trajectory

$$
\begin{aligned}
I_{xx,k} &= \int_{t_k}^{t_{k+1}} \mathrm{vecv}^\top(x(t))\mathrm{d}t,\\
I_{xu,k} &= \int_{t_k}^{t_{k+1}} x^\top(t)\otimes(u^\top(t)R)\mathrm{d}t,\\
I_{\Phi xx,k} &= \int_{t_k}^{t_{k+1}} \Gamma_{\Phi xx}(t)\mathrm{d}t,\\
I_{\Phi xu,k} &= \int_{t_k}^{t_{k+1}}\int_{-\tau}^{0} \Phi^\top(\theta)\otimes x_t^\top(\theta)\otimes(u^\top(t)R)\mathrm{d}\theta\mathrm{d}t,\\
I_{\Psi xx,k} &= \int_{t_k}^{t_{k+1}} \Gamma_{\Psi xx}(t)\mathrm{d}t,\ I_{\Lambda xx,k} = \int_{t_k}^{t_{k+1}} \Gamma_{\Lambda xx}(t)\mathrm{d}t.
\end{aligned}
\tag{17}
$$

Plugging (14) and (17) into (15) yields

$$
\begin{aligned}
&V(x_{t_{k+1}},s)-V(x_{t_k},s) = I_{xx,k}W_0(s)+2I_{\Phi xx,k}\\
&\mathrm{vec}(W_1^N(s))-I_{\Psi xx,k}\mathrm{vec}(W_2^N(s))-I_{\Lambda xx,k}\mathrm{vec}(W_3^N(s))\\
&+2I_{xu,k}U_0(s)+2I_{\Phi xu,k}\mathrm{vec}(U_1^N(s))+e_k^N(s),
\end{aligned}
\tag{18}
$$

where $e_k^N(s)$ is induced by the truncation errors in (14). Combining (18) for $k=1,2,...,L$, one can obtain the following linear equation with respect to the weighting matrices encoded in Ω_N

$$
\Theta_N\Omega_N(s)+E_L^N(s)=\Xi(s),
\tag{19}
$$

where

$$
\begin{aligned}
\Omega_N(s) &= [W_0^\top(s),\mathrm{vec}^\top(W_1^N(s)),\mathrm{vec}^\top(W_2^N(s)),\\
&\quad \mathrm{vec}^\top(W_3^N(s)),U_0^\top(s),\mathrm{vec}^\top(U_1(s))]^\top,\\
\Theta_N &= \left[\sigma_1^\top,...,\sigma_k^\top,...,\sigma_L^\top\right]^\top,\\
E_L^N(s) &= \left[e_1^N(s),...,e_k^N(s),...,e_L^N(s)\right]^\top,\\
\Xi(s) &= \left[V(x_t,s)|_{t=t_1}^{t_2},...,V(x_t,s)|_{t=t_k}^{t_{k+1}},...,V(x_t,s)|_{t=t_L}^{t_{L+1}}\right]^\top,\\
\sigma_k &= [I_{xx,k},2I_{\Phi xx,k},-I_{\Psi xx,k},-I_{\Lambda xx,k},2I_{xu,k},2I_{\Phi xu,k}].
\end{aligned}
\tag{20}
$$

The following assumption on the matrix Θ_N is made to ensure that the collected data is rich enough such that the least-square solution to (19) is unique.

Assumption 4. Given $N>0$, there exists $L^*>0$ and $\alpha>0$, such that for all $L>L^*$,

$$
\frac{1}{L}\Theta_N^\top\Theta_N\geq\alpha I.
\tag{21}
$$

Remark 5. Assumptions 4 is reminiscent of the condition of persistent excitation (Jiang et al. (2021); Åström and Wittenmark (1997)). As shown in the past literature of ADP (Jiang and Jiang (2017); Lewis and Liu (2013)), one can fulfill such a condition by means of added exploration noise, such as sinusoidal signals and random noise.

Now, at each fixed algorithmic time $s\in(-\infty,T]$, by solving (19) via least-square methods, one can get the weighting matrices in (14). Next, by differentiating (19) with respect to the algorithmic time s, we will solve (9) by a data-driven method. Since $V(x_t,s)$ is involved in the expression of $\Xi(s)$, the first thing is to differentiate $V(x_t,s)$ with respect to s. Recalling the expression of $V(x_t,s)$ in (11), we have

$$
\begin{aligned}
\frac{\mathrm{d}}{\mathrm{d}s}V(x_t,s) &= x^\top(t)\frac{\mathrm{d}P_0(s)}{\mathrm{d}s}x(t)\\
&+2x^\top(t)\int_{-\tau}^{0}\partial_sP_1(s,\theta)x_t(\theta)\mathrm{d}\theta\\
&+\int_{-\tau}^{0}\int_{-\tau}^{0}x_t^\top(\xi)\partial_sP_2(s,\xi,\theta)x_t(\theta)\mathrm{d}\xi\mathrm{d}\theta.
\end{aligned}
\tag{22}
$$

Plugging (9) into (22), and vectorizing the equation, we have

$$
\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}s}V(x_t,s)\\
&=\mathrm{vecv}^\top(x(t))[-W_0(s)-\mathrm{vecs}(Q)+\mathrm{vecs}(K_0^\top RK_0)]\\
&+2\int_{-\tau}^{0}x_t^\top(\theta)\otimes x^\top(t)[-\mathrm{vec}(H_1)+\mathrm{vec}(K_0^\top RK_1)]\mathrm{d}\theta\\
&+\int_{-\tau}^{0}\int_{-\tau}^{0}\mathrm{vecd}^\top(x_t(\xi),x_t(\theta))\mathrm{diag}(H_2)\\
&+\mathrm{vecp}^\top(x_t(\xi),x_t(\theta))\mathrm{vecu}(H_2)\\
&+x_t^\top(\theta)\otimes x_t^\top(\xi)\mathrm{vec}(K_1^\top RK_1)\mathrm{d}\xi\mathrm{d}\theta,
\end{aligned}
\tag{23}
$$

By the approximations of $K_0(s)$ and $K_1(s,\theta)$ in (14), $\mathrm{vecs}(K_0^\top RK_0)$, $\mathrm{vec}(K_0^\top RK_1)$, and $\mathrm{vec}(K_1^\top RK_1)$ can be approximated by $\mathcal{K}_{v,0}$, $\mathcal{K}_{v,1}^N$, and $\mathcal{K}_{v,2}^N$, which are

$$
\begin{aligned}
\mathcal{K}_{v,0} &= \mathrm{vecs}[\mathrm{vec}^{-\top}(U_0(s))R\mathrm{vec}^{-1}(U_0(s))],\\
\mathcal{K}_{v,1}^N &= \mathrm{vec}[\mathrm{vec}^{-\top}(U_0(s))R\mathrm{vec}^{-1}(U_1^N(s)\Phi(\theta))]\\
\mathcal{K}_{v,2}^N &= \mathrm{vec}[\mathrm{vec}^{-\top}(U_1^N(s)\Phi(\xi))R\mathrm{vec}^{-1}(U_1^N(s)\Phi(\theta))]
\end{aligned}
\tag{24}
$$

Plugging (14) and (24) into (23) gives us the following equation

$$
\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}s}V(x_t,s)=\mathrm{vecv}^\top(x(t))[-W_0(s)-\mathrm{vecs}(Q)+\mathcal{K}_{v,0}(s)]\\
&-2\Gamma_{\Phi xx}(t)\mathrm{vec}(W_1^N(s))+2\int_{-\tau}^{0}x_t^\top(\theta)\otimes x^\top(t)\mathcal{K}_{v,1}^N(s,\theta)\mathrm{d}\theta\\
&+\Gamma_{\Psi xx}(t)\mathrm{vec}(W_2^N(s))+\Gamma_{\Lambda xx}(t)\mathrm{vec}(W_3^N(s))\\
&+\int_{-\tau}^{0}\int_{-\tau}^{0}x_t^\top(\theta)\otimes x_t^\top(\xi)\mathcal{K}_{v,2}^N(s,\xi,\theta)\mathrm{d}\xi\mathrm{d}\theta+\varepsilon_N(t,s),
\end{aligned}
\tag{25}
$$

where $\varepsilon_N(t,s)$ is induced by the truncation errors in (14). By Lemma 8 and the expressions of $\mathcal{K}_{v,1}^N$ and $\mathcal{K}_{v,2}^N$ in (24), the integrals in (25) involving $\mathcal{K}_{v,1}^N$ and $\mathcal{K}_{v,2}^N$ can be further simplified, and $\frac{\mathrm{d}}{\mathrm{d}s}V(x_t,s)$ is finally derived as

$$
\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}s}V(x_t,s)=\mathrm{vecv}^\top(x(t))[-W_0(s)-\mathrm{vecs}(Q)+\mathcal{K}_{v,0}(s)]\\
&+2\Gamma_{\Phi xx}(t)[-\mathrm{vec}(W_1^N(s))+\mathcal{U}_1(U_0(s),U_1^N(s),R)]\\
&+\Gamma_{\Psi xx}(t)\mathrm{vec}(W_2^N(s))+\Gamma_{\Lambda xx}(t)\mathrm{vec}(W_3^N(s))\qquad\text{(26a)}\\
&+\Gamma_{\Phi\Phi xx}(t)\mathcal{U}_2(U_1^N(s),R)+\varepsilon_N(t,s)\\
&=\mathcal{W}_N^\top(x_t)\mathcal{V}(\Omega_N(s))+\varepsilon_N(t,s),\qquad\text{(26b)}
\end{aligned}
$$

where $\mathcal{W}_N$ and $\mathcal{V}$ are defined as

$$
\begin{aligned}
\mathcal{W}_N(x_t) &= [\mathrm{vecv}^\top(x(t)),2\Gamma_{\Phi xx}(t),\Gamma_{\Psi xx}(t),\qquad\text{(27a)}\\
&\quad \Gamma_{\Lambda xx}(t),\Gamma_{\Phi\Phi xx}(t)]^\top,\\
\mathcal{V}(\Omega_N(s)) &= \left[[-W_0(s)-\mathrm{vecs}(Q)+\mathcal{K}_{v,0}(s)]^\top,\right.\\
&\quad [-\mathrm{vec}(W_1^N(s))+\mathcal{U}_1(U_0(s),U_1^N(s),R)]^\top,\qquad\text{(27b)}\\
&\quad \left.\mathrm{vec}^\top(W_2^N(s)),\mathrm{vec}^\top(W_3^N(s)),\mathcal{U}_2^\top(U_1^N(s),R)\right]^\top.
\end{aligned}
$$

Under Assumption 4, following (26) and differentiating the both sides of (19) with respect to the algorithmic time s, we have

$$
\begin{aligned}
&\frac{\mathrm{d}}{\mathrm{d}s}\Omega_N(s)=\mathcal{H}_N(\Omega_N(s))+\mathcal{G}_N(s),\\
&\Omega_N(T)=0,
\end{aligned}
\tag{28}
$$

where $\Omega_N(T)=0$ is obtained by (9e). The expressions of $\mathcal{H}_N(\Omega_N(s))$ and $\mathcal{G}_N(\Omega_N(s),s)$ are

Algorithm 1 Data-driven Value Iteration

1: Choose T, and the vector of the basis functions $\Phi(\theta)$, $\Psi(\xi,\theta)$, and $\Lambda(\xi,\theta)$.
2: Choose the boundaries of the sampling windows $t_1 \leq t_k \leq t_{L+1}$.
3: Choose the driving input u to explore system (1) and collect the input-state data $u(t), x(t), t \in [t_1, t_{L+1}]$.
4: Construct data matrices Θ_N and Ξ_d^N.
5: Solve (30) backwards from $s = T$ to $s = 0$.
6: Get $\hat{K}_0(0)$ and $\hat{K}_1(0,\theta)$ by (31).

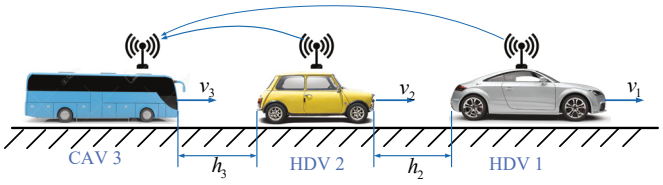

Fig. 1. A string of HDVs and an AV.

$$\mathcal{H}_N(\Omega_N(s)) = \Theta_N^{\dagger}\Xi_d^N\mathcal{V}(\Omega_N(s)), \tag{29a}$$
$$\mathcal{G}_N(s) = \Theta_N^{\dagger}(-\frac{\mathrm{d}}{\mathrm{d}s}E_L^N(s) + \Xi_e^N(s)), \tag{29b}$$
$$\Xi_d^N = [\mathcal{W}_N(x_t)|_{t_1}^{t_2}, \cdots, \mathcal{W}_N(x_t)|_{t_L}^{t_{L+1}}]^{\top}, \tag{29c}$$
$$\Xi_e^N(s) = [\varepsilon_N(t,s)|_{t_1}^{t_2}, \cdots, \varepsilon_N(t,s)|_{t_L}^{t_{L+1}}]^{\top}. \tag{29d}$$

It is seen that by utilizing the collected data, (9) is transferred to (28) where the system matrices (A, A_d, B) are not involved. In (28), $\mathcal{G}_N$ is induced by the truncation errors. Hence, if the truncation errors are small enough to be ignored, the solution to (28) can be approximated by the solution to the following differential equation

$$\frac{\mathrm{d}}{\mathrm{d}s}\hat{\Omega}_N(s) = \mathcal{H}_N(\hat{\Omega}_N(s)), \quad \hat{\Omega}_N(T) = 0. \tag{30}$$

With the obtained $\hat{\Omega}_N(s)$, by (20), $\hat{U}_0(s)$ and $\hat{U}_1^N(s)$ can be obtained from the corresponding elements encoded in $\hat{\Omega}_N(s)$. Following (14), the estimation of $K_0(s)$ and $K_1(s,\theta)$ can be obtained by

$$\begin{aligned}\hat{K}_0(s) &= \mathrm{vec}^{-1}([\hat{\Omega}_N(s)]_{n_3,n_4}),\\ \hat{U}_1^N(s) &= \mathrm{vec}^{-1}([\hat{\Omega}_N(s)]_{n_4+1,n_5}),\\ \hat{K}_1(s,\theta) &= \mathrm{vec}^{-1}(\hat{U}_1^N(s)\Phi(\theta)).\end{aligned} \tag{31}$$

where $n_3 = n_1 + n^2 + n + n_2 + 1$, $n_4 = n_3 + mn$, and $n_5 = n_4 + mnN$.

Algorithm 1 shows the detail of the data-driven VI algorithm. The following theorem shows the main result of the learning-based VI algorithm, i.e. the optimal control gains K_0^* and $K_1^*(\theta)$ can be well approximated by solving (30) backwards.

Theorem 6. For any $\epsilon > 0$, there exist $T^*(\epsilon) > 0$ and $N_3^*(\epsilon, T) > 0$, such that if $T > T^*(\epsilon)$ and $N > N_3^*(\epsilon, T)$, the following inequalities hold.

$$|\hat{U}_0(0) - \mathrm{vec}(K_0^*)| \leq \epsilon \tag{32a}$$
$$\left\|\hat{U}_1^N(0)\Phi(\theta) - \mathrm{vec}(K_1^*(\theta))\right\|_{\infty} \leq \epsilon \tag{32b}$$

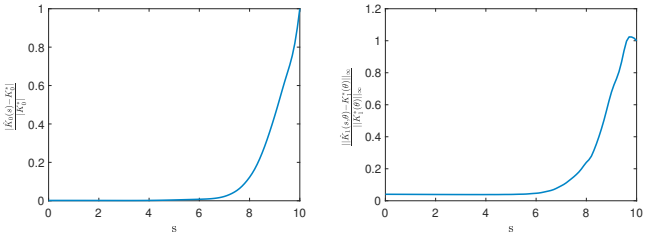

Fig. 2. Convergence of $\hat{K}_0(s)$ and $\hat{K}_1(s,\theta)$ to the optimal values K_0^* and $K_1^*(\theta)$ by VI algorithm.

5. NUMERICAL SIMULATION

In this section, we demonstrate the effectiveness of the proposed learning-based VI algorithm by the practical example with regards to connected and autonomous vehicles (CAVs) in mixed traffic consisting of both autonomous vehicles (AVs) and human-driven vehicles (HDVs).

Consider a string of two HDVs and one AV as shown in Fig. 1, where 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 $\Delta h_i = h_i - h^*$ and $\Delta v_i = v_i - v^*$, where (h^*, v^*) is the equilibrium of the vehicles. h^* depends on the human parameters and $v^* = v_1$. Assuming the velocity of the leading vehicle is constant, and considering the time-delay effect caused by human drivers' reaction time, the system can be described as (1) with $x = [\Delta h_2, \Delta v_2, \Delta h_3, \Delta v_3]^{\top}$, and (A, A_d, B) defined in (Cui et al., 2022, Section V.B). In the simulation, the weighting matrices of the performance index are $Q = \mathrm{diag}([1,1,10,10])$, and $R = 1$. The basis functions 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 analytical expression of optimal values K_0^* and K_1^* are derived by the method in Ge and Orosz (2017), where the precise model of the system is required.

In Algorithm 1, $\hat{\Omega}_N$ is iterated backwards from $T = 10$ to 0. It is noticed that $T = 10$ is the length of the algorithmic time instead of the physical time. In Fig. 2, it is seen that $\hat{K}_0(s)$ and $\hat{K}_1(s)$ converge to the optimal values eventually, and the relative approximation errors are $\frac{|\hat{K}_0(0) - K_0^*|}{|K_0^*|} = 0.0017$ and $\frac{||\hat{K}_1(0,\theta) - K_1^*(\theta)||_{\infty}}{||K_1^*(\theta)||_{\infty}} = 0.0406$. Therefore, the proposed VI algorithm is able to well approximate the optimal controller. Compared with Ge and Orosz (2017), our approach is learning-based and precise model knowledge is not required.

6. CONCLUSIONS

This paper has proposed for the first time a learning-based VI algorithm for a class of linear time-delay systems described by DDEs. The first major contribution is the development of a model-based VI approach for continuous-time linear time-delay systems. Second, by integrating RL techniques, a learning-based VI algorithm is proposed for learning adaptive optimal controllers from data in the absence of the precise system model knowledge. The efficacy of the proposed learning-based adaptive optimal control

design method has been validated by the application arising from connected vehicles.

Appendix A. AUXILIARY RESULTS

Lemma 7. Given $V \in \mathbb{R}^{m\times n}$ and $v = \text{vec}(V)$. For the integer $0 \le i \le (mn-1)$, let j and k be the quotient and reminder of i/m, respectively. $D_i \in \mathbb{R}^{m\times n}$ is defined such that the entry of D_i at the $(k+1)$th row and $(j+1)$th column is 1, and all the other entries are 0. Then,

$$V = \text{vec}^{-1}(v) = \sum_{i=0}^{mn-1} D_i v_{i+1}, \tag{A.1}$$

where v_i is the ith element of v.

Proof. By the definition of the operator $\text{vec}(\cdot)$, v_{i+1} is the entry of V at the $(k+1)$th row and $(j+1)$th column. Hence, V is reconstructed in (A.1) by iterative placing v_{i+1} to the position of the $(k+1)$th row and $(j+1)$th column.

Lemma 8. For any $\chi \in \mathbb{R}^{n^2}$, $U_0 \in \mathbb{R}^{mn}$, $U_1 \in \mathbb{R}^{mn\times N}$, $R \in \mathbb{R}^{m\times m}$, and $\Phi_1, \Phi_2 \in \mathbb{R}^N$,

$$\chi^\top \text{vec}[\text{vec}^{-\top}(U_0) R \text{vec}^{-1}(U_1\Phi_1)] = \Phi_1^\top \otimes \chi^\top \mathcal{U}_1(U_0, U_1, R)$$

$$\begin{aligned}&\chi^\top \text{vec}[\text{vec}^{-\top}(U_1\Phi_1) R \text{vec}^{-1}(U_1\Phi_2)]\\ &\quad = \Phi_2^\top \otimes \Phi_1^\top \otimes \chi^\top \mathcal{U}_2(U_1, R)\end{aligned}$$

where $\mathcal{U}_1$ and $\mathcal{U}_2$ are defined as

$$\mathcal{U}_1(U_0, U_1, R) = \text{vec}\left[\sum_{i=1}^{mn} \text{vec}\left(\text{vec}^{-\top}(U_0) R D_i\right)[U_1]_i\right],$$

$$\mathcal{U}_2(U_1, R) = \text{vec}\left[\sum_{i,j=1}^{mn} \text{vec}\left(D_i^\top R D_j\right)\text{vec}^\top([U_1]_i^\top [U_1]_j)\right].$$

Proof. The lemma is a consequence of Lemma 7.

REFERENCES

Asad Rizvi, S.A., Wei, Y., and Lin, Z. (2019). Model-free optimal stabilization of unknown time delay systems using adaptive dynamic programming. In *Proc. IEEE Conf. Decis. Control.*, 6536–6541.

Chakraborty, S., Cui, L., Ozbay, K., and Jiang, Z.P. (2022). Automated lane changing control in mixed traffic: An adaptive dynamic programming approach. In *25th IEEE International Conference on Intelligent Transportation Systems (ITSC)*, 1823–1828.

Cui, L. and Jiang, Z.P. (2022). A reinforcement learning look at risk-sensitive linear quadratic gaussian control. *arXiv preprint arXiv:2212.02072*.

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

Cui, L., Wang, S., Zhang, J., Zhang, D., Lai, J., Zheng, Y., Zhang, Z., and Jiang, Z.P. (2021). Learning-based balance control of wheel-legged robots. *IEEE Robotics and Automation Letters*, 6(4), 7667–7674.

Curtain, R.F. (1995). *An Introduction to Infinite-Dimensional Linear Systems Theory*. Springer, New York, NY.

Eidelman, Y., Milman, V., and Tsolomitis, A. (2004). *Functional Analysis, An Introduction*. American Mathematical Society, Rhode Island, USA.

Gao, W. and Jiang, Z. (2016). Adaptive dynamic programming and adaptive optimal output regulation of linear systems. *IEEE Trans. Autom. Control*, 61(12), 4164–4169.

Ge, J.I. and Orosz, G. (2017). Optimal control of connected vehicle systems with communication delay and driver reaction time. *IEEE Trans. Intell. Transp. Syst.*, 18(8), 2056–2070.

Huang, M., Jiang, Z.P., and Ozbay, K. (2022). Learning-based adaptive optimal control for connected vehicles in mixed traffic: robustness to driver reaction time. *IEEE Trans. Cybern.*, 52(6), 5267–5277.

Jiang, Y. and Jiang, Z.P. (2017). *Robust Adaptive Dynamic Programming*. Wiley-IEEE Press, NJ, USA.

Jiang, Y. and Jiang, Z.P. (2012). Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics. *Automatica*, 48(10), 2699–2704.

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

Jiang, Z.P., Bian, T., and Gao, W. (2020). Learning-based control: A tutorial and some recent results. *Found. Trends Syst. Control*, 8(3), 176–284.

Krasovskii, N. (1962). On the analytic construction of an optimal control in a system with time lags. *Journal of Applied Mathematics and Mechanics*, 26(1), 50–67.

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

Liu, Y., Zhang, H., Luo, Y., and Han, J. (2016). ADP based optimal tracking control for a class of linear discrete-time system with multiple delays. *Journal of the Franklin Institute*, 353(9), 2117–2136.

Pang, B. and Jiang, Z.P. (2021). Adaptive optimal control of linear periodic systems: an off-policy value iteration approach. *IEEE Trans. Autom. Control*, 66(2), 888–894.

Pang, B., Cui, L., and Jiang, Z.P. (2022). Human motor learning is robust to control-dependent noise. *Biological Cybernetics*, 116(3), 307–325.

Powell, M.J.D. (1981). *Approximation Theory and Methods*. Cambridge University Press, New York, NY.

Ross, D. and Flügge-Lotz, I. (1969). An optimal control problem for systems with differential-difference equation dynamics. *SIAM J. Control Optim.*, 7(4), 609–623.

Rueda-Escobedo, J.G., Fridman, E., and Schiffer, J. (2022). Data-driven control for linear discrete-time delay systems. *IEEE Trans. Autom. Control*, 67(7), 3321–3336.

Uchida, K., Shimemura, E., Kubo, T., and ABE, N. (1988). The linear-quadratic optimal control approach to feedback control design for systems with delay. *Automatica*, 24(6), 773–780.

Åström, K.J. and Wittenmark, B. (1997). *Adaptive control, 2nd Edition*. Addison-Wesley, MA, USA.