Supplementary Information for

Learning the Continuous-Time Optimal Decision Law from Discrete-Time Rewards

Ci Chen, Lihua Xie, Kan Xie, Frank L. Lewis, Yilu Liu, and Shengli Xie

E-mail: ci.chen@gdut.edu.cn, elhxie@ntu.edu.sg, shlxie@gdut.edu.cn

This PDF file includes:

Supporting Information Text

Below we provide preliminaries and additional details of the derivation of the presented analytical framework in the paper of *Learning the Continuous-Time Optimal Decision Law from Discrete-Time Rewards*.

1. Model-based Continuous-time Optimal Decision Law Design

This section serves as background material for elaborating the preliminaries for the optimal decision law design with prior knowledge of the model dynamics and its state.

A. Problem Formulation of Model-based Optimal Control. Here, we consider the following linear continuous-time dynamical systems that have long been a common concern of different communities, ranging from the control science (1), the neuroscience (2, 3), to the complex network science (4–6):

$$\begin{cases} \dot{\mathbf{x}}(t) = \mathbf{A}\mathbf{x}(t) + \mathbf{B}\mathbf{u}(t) \\ \mathbf{y}(t) = \mathbf{C}\mathbf{x}(t), \end{cases} \quad [1]$$

where t denotes the time, $\mathbf{x}(t) \in \mathbb{R}^n$ and $\mathbf{y}(t) \in \mathbb{R}^p$, respectively, denote the state and output of the system, $\mathbf{u}(t) \in \mathbb{R}^m$ is the system input to be designed, and the system matrices $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ are assumed to be constant.

We want to design a control policy $\mathbf{u}(t)$ in Eq.1 to solve the following optimization problem

$$\begin{cases} J(\mathbf{Q}, \mathbf{R}, \mathbf{u}(t), \mathbf{y}(t)) = \min\limits_{\mathbf{u}(t)} \int_0^\infty r(\mathbf{u}(t), \mathbf{y}(t))dt \\ \text{subject to Eq.1}, \end{cases} \quad [2]$$

where $r(\mathbf{u}(t), \mathbf{y}(t)) = \mathbf{u}^T(t)\mathbf{R}\mathbf{u}(t) + \mathbf{y}^T(t)\mathbf{Q}\mathbf{y}(t)$ with $\mathbf{Q} = \mathbf{Q}^T > 0$ and $\mathbf{R} = \mathbf{R}^T > 0$. The problem is called linear quadratic regulator (LQR) in the field of control system community, wherein the following standard assumption is made on Eq.1 for solving the above LQR problem.

***Assumption* 1** *The pair* $(\mathbf{A}, \mathbf{B})$ *is controllable and* $(\mathbf{A}, \mathbf{C})$ *is observable.*

We notice that there is a weaker condition of stabilizability and detectability in the literature. Here, *Assumption 1* is required so that the parameterization matrix has full row rank under the controllability condition (see *Lemma 2*, where the full row rank of the matrix is used in the proof) and all the open-loop poles can be placed through a companion matrix for the state parameterization under the observability condition (see the proof of *Lemma 1*).

B. Preliminaries to Model-based Optimal Control Solution. The problem formulated in Eq.2 is solvable according to the optimal control theory (1). If all the system dynamics including $\mathbf{A}$, $\mathbf{B}$, and $\mathbf{C}$ and the system state $\mathbf{x}(t)$ are available for the learning and control design, then the control policy u satisfying the optimization criterion in Eq.2 is given by

$$\mathbf{u}(t) = -\mathbf{K}^*\mathbf{x}(t), \quad [3]$$

where $\mathbf{K}^*$ is called an optimal control gain matrix satisfying

$$\mathbf{K}^* = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}^* \quad [4]$$

with $\mathbf{P}^*$ being the stabilizing solution to the algebraic Riccati equation,

$$\mathbf{C}^T\mathbf{Q}\mathbf{C} + \mathbf{A}^T\mathbf{P}^* - \mathbf{P}^*\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}^* + \mathbf{P}^*\mathbf{A} = \mathbf{O}. \quad [5]$$

We notice that calculating the optimal controller directly from Eq.3 requires full knowledge of the system dynamics of Eq.1 and the system state, namely, $\mathbf{A}$, $\mathbf{B}$, $\mathbf{C}$, and $\mathbf{x}(t)$. In what follows, we aim to solve the optimization problem Eq.2 using the input-output data along real-time system trajectories and discrete-time rewards.

2. Continuous-time Optimal Decision Learning Algorithm Based on Discrete-time Rewards

In this section, we aim to study the reinforcement learning (RL)-based optimal decision by means of discrete-time rewards. Policy iteration is used for implementing the RL. To this end, we assume that an initial stabilizing policy

$$\mathbf{u}(t) = -\mathbf{K}_0\mathbf{y}(t) \quad [6]$$

is available for control and learning. That is, under Eq.6, the matrix $\mathbf{A} - \mathbf{B}\mathbf{K}^0 \triangleq \mathbf{A} - \mathbf{B}\mathbf{K}_0\mathbf{C}$ is Hurwitz meaning that the eigenvalues of $\mathbf{A} - \mathbf{B}\mathbf{K}^0$ have strictly negative real parts, while the optimization in Eq.2 is not necessarily satisfied. The prior knowledge of the initial stabilizing gain in Eq.6 is required in this work and the existing works in the field of policy iteration such as (1), and can be obtained by (7).

A. Derivation of State Derivative Reconstruction Using Measured Input-Output Sampled Data. To bypass the state required in Eq.3, one requires the measured inputs and outputs along the system trajectory for reconstructing the system state. To do this, consider the following artificial linear systems

$$\dot{\eta}_{\mathbf{u}}(t) = (\mathbf{I}_m \otimes \mathbf{D}_\eta)\eta_{\mathbf{u}}(t) + \mathbf{u}(t) \otimes \mathbf{b}, \tag{7}$$

and

$$\dot{\eta}_{\mathbf{y}}(t) = (\mathbf{I}_p \otimes \mathbf{D}_\eta)\eta_{\mathbf{y}}(t) + \mathbf{y}(t) \otimes \mathbf{b}, \tag{8}$$

where $\eta_{\mathbf{u}}(0) = \mathbf{0}$, $\eta_{\mathbf{y}}(0) = \mathbf{0}$, the symbol $\otimes$ denotes the Kronecker product operator,

$$\mathbf{D}_\eta = \begin{bmatrix} 0 & 1 & 0 & \cdots & 0 \\ 0 & 0 & 1 & \cdots & 0 \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & 1 \\ -d_n & -d_{n-1} & \cdots & \cdots & -d_1 \end{bmatrix} \in \mathbb{R}^{n\times n}, \tag{9}$$

and

$$\mathbf{b} = [\quad 0, \quad 0, \quad \ldots, \quad 0, \quad 1 \quad]^T \in \mathbb{R}^n \tag{10}$$

with d_j for $j = 1, 2, \cdots, n$ being positive coefficients so that $\mathbf{D}_\eta$ can be Hurwitz. Define vectors

$$\eta(t) = [\eta_{\mathbf{u}}^T(t), \eta_{\mathbf{y}}^T(t)]^T \in \mathbb{R}^{n(m+p)} \tag{11}$$

and

$$\theta(t) = [\theta_{\mathbf{u}}^T(t), \theta_{\mathbf{y}}^T(t)]^T = [\dot{\eta}_{\mathbf{u}}^T(t), \dot{\eta}_{\mathbf{y}}^T(t)]^T \in \mathbb{R}^{n(m+p)},$$

wherein $\eta(t)$ is a specific form of the feedforward signal $\mathbf{\Phi}(\mathbf{u}(t), \mathbf{y}(t))$ given in Eq.**5** of the main context. From Eq.**7** and Eq.**8**, $\eta(t_i)$ and $\theta(t_i)$ are known vectors with the non-uniformly sampling time instants $t_1 < t_2 < \ldots < t_i < \ldots < t_{s-1} < t_s$, and t_1 and t_s being, respectively, time instants when the data collection starts and ends.

The following result shows that both the system state and its derivative at the sampling time t_i can be interpreted by the data of the system input and output.

***Lemma* 1** *Let Assumption 1 hold. Then, there exist matrices* $\mathbf{\Gamma} = [\mathbf{\Gamma}_{\mathbf{u}}, \mathbf{\Gamma}_{\mathbf{y}}] \in \mathbb{R}^{n\times n(m+p)}$ *and* $\mathbf{L} \in \mathbb{R}^{n\times p}$ *with* $\mathbf{\Gamma}_{\mathbf{u}} \in \mathbb{R}^{n\times nm}$ *and* $\mathbf{\Gamma}_{\mathbf{y}} \in \mathbb{R}^{n\times np}$ *satisfying*

$$\mathbf{x}(t_i) = \mathbf{\Gamma}\eta(t_i) + \epsilon(t_i), \tag{12}$$

and

$$\dot{\mathbf{x}}(t_i) = \mathbf{\Gamma}\theta(t_i) + (\mathbf{A} - \mathbf{LC})\epsilon(t_i), \tag{13}$$

where $\mathbf{\Gamma}$ *is a parameterization matrix having full row rank and* $\epsilon(t_i) = e^{(\mathbf{A}-\mathbf{LC})t_i}\mathbf{x}(0) \in \mathbb{R}^n$. *Moreover, both the errors* $\mathbf{x}(t_i) - \mathbf{\Gamma}\eta(t_i)$ *and* $\dot{\mathbf{x}} - \mathbf{\Gamma}\theta(t_i)$ *converge to zero as the time* t_i *approaches infinity.* ◇

Proof: We show the first part of the observational continuous-time data that satisfies

$$\mathbf{x}(t) = \mathbf{\Gamma}\eta(t) + \epsilon(t). \tag{14}$$

The proof for Eq.**12** was originally given in (8) by extending the Luenberger observer (9). For completeness, we present an intuitive proof for Eq.**12** as follows from a different point of view. Recall the Luenberger observer(9)

$$\dot{\hat{\mathbf{x}}}(t) = \mathbf{A}\hat{\mathbf{x}}(t) + \mathbf{Bu}(t) + \mathbf{L}(\mathbf{y}(t) - \mathbf{C}\hat{\mathbf{x}}(t)), \tag{15}$$

where $\mathbf{L}$ is a matrix making $\mathbf{A} - \mathbf{LC}$ Hurwitz. By the Cayley-Hamilton theorem, given a matrix $\mathbf{A}$, one has

$$\begin{aligned} e^{(\mathbf{A}-\mathbf{LC})t} = {} & \alpha_{n-1}(t)(\mathbf{A} - \mathbf{LC})^{n-1} + \alpha_{n-2}(t)(\mathbf{A} - \mathbf{LC})^{n-2} \\ & + \ldots + \alpha_1(t)(\mathbf{A} - \mathbf{LC}) + \alpha_0(t)I, \end{aligned} \tag{16}$$

where $\alpha_i(t)$ for $i = 0, 1, \ldots, n-1$ are time-varying scalars and can be computed using the predetermined eigenvalues of $\mathbf{A} - \mathbf{LC}$. Solving Eq.**15** over the time interval $[0, t]$ yields

$$\begin{aligned} \hat{\mathbf{x}}(t) &= \int_0^t e^{(\mathbf{A}-\mathbf{LC})(t-\tau)}[\mathbf{Bu}(\tau) + \mathbf{Ly}(\tau)]d\tau + e^{(\mathbf{A}-\mathbf{LC})t}\hat{\mathbf{x}}(0) \\ &= \underbrace{\big[[\mathbf{B}\ \mathbf{L}], (\mathbf{A} - \mathbf{LC})[\mathbf{B}\ \mathbf{L}], \ldots, (\mathbf{A} - \mathbf{LC})^{n-1}[\mathbf{B}\ \mathbf{L}]\big]}_{\triangleq\ \bar{\mathbf{\Gamma}} \in \mathbb{R}^{n\times n(m+p)}} \underbrace{\begin{bmatrix} \int_0^t \alpha_0(t-\tau)[\mathbf{u}^T(\tau), \mathbf{y}^T(\tau)]^T d\tau \\ \int_0^t \alpha_1(t-\tau)[\mathbf{u}^T(\tau), \mathbf{y}^T(\tau)]^T(\tau)d\tau \\ \vdots \\ \int_0^t \alpha_{n-1}(t-\tau)[\mathbf{u}^T(\tau), \mathbf{y}^T(\tau)]^T(\tau)d\tau \end{bmatrix}}_{\triangleq\ \bar{\eta}(t) \in \mathbb{R}^{n(m+p)}} + e^{(\mathbf{A}-\mathbf{LC})t}\hat{\mathbf{x}}(0). \end{aligned} \tag{17}$$

Note that the matrix $\bar{\mathbf{\Gamma}} = \big[[\mathbf{B}\ \mathbf{L}], (\mathbf{A}-\mathbf{LC})[\mathbf{B}\ \mathbf{L}], \ldots, (\mathbf{A}-\mathbf{LC})^{n-1}[\mathbf{B}\ \mathbf{L}]\big]$ is called Kalman's controllability matrix related to the system $(\mathbf{A}-\mathbf{LC}, [\mathbf{B}\ \mathbf{L}])$. Note that the state feedback will not change the controllability. That is, the controllability of $(\mathbf{A}-\mathbf{LC}, [\mathbf{B}\ \mathbf{L}])$ is equivalent to that of $(\mathbf{A}, [\mathbf{B}\ \mathbf{L}])$. Now, under *Assumption 1*, Kalman's controllability matrix $\bar{\mathbf{\Gamma}}$ has the full row rank. Considering that d_i, for $i = 1, 2, \ldots, n$, in Eq.9 are predetermined by the designer, it is feasible to set d_i to be distinct. As a result, it follows from the Vandermonde matrix that $\alpha_i(t)$ in Eq.16 can be uniquely determined. With $\alpha_i(t)$ in Eq.16 ready, it is feasible to obtain $\bar{\eta}(t)$ in Eq.17 using the input and output data $\mathbf{x}(t)$ and $\mathbf{y}(t)$. This falls into the expression of Eq.14. At the sampling time t_i, Eq.14 turns into Eq.12. Note that $\mathbf{D}_\eta$ in Eq.9 is related to $\mathbf{A}-\mathbf{LC}$ through a nonsingular matrix, which is proved in (8) that the design of $\mathbf{D}_\eta$ determines the observer dynamics.

Next, we aim to prove the result in Eq.13. Taking the time derivative of Eq.12 yields

$$\begin{aligned}\dot{\mathbf{x}}(t) &= \mathbf{\Gamma}\dot{\eta}(t) + (\mathbf{A}-\mathbf{LC})\epsilon(t)\\ &= \mathbf{\Gamma}\begin{bmatrix}(\mathbf{I}_m \otimes \mathbf{D}_\eta)\eta_{\mathbf{u}}(t) + \mathbf{u}(t)\otimes\mathbf{b}\\ (\mathbf{I}_p \otimes \mathbf{D}_\eta)\eta_{\mathbf{y}}(t) + \mathbf{y}(t)\otimes\mathbf{b}\end{bmatrix} + (\mathbf{A}-\mathbf{LC})\epsilon(t),\end{aligned} \tag{18}$$

where the results in Eq.7 and Eq.8 are employed. Therefore, at the time instant $t = t_i$, one changes Eq.18 into Eq.13. This completes the proof. □

It follows from Eq.7 and Eq.8 that $\mathbf{x}(t_i)$ and $\dot{\mathbf{x}}(t_i)$ can be, respectively, reconstructed by $\mathbf{\Gamma}\eta(t_i)$ and $\mathbf{\Gamma}\theta(t_i)$ with the reconstruction errors converging to zero. This allows us to feedback both the state $\mathbf{x}(t_i)$ and its derivative $\dot{\mathbf{x}}(t_i)$ using the system input and output data. The idea of feedback the state derivative into the learning process makes this paper unique among the previous results in the field. The power of exploiting the state derivative further allows us to establish an analytical RL framework using the discrete-time rewards as illustrated in the next subsection.

B. Reinforcement Learning Using Discrete-time Rewards. From Eq.1, one has the following off-policy Bellman equation by subtracting and adding $\mathbf{BK}^k\mathbf{x}(t_i)$

$$\begin{aligned}2\mathbf{x}^T(t_i)\mathbf{P}^k\dot{\mathbf{x}}(t_i) &= 2\mathbf{x}^T(t_i)\mathbf{P}^k\mathbf{A}\mathbf{x}(t_i) + 2\mathbf{x}^T(t_i)\mathbf{P}^k\mathbf{B}[\mathbf{u}(t_i)+\mathbf{K}^k\mathbf{x}(t_i)] - 2\mathbf{x}^T(t_i)\mathbf{P}^k\mathbf{B}\mathbf{K}^k\mathbf{x}(t_i)\\ &= \mathbf{x}^T(t_i)(\mathbf{P}^k\mathbf{A}+\mathbf{A}^T\mathbf{P}^k)\mathbf{x}(t_i) + 2\mathbf{x}^T(t_i)\mathbf{P}^k\mathbf{B}[\mathbf{u}(t_i)+\mathbf{K}^k\mathbf{x}(t_i)] - 2\mathbf{x}^T(t_i)\mathbf{P}^k\mathbf{B}\mathbf{K}^k\mathbf{x}(t_i)\\ &= \mathbf{x}^T(t_i)[\mathbf{P}^k(\mathbf{A}-\mathbf{B}\mathbf{K}^k) + (\mathbf{A}-\mathbf{B}\mathbf{K}^k)^T\mathbf{P}^k]\mathbf{x}(t_i) + 2\mathbf{x}^T(t_i)\mathbf{P}^k\mathbf{B}[\mathbf{u}(t_i)+\mathbf{K}^k\mathbf{x}(t_i)]\\ &= -\mathbf{x}^T(t_i)[(\mathbf{K}^k)^T\mathbf{R}\mathbf{K}^k+\mathbf{Q}]\mathbf{x}(t_i) + 2\mathbf{x}^T(t_i)\mathbf{P}^k\mathbf{B}[\mathbf{u}(t_i)+\mathbf{K}^k\mathbf{x}(t_i)]\\ &= -\mathbf{x}^T(t_i)[(\mathbf{K}^k)^T\mathbf{R}\mathbf{K}^k+\mathbf{Q}]\mathbf{x}(t_i) + 2[\mathbf{u}(t_i)+\mathbf{K}^k\mathbf{x}(t_i)]^T\mathbf{R}\mathbf{K}^{k+1}\mathbf{x}(t_i),\end{aligned} \tag{19}$$

where $\mathbf{K}^{k+1} = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}^k$. Now, substituting Eq.12 and Eq.13 into Eq.19 yields

$$\begin{aligned}&2[\mathbf{\Gamma}\eta(t_i)+\epsilon(t_i)]^T\mathbf{P}^k[\mathbf{\Gamma}\theta(t_i)+(\mathbf{A}-\mathbf{LC})\epsilon(t_i)]\\ &= -[\mathbf{\Gamma}\eta(t_i)+\epsilon(t_i)]^T[(\mathbf{K}^k)^T\mathbf{R}\mathbf{K}^k+\mathbf{Q}][\mathbf{\Gamma}\eta(t_i)+\epsilon(t_i)]^T\\ &\quad + 2\big[\mathbf{u}(t_i)+\mathbf{K}^k[\mathbf{\Gamma}\eta(t_i)+\epsilon(t_i)]\big]^T\mathbf{R}\mathbf{K}^{k+1}[\mathbf{\Gamma}\eta(t_i)+\epsilon(t_i)],\end{aligned} \tag{20}$$

wherein both the system state and its derivative are now fedback into Eq.19 for solving the feedback gain $\mathbf{K}^{k+1}$. From

$$\begin{aligned}&\eta^T(t_i)\mathbf{\Gamma}^T\mathbf{P}^k\mathbf{\Gamma}\theta(t_i) - 2[\mathbf{u}(t_i)+\mathbf{K}^k\mathbf{\Gamma}\eta(t_i)]^T\mathbf{R}\mathbf{K}^{k+1}\mathbf{\Gamma}\eta(t_i)\\ &= -\eta^T(t_i)\mathbf{\Gamma}^T[(\mathbf{K}^k)^T\mathbf{R}\mathbf{K}^k+\mathbf{Q}]\mathbf{\Gamma}\eta(t_i) + e_k(t_i),\end{aligned} \tag{21}$$

The input-output signals, $\mathbf{x}(t)$ and $\mathbf{y}(t)$, determine the data sets of $\mathbf{U}$ and $\mathbf{Y}$

$$\mathbf{U} = \begin{bmatrix}\mathbf{u}^T(t_1)\\ \vdots\\ \mathbf{u}^T(t_i)\\ \vdots\\ \mathbf{u}^T(t_s)\end{bmatrix} = \begin{bmatrix}u_1(t_1) & u_2(t_1) & \ldots & u_m(t_1)\\ \vdots & \vdots & \ddots & \vdots\\ u_1(t_i) & u_2(t_i) & \ldots & u_m(t_i)\\ \vdots & \vdots & \ddots & \vdots\\ u_1(t_s) & u_2(t_s) & \ldots & u_m(t_s)\end{bmatrix},$$

$$\mathbf{Y} = \begin{bmatrix}\mathbf{y}^T(t_1)\\ \vdots\\ \mathbf{y}^T(t_i)\\ \vdots\\ \mathbf{y}^T(t_s)\end{bmatrix} = \begin{bmatrix}y_1(t_1) & y_2(t_1) & \ldots & y_p(t_1)\\ \vdots & \vdots & \ddots & \vdots\\ y_1(t_i) & y_2(t_i) & \ldots & y_p(t_i)\\ \vdots & \vdots & \ddots & \vdots\\ y_1(t_s) & y_2(t_s) & \ldots & y_p(t_s)\end{bmatrix},$$

and also the feedforward signals of $\eta(t)$ and $\theta(t)$ in Eq.12 and Eq.13, which further generate the following data sets $\mathbf{\Theta}_\eta \in \mathbb{R}^{s\times n(p+m)}$ and $\mathbf{\Theta}_\theta \in \mathbb{R}^{s\times n(p+m)}$ collected over several time instants as

$$\mathbf{\Theta}_\eta = \begin{bmatrix} \eta_1(t_1) & \eta_2(t_1) & \dots & \eta_{n(p+m)}(t_1) \\ \vdots & \vdots & \ddots & \vdots \\ \eta_1(t_i) & \eta_2(t_i) & \dots & \eta_{n(p+m)}(t_i) \\ \vdots & \vdots & \ddots & \vdots \\ \eta_1(t_s) & \eta_2(t_s) & \dots & \eta_{n(p+m)}(t_s) \end{bmatrix}$$

$$\mathbf{\Theta}_\theta = \begin{bmatrix} \theta_1(t_1) & \theta_2(t_1) & \dots & \theta_{n(p+m)}(t_1) \\ \vdots & \vdots & \ddots & \vdots \\ \theta_1(t_i) & \theta_2(t_i) & \dots & \theta_{n(p+m)}(t_i) \\ \vdots & \vdots & \ddots & \vdots \\ \theta_1(t_s) & \theta_2(t_s) & \dots & \theta_{n(p+m)}(t_s) \end{bmatrix}$$

with n, p, and m being the row dimensions of $\mathbf{x}(t)$, $\mathbf{y}(t)$, and $\mathbf{u}(t)$, respectively, and s denoting the number of time samples.

For ease of use, define the notation $\otimes$ as the Kronecker product operator. A collection of real-time system data are given as

$$\mathbf{\Theta}_{\eta\mathbf{u}} = [\eta(t_1)\otimes\mathbf{u}(t_1) \ \cdots \ \eta(t_i)\otimes\mathbf{u}(t_i) \ \cdots \ \eta(t_s)\otimes\mathbf{u}(t_s)]^T,$$

$$\mathbf{\Theta}_{\mathbf{yy}} = [\mathbf{y}(t_1)\otimes\mathbf{y}(t_1) \ \cdots \ \mathbf{y}(t_i)\otimes\mathbf{y}(t_i) \ \cdots \ \mathbf{y}(t_s)\otimes\mathbf{y}(t_s)]^T,$$

$$\mathbf{\Theta}_{\eta\mathbf{y}} = [\eta(t_1)\otimes\mathbf{y}(t_1) \ \cdots \ \eta(t_i)\otimes\mathbf{y}(t_i) \ \cdots \ \eta(t_s)\otimes\mathbf{y}(t_s)]^T,$$

$$\mathbf{\Theta}_{\eta\eta} = [\eta(t_1)\otimes\eta(t_1) \ \cdots \ \eta(t_i)\otimes\eta(t_i) \ \cdots \ \eta(t_s)\otimes\eta(t_s)]^T,$$

$$\mathbf{\Theta}_{\theta\eta} = [\theta(t_1)\otimes\eta(t_1) \ \cdots \ \theta(t_i)\otimes\eta(t_i) \ \cdots \ \theta(t_s)\otimes\eta(t_s)]^T,$$

and

$$\mathbf{\Theta}_{\mathbf{e}_k} = [e_k(t_1), e_k(t_2), \cdots, e_k(t_i), \cdots, e_k(t_{s-1}), e_k(t_s)]^T,$$

where the term $e_k(t_i)$ contains $\epsilon(t_i)$, $t_1 < t_2 < \ldots < t_{s-1} < t_s$ with t_1 and t_s being, respectively, time instants when the data collection starts and ends.

From Eq.20, one has the off-policy Bellman equation satisfying

$$\mathbf{\Theta}^k(\bar{\mathbf{K}}^k, \mathbf{\Theta}_{\theta\eta}, \mathbf{\Theta}_{\eta\eta}, \mathbf{\Theta}_{\eta\mathbf{y}}, \mathbf{\Theta}_{\eta\mathbf{u}}) \begin{bmatrix} \mathbf{vec}(\bar{\mathbf{P}}^k) \\ \mathbf{vec}(\bar{\mathbf{K}}^{k+1}) \end{bmatrix} = -\mathbf{\Theta}_r(\mathbf{U}_k, \mathbf{Y}) + \mathbf{\Theta}_{\mathbf{e}_k}, \quad [22]$$

where $\bar{\mathbf{P}}^k = \mathbf{\Gamma}^T\mathbf{P}^k\mathbf{\Gamma}$, $\bar{\mathbf{K}}^{k+1} = \mathbf{K}^k\mathbf{\Gamma}$, the notation $\mathbf{vec}(\cdot)$ denotes the vectorization operator, $\mathbf{\Theta}_r(\mathbf{U}_k, \mathbf{Y})$ denotes the data of the discrete-time reward defined as

$$\mathbf{\Theta}_r(\mathbf{U}_k, \mathbf{Y}) = \begin{bmatrix} | \\ r(\mathbf{u}_k(t_i), \mathbf{y}(t_i)) \\ | \end{bmatrix}, \quad [23]$$

with $r(\mathbf{u}_k(t_i), \mathbf{y}(t_i)) = \mathbf{\Theta}_{\eta\eta}\mathbf{vec}((\bar{\mathbf{K}}^k)^T\mathbf{R}\bar{\mathbf{K}}^k) + \mathbf{\Theta}_{\mathbf{yy}}\mathbf{vec}(\mathbf{Q})$ and $\mathbf{u}_k(t) = -\bar{\mathbf{K}}^k\eta(t)$ being an iterative decision law at the kth iteration step,

$$\mathbf{\Theta}^0(\bar{\mathbf{K}}^0, \mathbf{\Theta}_{\theta\eta}, \mathbf{\Theta}_{\eta\eta}, \mathbf{\Theta}_{\eta\mathbf{y}}, \mathbf{\Theta}_{\eta\mathbf{u}}) = [2\mathbf{\Theta}_{\theta\eta}, -2\mathbf{\Theta}_{\eta\mathbf{y}}(\mathbf{I}\otimes(\mathbf{K}_0)^T\mathbf{R}) - 2\mathbf{\Theta}_{\eta\mathbf{u}}(\mathbf{I}\otimes\mathbf{R})], \quad [24]$$

and

$$\mathbf{\Theta}^j(\bar{\mathbf{K}}^j, \mathbf{\Theta}_{\theta\eta}, \mathbf{\Theta}_{\eta\eta}, \mathbf{\Theta}_{\eta\mathbf{y}}, \mathbf{\Theta}_{\eta\mathbf{u}}) = [2\mathbf{\Theta}_{\theta\eta}, -2\mathbf{\Theta}_{\eta\eta}(\mathbf{I}\otimes(\bar{\mathbf{K}}^j)^T\mathbf{R}) - 2\mathbf{\Theta}_{\eta\mathbf{u}}(\mathbf{I}\otimes\mathbf{R})], \quad \text{for } j = 1, 2, \ldots \quad [25]$$

The term $\mathbf{\Theta}_{\mathbf{e}_k}$, contributed by $e^{(\mathbf{A}-\mathbf{LC})t_i}\mathbf{x}(0)$, is brought into Eq.22. Here, we follow the work of (8) and handle $\mathbf{\Theta}_{\mathbf{e}_k}$ by considering the following two cases: 1) a zero initial state case, and 2) a nonzero initial state case. It was shown in (8) that the nonzero case can be made to approximate the zero case by setting the starting time for data collection t_1 large. Therefore, in the remaining analysis, we only focus on the zero initial state case. That is, with the time t_1 being large enough, Eq.22 reduces to the following form

$$\mathbf{\Theta}^k(\bar{\mathbf{K}}^k, \mathbf{\Theta}_{\theta\eta}, \mathbf{\Theta}_{\eta\eta}, \mathbf{\Theta}_{\eta\mathbf{y}}, \mathbf{\Theta}_{\eta\mathbf{u}}) \begin{bmatrix} \mathbf{vec}(\bar{\mathbf{P}}^k) \\ \mathbf{vec}(\bar{\mathbf{K}}^{k+1}) \end{bmatrix} = -\mathbf{\Theta}_r(\mathbf{U}_k, \mathbf{Y}). \quad [26]$$

Now, given any symmetric matrix $\mathbf{X} \in \mathbb{R}^{m\times m}$, the notation $\mathbf{vecs}(\mathbf{X}) = [x_{11}, 2x_{12}, ..., 2x_{1m}, x_{22}, 2x_{23}, ..., 2x_{m-1,m}, x_{m,m}]^T \in \mathbb{R}^{\frac{1}{2}m(m+1)}$ denotes the half-vectorization of $\mathbf{X}$, where x_{ij} denotes an entry in the ith row and jth column of the matrix $\mathbf{X}$. For column vectors $\mathbf{v} \in \mathbb{R}^n$ and $\mathbf{x} \in \mathbb{R}^n$, $\mathbf{vecv}(\mathbf{v}, \mathbf{x}) = [v_1x_1, v_1x_2, ..., v_1x_n, v_2x_2, v_2x_3, ..., v_{n-1}x_n, v_nx_n]^T \in \mathbb{R}^{\frac{1}{2}n(n+1)}$.

Note that Eq.7 and Eq.8 share the common matrix $\mathbf{\Gamma}$, meaning that the matrix $\bar{\mathbf{P}}^k = \mathbf{\Gamma}^T\mathbf{P}^k\mathbf{\Gamma}$ is symmetric. Based on the half-vectorization of $\bar{\mathbf{P}}^k$, the term $\eta^T(t_i)\mathbf{\Gamma}^T\mathbf{P}^k\mathbf{\Gamma}\theta(t_i)$ in Eq.26 is further changed into

$$\begin{aligned}\eta^T(t_i)\mathbf{\Gamma}^T\mathbf{P}^k\mathbf{\Gamma}\theta(t_i) &= (\theta^T(t_i)\otimes\eta^T(t_i))\mathbf{vec}(\bar{\mathbf{P}}^k)\\ &= \mathbf{vecv}^T(\theta(t_i),\eta(t_i))\mathbf{vecs}(\bar{\mathbf{P}}^k).\end{aligned} \tag{27}$$

Given the vector $\mathbf{vecv}(\theta(t_i),\eta(t_i))$ in Eq.27, it is ready to define the following data set as

$$\bar{\mathbf{\Theta}}_{\theta\eta} = [\mathbf{vecv}(\theta(t_1),\eta(t_1)),\ldots,\mathbf{vecv}(\theta(t_i),\eta(t_i)),\ldots,\mathbf{vecv}(\theta(t_s),\eta(t_s))]^T. \tag{28}$$

From Eqs.27–28, Eq.26 is equivalent to

$$\bar{\mathbf{\Theta}}^k(\bar{\mathbf{K}}^k,\bar{\mathbf{\Theta}}_{\theta\eta},\mathbf{\Theta}_{\eta\eta},\mathbf{\Theta}_{\eta\mathbf{y}},\mathbf{\Theta}_{\eta\mathbf{u}})\begin{bmatrix}\mathbf{vecs}(\bar{\mathbf{P}}^k)\\ \mathbf{vec}(\bar{\mathbf{K}}^{k+1})\end{bmatrix} = -\mathbf{\Theta}_r(\mathbf{U}_k,\mathbf{Y}). \tag{29}$$

In the next subsection, we will give a sufficient condition for solving the gain matrix $\bar{\mathbf{K}}^{k+1}$ from Eq.29 and for obtaining the decision law satisfying the performance criterion in Eq.2.

C. Derivation of Unique Control Gain Learning Using Discrete-time Rewards. A verifiable condition is given as follows to solve the gain matrix $\bar{\mathbf{K}}^{k+1}$ from Eq.29.

***Lemma* 2** *The control gain matrix $\bar{\mathbf{K}}^{k+1}$ is uniquely solvable from Eq.29, if*

1. *The collected data satisfy* $\mathrm{rank}([\mathbf{\Theta}_{\eta\eta},\mathbf{\Theta}_{\eta\mathbf{u}}]) = (nm+np)\big(\frac{nm+np+1}{2}+m\big)$;
2. *The matrix $\mathbf{\Gamma}$ in Eq.13 is full row rank.* ◇

Proof: We prove this lemma by seeking a contradiction inspired by (10). Proving the uniqueness of the control gain matrix $\bar{\mathbf{K}}^{k+1}$ is equivalent to proving a unique solution $\mathbf{U}_v = 0$ to the equation

$$\bar{\mathbf{\Theta}}^k(\bar{\mathbf{K}}^k,\bar{\mathbf{\Theta}}_{\theta\eta},\mathbf{\Theta}_{\eta\eta},\mathbf{\Theta}_{\eta\mathbf{y}},\mathbf{\Theta}_{\eta\mathbf{u}})\mathbf{\Xi}_v = \mathbf{0}, \tag{30}$$

where $\mathbf{\Xi}_v = [\mathbf{M}_v,\mathbf{U}_v]^T$, $\mathbf{M}_v = \mathbf{vecs}(\mathbf{M}_m)$, and $\mathbf{U}_v = \mathbf{vec}(\mathbf{U}_m)$ with the dimensions satisfying $\mathbf{M}_m = (\mathbf{M}_m)^T \in \mathbb{R}^{n\times n}$ and $\mathbf{U}_m \in \mathbb{R}^{m\times n}$. Considering that $\mathbf{\Gamma}$ is full row rank, one has

$$\mathbf{Z}_m = (\mathbf{\Gamma}\mathbf{\Gamma}^T)^{-1}\mathbf{\Gamma}\mathbf{M}_m\mathbf{\Gamma}^T(\mathbf{\Gamma}\mathbf{\Gamma}^T)^{-1}. \tag{31}$$

From Eq.19, $\mathbf{\Theta}^k(\bar{\mathbf{K}}^k,\bar{\mathbf{\Theta}}_{\theta\eta},\mathbf{\Theta}_{\eta\eta},\mathbf{\Theta}_{\eta\mathbf{y}},\mathbf{\Theta}_{\eta\mathbf{u}})\mathbf{\Xi}_v = 0$ is changed as

$$[\mathbf{\Theta}_{\eta\eta},2\mathbf{\Theta}_{\eta\mathbf{u}}][\mathbf{vec}^T(\Omega_1),\mathbf{vec}^T(\Omega_2)]^T = \mathbf{0}, \tag{32}$$

where

$$\begin{aligned}\Omega_1 = {}& (\bar{\mathbf{K}}^k)^T(\mathbf{B}^T\mathbf{Z}_m\mathbf{\Gamma}-\mathbf{R}\mathbf{U}_m)+(\mathbf{B}^T\mathbf{Z}_m\mathbf{\Gamma}-\mathbf{R}\mathbf{U}_m)^T\bar{\mathbf{K}}^k\\ &+\mathbf{\Gamma}^T[(\mathbf{A}-\mathbf{B}\mathbf{K}^k)^T\mathbf{Z}_m+\mathbf{Z}_m(\mathbf{A}-\mathbf{B}\mathbf{K}^k)]\mathbf{\Gamma},\end{aligned} \tag{33}$$

and

$$\Omega_2 = \mathbf{B}^T\mathbf{Z}_m\mathbf{\Gamma}-\mathbf{R}\mathbf{U}_m. \tag{34}$$

Given that the condition of $\mathrm{rank}([\mathbf{\Theta}_{\eta\eta},\mathbf{\Theta}_{\eta\mathbf{u}}]) = (nm+np)\big(\frac{nm+np+1}{2}+m\big)$ is satisfied, Eq.32 leads to

$$\mathbf{vec}(\Omega_1) = \mathbf{0} \quad \text{and} \quad \mathbf{vec}(\Omega_2) = \mathbf{0}. \tag{35}$$

From Eq.33 and Eq.35, one has

$$\mathbf{U}_m = \mathbf{R}^{-1}\mathbf{B}^T\mathbf{Z}_m\mathbf{\Gamma}. \tag{36}$$

Now, under the condition of the full row rank of the matrix $\mathbf{\Gamma}$, substituting Eq.36 into Eq.32 yields

$$(\mathbf{A}-\mathbf{B}\mathbf{K}^k)^T\mathbf{Z}_m+\mathbf{Z}_m(\mathbf{A}-\mathbf{B}\mathbf{K}^k) = \mathbf{O}, \tag{37}$$

where $\mathbf{A}-\mathbf{B}\mathbf{K}^k$ is Hurwitz. Therefore, $\mathbf{Z}_m = \mathbf{O}$, based on which and Eq.36 one has $\mathbf{U}_m = \mathbf{O}$. This completes the proof. □

***Lemma* 2** provides two conditions under which the gain matrix $\bar{\mathbf{K}}^{k+1}$ can be uniquely solved. The first condition denotes the data richness, which requires that the system be stimulated. The exploration noise is usually introduced until the collected data satisfy the rank condition. The choice for the exploration noise may not be fixed and its necessary condition is that the system data should be bounded under the exploration noise. The second condition can be satisfied by constraining the system with controllability and observability, which turns out to be standard in the control community as specified in *Assumption 1*.

The computation in Eq.29 is carried out iteratively using the least squares technique by replacing $\bar{\mathbf{K}}^{k+1}$ from a previous step k with that from the current one until the stopping criterion is met, namely, the norm of the error of $\bar{\mathbf{K}}^{k+1}-\bar{\mathbf{K}}^k$ is small enough. Such an iteration ultimately leads to the unique optimal feedback gain.

In what follows, we aim to provide rigorous mathematical reasoning for the optimal feedback gain learning $\mathbf{K}_o^*$ through the computation of $\bar{\mathbf{K}}^{k+1}$ from Eq.29. As given in (1), the solution to Eq.29 relates to a model-based iterative algorithm for computing $\mathbf{P}^*$ from Eq.5, which was given in (11). The algorithm is recalled as follows.

***Lemma* 3** *(11) Let* $\mathbf{K}^0$ *be any stabilizing gain matrix so that* $\mathbf{A}-\mathbf{B}\mathbf{K}^0$ *is Hurwitz. Solve* $\mathbf{P}^k=(\mathbf{P}^k)^T$ *and* $\mathbf{K}^{k+1}$ *from the following equations*

$$\mathbf{P}^k(\mathbf{A}-\mathbf{B}\mathbf{K}^k)+\mathbf{C}^T\mathbf{Q}\mathbf{C}+(\mathbf{A}-\mathbf{B}\mathbf{K}^k)^T\mathbf{P}^k=-(\mathbf{K}^k)^T\mathbf{R}\mathbf{K}^k, \quad [38]$$

$$\mathbf{K}^{k+1}=\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}^k, \quad [39]$$

for $k=0,1,2,\ldots$. *Then, one has 1)* $\mathbf{A}-\mathbf{B}\mathbf{K}^k$ *is Hurwitz, 2)* $\mathbf{P}^*\leq\mathbf{P}^{k+1}\leq\mathbf{P}^k$, *and 3)* $\lim_{k\to\infty}\mathbf{K}^k=\mathbf{K}^*$. ◇

The overall RL framework of using the discrete-time reward is now summarized in the following theorem.

***Theorem* 1** *Given a dynamical system in Eq.1, an initial stabilizing policy in Eq.6, and the data of the discrete-time reward in Eq.23, if the control policy for Eq.1 is designed as* $u(t)=-\mathbf{K}_o^*\eta(t)$, *where the feedback gain matrix* $\mathbf{K}_o^*$ *is the converged* $\bar{\boldsymbol{K}}^{k+1}$ *from Eq.29 and the feedforward signal* $\eta(t)$ *is given in Eq.11, then the optimization problem in Eq.2 is solved, meaning that the analytical RL framework of the discrete-time reward is established.* ■

Proof: The proof is completed if the gain matrix $\bar{\mathbf{K}}^{k+1}$, recursively obtained from Eq.29, converges to $\mathbf{K}^*\mathbf{\Gamma}$, where $\mathbf{\Gamma}$ is given in Eq.13. Considering the stabilizing policy in Eq.6, the matrix $\mathbf{A}-\mathbf{B}\mathbf{K}^0$ is Hurwitz so that the gain matrix $\mathbf{K}^0$ is a well-defined stabilizing gain. Let $\bar{\mathbf{K}}^0=\mathbf{K}^0\mathbf{\Gamma}$, which can be used for triggering the iteration in Eq.29 with the discrete-time reward in Eq.23. By setting the initial sampling time t_1 large enough, solutions to Eq.22 converge to that of Eq.29. By ***Lemma* 2**, $\bar{\mathbf{K}}^{k+1}$ solved from Eq.29 converges to a unique solution, namely, the multiplication of $\mathbf{K}^{k+1}$ in Eq.39 and $\mathbf{\Gamma}$ in Eq.12. By ***Lemma* 3**, $\mathbf{K}^{k+1}$ converges to $\mathbf{K}^*$ as the iteration step k approaches to the infinity. With the converged $\bar{\mathbf{K}}^{k+1}$, the optimal control gain matrix $\mathbf{K}^*\mathbf{\Gamma}$ is approximated by $\bar{\mathbf{K}}^{k+1}$. Now, label the converged $\bar{\mathbf{K}}^{k+1}$ as $\mathbf{K}_o^*$. From Eq.12, the feedforward signal $\eta(t)$ in Eq.11 multiplied with the learned output-feedback gain $\mathbf{K}_o^*$ in Eq.29 converges to the state-feedback control policy $\mathbf{K}^*\mathbf{x}(t)$ in Eq.3. Therefore, the decision law $u(t)=-\mathbf{K}_o^*\eta(t)$ is learned and it solves the optimization problem in Eq.2. □

D. Robustness Analysis. The following is to show the robustness of the control policy obtained using the proposed framework.

For rejecting the disturbance, the robustness can be found in the inexact model-based result (see *Theorem 3.2* of (12)), which is recalled below.

Inexact Kleiman-Newton algorithm

Step 0: Choose X_0 such that $A-BX_0$ is stable, and a sequence of positive numbers $\{\eta_k\}$ such that $\eta_{k+1}<\eta_k$ for all k.

Step 1: Suppose $A_k=A-BB^TX_k$ and determine a solution X_{k+1} from the Lyapunov equation up to a residual R_k

$$X_{k+1}A_k+A_k^TX_{k+1}=-X_{k+1}BB^TX_k-C^TC+R_k \quad [40]$$

with

$$||R_k||\leq\eta_k||X_kA_k+A_k^TX_k-X_kBB^TX_k+C^TC||. \quad [41]$$

Step 2: Set $k\leftarrow k=1$ and return to Step 1.

It is proved in (12) the sequence of X_{k+1} given by Eq.40 converges to a case without the disturbance as long as the residual R_k satisfies Eq.41.

Recall the proof of ***Theorem* 1**, which indeed shows that our algorithm is equal to solving the following equation

$$\mathbf{P}^{k+1}(\mathbf{A}-\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}^k)+\mathbf{C}^T\mathbf{Q}\mathbf{C}+(\mathbf{A}-\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}^k)^T\mathbf{P}^{k+1}=-(\mathbf{B}^T\mathbf{P}^k)^T\mathbf{R}^{-1}\mathbf{B}^T\mathbf{P}^k. \quad [42]$$

By viewing $\mathbf{B}\mathbf{R}^{-1}\mathbf{B}^T$ in Eq.42 as BB^T in Eq.40 and $\mathbf{C}^T\mathbf{Q}\mathbf{C}$ as C^TC, one obtains that Eq.42 becomes Eq.40 under the condition that the residual R_k is zero. Now, it follows from *Theorem 3.2* of (12) that the proposed framework can reject the disturbance if it satisfies Eq.41.

References

1. FL Lewis, D Vrabie, VL Syrmos, *Optimal control.* (John Wiley & Sons), (2012).
2. E Todorov, MI Jordan, Optimal feedback control as a theory of motor coordination. *Nat. Neurosci.* **5**, 1226–1235 (2002).
3. R Shadmehr, S Mussa-Ivaldi, *Biological learning and control: how the brain builds representations, predicts events, and makes decisions.* (MIT Press), (2012).
4. YY Liu, JJ Slotine, AL Barabási, Controllability of complex networks. *Nature* **473**, 167–173 (2011).
5. J Ruths, D Ruths, Control profiles of complex networks. *Science* **343**, 1373–1376 (2014).
6. A Li, SP Cornelius, YY Liu, L Wang, AL Barabási, The fundamental advantages of temporal networks. *Science* **358**, 1042–1046 (2017).
7. C Chen, LF Lewis, B Li, Homotopic policy iteration-based learning design for unknown linear continuous-time systems. *Automatica* **138** (2022).
8. C Chen, L Xie, K Xie, FL Lewis, S Xie, Adaptive optimal output tracking of continuous-time systems via output-feedback-based reinforcement learning. *Automatica* **146** (2022).
9. G Tao, *Adaptive control design and analysis.* (John Wiley & Sons), (2003).
10. Y Jiang, ZP Jiang, *Robust Adaptive Dynamic Programming.* (John Wiley & Sons), (2017).

11. D Kleinman, On an iterative technique for Riccati equation computations. *IEEE Trans. Autom. Control.* **13**, 114–115 (1968).
12. F Feitzinger, T Hylla, EW Sachs, Inexact kleinman-newton method for riccati equations. *SIAM J. Matrix Anal. Appl.* **31**, 272–288 (2009).