ADP-Based Optimal Control of Linear Singularly Perturbed Systems With Uncertain Dynamics: A Two-Stage Value Iteration Method

Jianguo Zhao, Chunyu Yang, *Senior Member, IEEE*, Weinan Gao, *Senior Member, IEEE*, and Ju H. Park, *Senior Member, IEEE*

***Abstract*—We study the problem of adaptive dynamic programming (ADP) based on optimal control of linear singularly perturbed systems (SPSs) subject to completely unknown dynamics. Previous works on ADP-based optimal control of SPSs require that the fast dynamics of the system are either known *a priori* or unknown but asymptotically stable, and such results are limited to standard cases, i.e., the internal dynamics of their fast subsystems are nonsingular. In this brief, these conditions can be removed through a sequential procedure to design feedback gains of both fast and slow states in the framework of ADP. In this procedure, two optimal control problems are formulated for the prefeedback fast subsystem and the modified slow subsystem. Then, a data-driven two-stage value iteration algorithm is imposed to learn the optimal controller without using any system dynamics. Fundamentally different from existing works, an initial admissible control policy is no longer needed during learning. Finally, the effectiveness of the learning algorithm is proved by a simulation example.**

***Index Terms*—Adaptive dynamic programming (ADP), optimal control, singularly perturbed systems (SPSs), value iteration, data-driven control.**

I. Introduction

MANY dynamic systems in engineering fields can be modeled as singularly perturbed systems (SPSs) such as unmanned aerial vehicles [1], flexible manipulators [2], industrial processes [3], and electric circuits [4]. Due to the coupled fast and slow phenomena, it is known that there exist potential high dimensionality and numerical stiffness issues in the control and synthesis of SPSs [5], [6]. To alleviate these problems, a common method is to decompose SPSs into two reduced-order parts in different time scales, and then design a composite controller in terms of control problems of separated slow and fast subsystems. Such composite controller can stabilize the original full-order SPS for sufficiently small singular perturbation parameter [5]. However, this model reduction technology is highly dependent on the information of model parameters, no matter if decomposing systems or implementing composite controllers. In reality, accurate knowledge of system dynamics is hardly captured.

As a computational intelligence method, adaptive dynamic programming (ADP), an important branch in reinforcement learning, has been broadly studied by researchers from academia and industry [7]. When system dynamics are uncertain, ADP can employ data measurements to seek optimal controllers. Specifically, ADP can be leveraged to iteratively compute an approximate solution to the algebraic Riccati equation (ARE) relating to the optimal control problem of a linear system, without requiring any information of model parameters [8]. A comprehensive survey on ADP control and application was given in [9].

Since ADP can be independent of specific model parameters, extending ADP to reduced-order control of SPSs is of significance [2], [6], [10]. When the internal dynamic of a fast subsystem is Hurwitz, state feedback and output feedback based model-free reduced control strategies were proposed in [11] to solve linear quadratic regulation using ADP. In [12], the method was further extended to a robust control case. But such a reduced feedback controller could cause the system to be far from its desired performance because of ignoring fast dynamics [5]. When the slow dynamics are unknown and the fast dynamics are known, some researchers posed ADP-based reduced-order composite control approaches for linear [13], interconnected [14], Markov jump [15], and nonlinear SPSs [16]. It should be pointed out, however, that these approaches are only available for standard SPSs, i.e., the internal dynamics of their fast subsystems are nonsingular, which are used to decompose system. Therefore, a natural question to ask is: how can we develop an ADP algorithm for both standard and nonstandard SPSs to overcome the aforementioned limitations (for instance, the fast dynamics

Manuscript received 27 April 2023; accepted 15 May 2023. Date of publication 18 May 2023; date of current version 8 December 2023. The work of Chunyu Yang was supported by the National Natural Science Foundation of China under Grant 62073327 and Grant 62273350. The work of Ju H. Park was supported by the National Research Foundation of Korea Grant funded by the Korea Government (Ministry of Science and ICT) under Grant 2019R1A5A8080290. This brief was recommended by Associate Editor C.-T. Cheng. *(Corresponding authors: Chunyu Yang; Ju H. Park.)*

Jianguo Zhao is with the School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221116, China, and also with the Department of Electrical Engineering, Yeungnam University, Gyeongsan 38541, South Korea (e-mail: jianguozhao@cumt.edu.cn).

Chunyu Yang is with the Engineering Research Center of Intelligent Control for Underground Space, Ministry of Education, and the School of Information and Control Engineering, China University of Mining and Technology, Xuzhou 221116, China (e-mail: chunyuyang@cumt.edu.cn).

Weinan Gao is with the State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang 110819, China (e-mail: gaown@mail.neu.edu.cn).

Ju H. Park is with the Department of Electrical Engineering, Yeungnam University, Gyeongsan 38541, South Korea (e-mail: jessie@ynu.ac.kr).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSII.2023.3277528.

Digital Object Identifier 10.1109/TCSII.2023.3277528

are either known *a priori* or unknown but asymptotically stable)?

Motivated by the above-mentioned question, we propose an alternative data-driven ADP control approach for linear SPSs, whose structures have no standard limitation. In particular, two optimal control problems in a sequential procedure are formulated to design feedback gains of both fast and slow states. The fast component is designed first to stabilize a fast mode, and then the slow component is to stabilize a slow mode of the prefeedback system. We provide sufficient conditions for the existence of these solutions. In the framework of ADP, a value iteration-based data-driven two-stage learning algorithm is leveraged to seek the target policy without using any system dynamics, which does not require an initial admissible control policy during learning. To the best of our knowledge, value iteration [8] is generalized to SPSs for the first time in this brief. Moreover, we prove the closed-loop stability under mild conditions. We believe that the proposed scheme is helpful to facilitate the development of ADP-based data-driven control for SPSs with uncertain dynamics.

II. Problem Formulation and Preliminaries

Consider the linear SPS

$$\dot{x} = A_{11}x + A_{12}z + B_1u \tag{1}$$

$$\varepsilon\dot{z} = A_{21}x + A_{22}z + B_2u \tag{2}$$

where $x \in \mathbb{R}^{n_1}$ and $z \in \mathbb{R}^{n_2}$ are the slow and fast states, respectively, $u \in \mathbb{R}^m$ is the control input, $\varepsilon > 0$ is the singular perturbation parameter which denotes the degree of time-scale separation between x and z, $A_{ij} \in \mathbb{R}^{n_i\times n_j}$ and $B_i \in \mathbb{R}^{n_i\times m}$ $(i, j = 1, 2)$ are constant matrices.

In this brief, when A_{ij} and B_i $(i, j = 1, 2)$ are unknown, we want to find an optimal feedback controller

$$u = -F_xx - F_zz \tag{3}$$

that asymptotically stabilizes (1), (2) for all $\varepsilon \in (0, \varepsilon_0]$. Note that the structure of the conventional composite controller contains the information of $A_{22}^{-1}B_2$ and $A_{22}^{-1}A_{21}$ [5], [13], [14], [15]. Although the authors of [2] and [10] tried to identify the fast dynamics by setting $\varepsilon = 0$, the parameter perturbations from $\varepsilon \neq 0$ must result in estimation errors. Unlike existing results, we shall develop ADP algorithm to directly learn the feedback gains F_z and F_x in a sequential procedure, so as to realize the model-free control of SPSs. We assume that ε is known whose rationale has been pointed out in [6, Remark 1].

At the outset, we define

$$u_z \triangleq -F_zz \tag{4}$$

$$u_x \triangleq -F_xx = u - u_z. \tag{5}$$

By replacing u with a prefeedback control u_z in (1), (2), the decoupled fast subsystem is given by [10]

$$\varepsilon\dot{z}_{zf} = A_{22}z_{zf} + B_2u_{zf} \tag{6}$$

where $z_{zf} \in \mathbb{R}^{n_2}$ and $u_{zf} \in \mathbb{R}^m$ are the fast subsystem state and control input for (1), (2) under $u = u_z$. We first formulate an optimization problem for designing the feedback gain F_z.

Problem 1: For fast subsystem (6), find the optimal feedback gain F_z such that the control input $u_{zf} = -F_zz_{zf}$ minimizes the performance index

$$J_z = \int_0^\infty (z_{zf}^TQ_zz_{zf} + u_{zf}^TRu_{zf})dt \tag{7}$$

where $Q_z \succ 0$ and $R_z \succ 0$ are the predefined weight matrices.

A routine assumption in SPSs is made for control designs.

Assumption 1: The pair (A_{22}, B_2) is stabilizable.

When the prefeedback input $u_z = -F_zz$ obtained from Problem 1 is applied to (1), (2), by (5), we have

$$\dot{x} = A_{11}x + (A_{12} - B_1F_z)z + B_1u_x \tag{8}$$

$$\varepsilon\dot{z} = A_{21}x + (A_{22} - B_2F_z)z + B_2u_x. \tag{9}$$

It is shown in [11] that the reduced control input $u_x = -F_xx$ can be designed based on the slow subsystem of the system (8), (9) to stabilize the full system (8), (9) for all $\varepsilon \in (0, \varepsilon_0]$ if $(A_{22} - B_2F_z)$ is Hurwitz. Setting $\varepsilon = 0$ and then using (9) to eliminate z from (8), the modified slow subsystem is expressed as

$$\dot{x}_{xs} = A_sx_{xs} + B_su_{xs} \tag{10}$$

where $x_{xs} \in \mathbb{R}^{n_1}$, $u_{xs} \in \mathbb{R}^m$ are the state and control input respectively, and $A_s = A_{11} - (A_{12} - B_1F_z)(A_{22} - B_2F_z)^{-1}A_{21}$ and $B_s = B_1 - (A_{12} - B_1F_z)(A_{22} - B_2F_z)^{-1}B_2$. It is worth emphasizing that if the solution to Problem 1 exists, then $(A_{22} - B_2F_z)$ must be a nonsingular matrix by optimal control theory [17]. The modified slow system (10) derived in this brief can circumvent the limitation on nonsingularity of A_{22} in the past literature [10], [11], [12], [13], [14], [15]. Consequently, our results shall be available for both standard and nonstandard SPSs. To design F_x, we formulate second optimization problem.

Problem 2: For modified slow subsystem (10), find the optimal feedback gain F_x such that the control input $u_{xs} = -F_xx_{xs}$ minimizes the performance index

$$J_x = \int_0^\infty (x_{xs}^TQ_xx_{xs} + u_{xs}^TRu_{xs})dt \tag{11}$$

where $Q_x \succ 0$ and $R_x \succ 0$ are the predefined weight matrices.

The following lemma is recalled to reveal the relationship between x and x_{xs}, which is used in our learning algorithm.

Lemma 1 [5]: Consider the full system (8), (9) and the modified slow subsystem (10). If $(A_{22} - B_2F_z)$ is Hurwitz, then there exists a scalar $\varepsilon_0 > 0$ such that for any $\varepsilon \in (0, \varepsilon_0]$, the trajectories $x(t)$ and $x_{xs}(t)$ satisfy

$$x_{xs}(t) = x(t) + O_x(\varepsilon) \tag{12}$$

for all finite $t \geq 0$.

III. Main Results

In this section, in the absence of knowledge of A_{ij} and B_i $(i, j = 1, 2)$, we devise a data-driven value iteration algorithm to find the controller (3) optimized from Problems 1 and 2.

A. Solutions to Problems 1 and 2

According to standard LQR theory [17], under Assumption 1, the optimal feedback gain of Problem 1 associated with (6) and (7) is

$$F_z = R^{-1}B_2^T P_z \tag{13}$$

with $P_z \succ 0$ being the unique solution to the ARE

$$A_{22}^T P_z + P_z A_{22} + Q_z - P_z B_2 R^{-1} B_2^T P_z = 0. \tag{14}$$

In what follows, one rank condition, which is independent of the predesign gain matrix F_z, is given to guarantee the stabilizability of (10).

Lemma 2: Let Assumption 1 hold. The pair (A_s, B_s) is stabilizable if

$$\operatorname{rank}\begin{bmatrix} sI_{n_1} - A_{11} & -A_{12} & B_1 \\ -A_{21} & -A_{22} & B_2 \end{bmatrix} = n + m, \ \forall s \in \mathbb{C}^+. \tag{15}$$

Proof: It is well known that $\bar{A}_{22} = A_{22} - B_2 F_z$ is always Hurwitz for any F_z in (13). Then by (15) we have

$$\begin{aligned}
&\operatorname{rank}\begin{bmatrix} sI_{n_1} - A_{11} & -A_{12} & B_1 \\ -A_{21} & -A_{22} & B_2 \end{bmatrix} \\
&= \operatorname{rank}\begin{bmatrix} sI_{n_1} - A_{11} & -A_{12} + B_1 F_z & B_1 \\ \bar{A}_{22}^{-1} A_{21} & I_{n_2} & -\bar{A}_{22}^{-1} B_2 \end{bmatrix} \\
&= \operatorname{rank}\begin{bmatrix} sI_{n_1} - A_s & 0 & B_s \\ \bar{A}_{22}^{-1} A_{21} & I_{n_2} & -\bar{A}_{22}^{-1} B_2 \end{bmatrix} \\
&= \operatorname{rank}\begin{bmatrix} sI_{n_1} - A_s & 0 & B_s \\ 0 & I_{n_2} & 0 \end{bmatrix} = n + m, \ \forall s \in \mathbb{C}^+
\end{aligned}$$

which implies (A_s, B_s) being stabilizable. ∎

Remark 1: By analogy to the proofs of Lemma 2, when A_{22} is nonsingular, it is not difficult to derive

$$\begin{aligned}
&\operatorname{rank}\begin{bmatrix} sI_{n_1} - A_{11} & -A_{12} & B_1 \\ -A_{21} & -A_{22} & B_2 \end{bmatrix} \\
&= \operatorname{rank}\begin{bmatrix} sI_{n_1} - \bar{A}_s & 0 & \bar{B}_s \\ 0 & I_{n_2} & 0 \end{bmatrix} = n + m, \ \forall s \in \mathbb{C}^+
\end{aligned}$$

with $\bar{A}_s = A_{11} - A_{12}A_{22}^{-1}A_{21}$ and $\bar{B}_s = B_1 - A_{12}A_{22}^{-1}B_2$. This means that the modified slow subsystem (10) and the inherent slow subsystem of (1), (2) have the same condition on stabilizability.

Likewise, under the conditions in Lemma 2, the optimal feedback gain of Problem 2 associated with (10) and (11) is

$$F_x = R^{-1}B_s^T P_x \tag{16}$$

with $P_x \succ 0$ being the unique solution to the ARE

$$A_s^T P_x + P_x A_s + Q_x - P_x B_s R^{-1} B_s^T P_x = 0. \tag{17}$$

In the following proposition, we show that the controller (3) obtained from (13) and (16) is an optimal input for system (1), (2) with one performance index.

Proposition 1: Consider (1), (2) and let the conditions in Lemma 2 hold. Then, the controller (3) obtained from (13) and (16) minimizes the performance index

$$J_\varepsilon = \int_0^\infty (X^T Q X + u^T R u) dt. \tag{18}$$

Proof: We rewrite (1), (2) in the more compact form

$$\dot{X} = A_\varepsilon X + B_\varepsilon u \tag{19}$$

where $X = \begin{bmatrix} x \\ z \end{bmatrix}$, $A_\varepsilon = \begin{bmatrix} A_{11} & A_{12} \\ \varepsilon^{-1}A_{21} & \varepsilon^{-1}A_{22} \end{bmatrix}$, and $B_\varepsilon = \begin{bmatrix} B_1 \\ \varepsilon^{-1}B_2 \end{bmatrix}$. Let

$$P_\varepsilon = \begin{bmatrix} P_1 & 0 \\ 0 & \varepsilon P_z \end{bmatrix} \succ 0 \tag{20}$$

where $B_1^T P_1 - B_s^T P_x = 0$. By (14), it is checkable that P_ε is the solution to the ARE

$$A_\varepsilon^T P_\varepsilon + P_\varepsilon A_\varepsilon + Q - P_\varepsilon B_\varepsilon R^{-1} B_\varepsilon^T P_\varepsilon = 0 \tag{21}$$

where $Q = \begin{bmatrix} Q_1 & Q_2 \\ Q_2^T & Q_z \end{bmatrix}$ with

$$\begin{aligned}
Q_1 &= P_1 B_1 R^{-1} B_1^T P_1 - A_{11}^T P_1 - P_1 A_{11} \\
Q_2 &= P_1 B_1 R^{-1} B_2^T P_z - A_{21}^T P_z - P_1 A_{12}.
\end{aligned}$$

By [17], the solution to the standard LQR problem associated with (18) and (19) is determined by $u = -FX$, where $F = R^{-1}B_\varepsilon^T P_\varepsilon$. It is not difficult to derive $u = -FX = -F_x x - F_z z$. This completes the proof. ∎

B. Data-Driven ADP Scheme

Before proceeding, for $\mathscr{A} = [a_1, a_2, \ldots, a_m] \in \mathbb{R}^{n\times m}$, $\mathscr{B} = [b_{ij}] \in \mathbb{R}^{n\times n}$, and $\mathscr{C} = [c_i] \in \mathbb{R}^n$, we define

$$\begin{aligned}
\operatorname{vec}(\mathscr{A}) &\triangleq [a_1^T, a_2^T, \ldots, a_n^T]^T \in \mathbb{R}^{nm} \\
\operatorname{ves}(\mathscr{B}) &\triangleq [b_{11}, 2b_{12}, \ldots, 2b_{1n}, b_{22}, 2b_{23}, \ldots, 2b_{n-1,n}, b_{nn}]^T \\
&\quad \in \mathbb{R}^{0.5n(n+1)} \\
\operatorname{vev}(\mathscr{C}) &\triangleq [c_1^2, c_1c_2, \ldots, c_1c_n, c_2^2, c_2c_3, \ldots, c_{n-1}c_n, c_n^2]^T \\
&\quad \in \mathbb{R}^{0.5n(n+1)}.
\end{aligned}$$

For any two vectors α and β, we further define

$$\begin{aligned}
\mathrm{T}_\alpha &= [\int_{t_0}^{t_1} \operatorname{vev}(\alpha) d\tau, \int_{t_1}^{t_2} \operatorname{vev}(\alpha) d\tau, \ldots, \int_{t_{q-1}}^{t_q} \operatorname{vev}(\alpha) d\tau]^T \\
\Pi_{\alpha\beta} &= [\int_{t_0}^{t_1} \alpha \otimes \beta d\tau, \int_{t_1}^{t_2} \alpha \otimes \beta d\tau, \ldots, \int_{t_{q-1}}^{t_q} \alpha \otimes \beta d\tau]^T \\
\Pi_{\alpha\alpha} &= [\alpha \otimes \alpha|_{t_0}^{t_1}, \alpha \otimes \alpha|_{t_1}^{t_2}, \ldots, \alpha \otimes \alpha|_{t_{q-1}}^{t_q}]^T
\end{aligned}$$

where $0 \le t_0 < t_1 < \cdots < t_{q-1} < t_q$ are constants, q is a positive integer, and $\otimes$ denotes the Kronecker product operator.

Inspired by [8], the model-based value iteration algorithm is leveraged to solve the ARE (14) by

$$\begin{aligned}
P_z^{k+1} &= h_k(A_{22}^T P_z^k + P_z^k A_{22} + Q_z - P_z^k B_2 R^{-1} B_2^T P_z^k) + P_z^k \\
F_z^k &= R^{-1} B_2^T P_z^k, \ k = 0, 1, 2, \ldots
\end{aligned} \tag{22}$$

where $P_z^0 \succ 0$ and h_k denotes the step size satisfying $h_k > 0$, $\sum_{k=0}^\infty h_k = \infty$, and $\sum_{k=0}^\infty h_k^2 < \infty$. Based on (2) and (22), we have

$$\begin{aligned}
&z^T(t + \Delta t)\varepsilon P_z^k z(t + \Delta t) - z^T(t)\varepsilon P_z^k z(t) \\
&= \int_t^{t+\Delta t} [z^T \mathcal{H}_z^k z + 2x^T A_{21}^T P_z^k z + 2(Ru)^T F_z^k z] d\tau
\end{aligned} \tag{23}$$

where $\mathcal{H}_z^k = A_{22}^T P_z^k + P_z^k A_{22}$ and Δt is the sampling interval. Using Kronecker product, (23) is reformulated as the algebraic matrix equation

$$\mathcal{X}_z \begin{bmatrix} \text{ves}(\mathcal{H}_z^k) \\ \text{vec}(A_{21}^T P_z^k) \\ \text{vec}(F_z^k) \end{bmatrix} = \mathcal{Y}_z^k \tag{24}$$

where $\mathcal{X}_z = [\mathrm{T}_z, 2\Pi_{zx}, 2\Pi_{z(Ru)}]$ and $\mathcal{Y}_z^k = \varepsilon \Pi_{zz} \text{vec}(P_z^k)$. As shown in [18], we choose u to satisfy the rank condition

$$\text{rank}(\mathcal{X}_z) = \frac{n_2(n_2+1)}{2} + n_1 n_2 + n_2 m \tag{25}$$

then the uniqueness of both $\mathcal{H}_z^k$ and F_z^k to (24) is guaranteed.

In what follows, we find the slow state gain F_x using system measurements. The ARE (17) can be solved by the model-based value iteration algorithm

$$\begin{aligned} P_x^{l+1} &= h_l(A_s^T P_x^l + P_x^l A_x + Q_x - P_x^l B_s R^{-1} B_s^T P_x^l) + P_x^l \\ F_x^l &= R^{-1} B_s^T P_x^l, \ \ l = 0, 1, 2, \ldots \end{aligned} \tag{26}$$

where $P_x^0 \succ 0$ and h_l denotes the step size satisfying $h_l > 0$, $\sum_{l=0}^{\infty} h_l = \infty$, and $\sum_{l=0}^{\infty} h_l^2 < \infty$. Based on (10) and (26), we have

$$\begin{aligned} &x_{xs}^T(t+\Delta t) P_x^l x_{xs}(t+\Delta t) - x_{xs}^T(t) P_x^l x_{xs}(t) \\ &\quad = \int_t^{t+\Delta t} [x_{xs}^T \mathcal{H}_x^l x_{xs} + 2(Ru_{xs})^T F_x^l x_{xs}] d\tau \end{aligned} \tag{27}$$

where $\mathcal{H}_x^l = A_s^T P_x^l + P_x^l A_s$. Since x_{xs} and u_{xs} are the state and input of the decomposed virtual subsystem (10), these signals cannot be measured from the full-order plant (8), (9). Thanks to Lemma 1, we replace x_{xs} and u_{xs} with x and u_x during learning, which has also been commonly adopted in the past literature [2], [3], [10], [11], [12], [13], [14], [15], [16]. Then, we rewrite (27) as

$$x^T(t+\Delta t) P_x^l x(t+\Delta t) - x^T(t) P_x^l x(t) = \int_t^{t+\Delta t} [x^T \mathcal{H}_x^l x + 2(Ru_x)^T F_x^l x] d\tau \tag{28}$$

which implies the algebraic matrix equation

$$\mathcal{X}_x \begin{bmatrix} \text{ves}(\mathcal{H}_x^l) \\ \text{vec}(F_x^l) \end{bmatrix} = \mathcal{Y}_x^l \tag{29}$$

where $\mathcal{X}_x = [\mathrm{T}_x, 2\Pi_{x(Ru_x)}]$ and $\mathcal{Y}_x^l = \Pi_{xx} \text{vec}(P_x^l)$. Similarly, u is chosen to satisfy the rank condition

$$\text{rank}(\mathcal{X}_x) = \frac{n_1(n_1+1)}{2} + n_1 m \tag{30}$$

then the uniqueness of both $\mathcal{H}_x^l$ and F_x^l to (29) is guaranteed.

We define the collection of bounded nonempty sets $\{\mathcal{B}_b^i\}_{b=0}^{\infty}$ as $\mathcal{B}_b^i \subseteq \mathcal{B}_{b+1}^i$, $b \in \mathbb{Z}_+$, and $\lim_{b\to\infty} \mathcal{B}_b^i = \mathbb{P}^{n_i}$, where $i = 1, 2$ and $\mathbb{P}^n$ denotes the set of $n \times n$ symmetric and positive definite matrices. Now, the data-driven value iteration-based ADP algorithm to learn the target controller (3) optimized from Problems 1 and 2 is presented in Algorithm 1. Clearly, there exist two learning stages. In the first stage, the measurements of x, z, and u are collected until (25) holds for computing the gain F_z. Then, based on the learned F_z^k, the measurements of x and u_x are collected until (30) holds for computing the gain F_x in the second stage. Note that these rank conditions can be ensured in the spirit of exploration noise [8], [18].

Algorithm 1 Two-Stage Value Iteration Algorithm

1: Apply a locally essentially bounded input u to excite (1), (2) such that (25) holds. Select a threshold $\sigma_z > 0$ and a symmetric $P_z^0 \succ 0$
2: $k, b \leftarrow 0$
3: **repeat**
4: Solve $\mathcal{H}_z^k$ and F_z^k from (24)
5: $\tilde{P}_z^{k+1} \leftarrow P_z^k + h_k(\mathcal{H}_z^k + Q_z - (F_z^k)^T R F_z^k)$
6: **if** $\tilde{P}_z^{k+1} \notin \mathcal{B}_b^1$ **then**
7: $P_z^{k+1} \leftarrow P_z^0$, $b \leftarrow b+1$
8: **else** $P_z^{k+1} \leftarrow \tilde{P}_z^{k+1}$
9: **end if**
10: $k \leftarrow k+1$
11: **until** $||P_z^k - P_z^{k-1}||/h_k < \sigma_z$
12: Apply a locally essentially bounded input $u = u_x - F_z^k z$ to excite (1), (2) such that (30) holds. Select a threshold $\sigma_x > 0$ and a symmetric $P_x^0 \succ 0$
13: $l, b \leftarrow 0$
14: **repeat**
15: Solve $\mathcal{H}_x^l$ and F_x^l from (29)
16: $\tilde{P}_x^{l+1} \leftarrow P_x^l + h_l(\mathcal{H}_x^l + Q_x - (F_x^l)^T R F_x^l)$
17: **if** $\tilde{P}_x^{l+1} \notin \mathcal{B}_b^2$ **then**
18: $P_x^{l+1} \leftarrow P_x^0$, $b \leftarrow b+1$
19: **else** $P_x^{l+1} \leftarrow \tilde{P}_x^{l+1}$
20: **end if**
21: $l \leftarrow l+1$
22: **until** $||P_x^l - P_x^{l-1}||/h_l < \sigma_x$
23: Use $u = -F_x^l x - F_z^k z$ as the optimal feedback controller

Theorem 1: Let Assumption 1 and (15) hold. For Algorithm 1, $\lim_{k\to\infty} F_z^k = F_z$ and $\lim_{k,l\to\infty} F_x^l = F_x + O_{F_x}(\varepsilon)$.

Proof: Algorithm 1 is deduced from the model-based value iteration algorithms (22) and (26). For stage-one learning, by [8], $\lim_{k\to\infty} F_z^k = F_z$ on the basis of its equivalence. Since $A_{22} - B_2 F_z^k$ is Hurwitz for $k \to \infty$, by Lemma 1, there exists $x_{xs}(t) = x(t) + O_x(\varepsilon)$ for $t \geq 0$. Then, by adopting the same line of proofs as in [10] and [11], it can be easily shown $\lim_{k,l\to\infty} F_x^l = F_x + O_{F_x}(\varepsilon)$ during stage-two learning. ■

Theorem 2: Under the conditions of Theorem 1, let $u = -(F_x + O_{F_x}(\varepsilon))x - F_z z$ be the target controller obtained from Algorithm 1 with $||O_{F_x}(\varepsilon)|| \leq d$. If we choose $Q_x > I_{n_1}$ and $R^{-1} \geq d^2 I_m$, then u asymptotically stabilizes system (1), (2).

Proof: Substituting u into (1), (2) yields

$$\dot{x} = (A_{11} - B_1 F_x + B_1 O_{F_x}(\varepsilon))x + (A_{12} - B_1 F_z) z \tag{31}$$

$$\varepsilon \dot{z} = (A_{21} - B_2 F_x + B_2 O_{F_x}(\varepsilon))x + (A_{22} - B_2 F_z) z \tag{32}$$

whose fast subsystem is stable because of $\bar{A}_{22}$ being Hurwitz. The stability of closed-loop system (31), (32) is guaranteed if its slow subsystem

$$\dot{x}_{xs} = [A_s - B_s F_x + B_s O_{F_x}(\varepsilon)] x_{xs} \tag{33}$$

is stable [5]. Define the Lyapunov candidate $V_{xs} = x_{xs}^T P_x x_{xs}$. By (16) and (17), along the trajectories of (33), it follows that

$$\begin{aligned} \dot{V}_{xs} &= -x_{xs}^T (Q_x + F_x^T R F_x) x_{xs} + 2 x_{xs}^T O_{F_x}^T(\varepsilon) B_s^T P_x x_{xs} \\ &= -x_{xs}^T [Q_x - O_{F_x}^T(\varepsilon) R O_{F_x}(\varepsilon)] x_{xs} \\ &\quad - x_{xs}^T [F_x - O_{F_x}(\varepsilon)]^T R [F_x - O_{F_x}(\varepsilon)] x_{xs} \leq 0 \end{aligned}$$

which implies that (33) is asymptotically stable. ■

Remark 2: In Algorithm 1, the dimensions of unknown parameters in (24) and (29) are $c_1 = n_2(n_2+1)/2 + n_1 n_2 + n_2 m$ and $c_2 = n_1(n_1+1)/2 + n_1 m$, respectively. As a contrast, the unknown parameters, for the full-order value iteration, in (17) of [8] is $c = (n_1+n_2)(n_1+n_2+1)/2 + (n_1+n_2)m$. Note

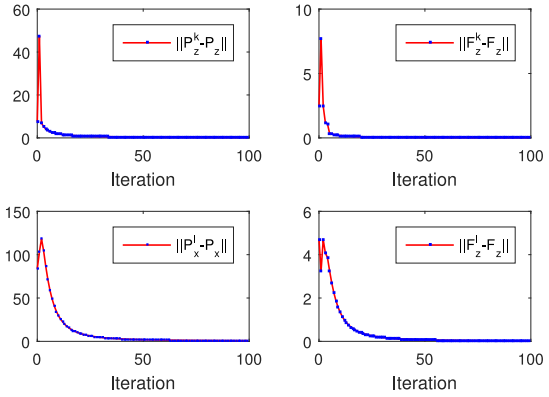

Fig. 1. Convergence of P_z, F_z, P_x, and F_x.

that the computational complexity of inverting a p-dimensional matrix is $\mathcal{O}(p^{2.376})$ [19]. Since $c = c_1 + c_2$, the proposed two-stage learning Algorithm 1 has much lower computational complexity per iteration than [8, Algorithm 2].

IV. Simulation

In this section, a benchmark SPS is considered as a numerical example to confirm Algorithm 1. The model matrices of the form (1), (2) are as follows [5], [11]

$$A_{11} = \begin{bmatrix} 0 & 0.4 \\ 0 & 0 \end{bmatrix},\ A_{12} = \begin{bmatrix} 0 & 0 \\ 0.345 & 0 \end{bmatrix},\ B_1 = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$
$$A_{21} = \begin{bmatrix} 0 & -0.524 \\ 0 & 0 \end{bmatrix},\ A_{22} = \begin{bmatrix} -0.465 & 0.262 \\ 0 & -1 \end{bmatrix},\ B_2 = \begin{bmatrix} 0 \\ 1 \end{bmatrix}$$

with $\varepsilon = 0.001$. We set weight matrices in Problems 1 and 2 as $Q_z = Q_x = 10I_2$ and $R = 1$. In Algorithm 1, we choose the parameters

$$\mathcal{B}_b^i = \{P \in \mathbb{P}^2 : ||P|| \le 210(b+1)\},\ \forall b = 0, 1, 2, \ldots$$

for $i = 1, 2$, $\sigma_z = \sigma_x = 0.001$, $h_k = 1/k$, and $h_l = 5.2/l$. The data for the stage-one learning is collected on $[0, 2]$ and the stage-two learning is $[2, 4]$ to guarantee (25) and (30) with the initial values $x(0) = [1, 1]^T$ and $z(0) = [-1, -1]^T$.

The simulation tests are performed in 64-bit MATLAB and run on a 1.80 GHZ, 4 GB of RAM, Intel Core i5 computer. Fig. 1 shows the trajectories of $||P_z^k - P_z||$, $||F_z^k - F_z||$, $||P_x^l - P_x||$, and $||F_x^l - F_x||$ during learning. The convergent kernel matrices are given below

$$P_z = \begin{bmatrix} 10.0402 & 0.6869 \\ 0.6869 & 2.3663 \end{bmatrix},\ P_x = \begin{bmatrix} 80.5684 & 60.8782 \\ 60.8782 & 66.8253 \end{bmatrix}$$

which are approximately identical to their optimal values. In addition, we list Table I to compare the performance between our method and existing methods [13], [14], [15], [16].

V. Conclusion

This brief studied data-driven ADP control problem for linear SPSs with completely unknown dynamics. A novel two-stage value iteration algorithm was developed to sequentially solve two optimal control problems related to the separated fast and slow state feedback gains. The first contribution of this brief is that the developed model-free learning algorithm is applicable to both standard and nonstandard SPSs.

TABLE I
Performance Comparison of Two-Stage Learning Method and Existing Methods

Algorithm	Our Method	Methods of [13]–[16]
Initial Admissible F_x^0, F_z^0	No	Yes
Nonsingular of A_{22}	No	Yes
Knowledge of A_{21}, A_{22}, B_2	No	Yes
No. of Iterations	$100 + 100$	6

The second contribution is that the present algorithm does not require an initial admissible control policy. For SPSs, the slow state varies much slower than the fast state does. Thus, control designs based on asynchronously sampled measurements may be more computationally efficient, as will be considered in our future work.

References

[1] Y. Xie, X. Yu, Y. Shi, and L. Guo, "SPT-based composite hierarchical antidisturbance control applied to a quadrotor UAV," *IEEE Trans. Ind. Electron.*, vol. 70, no. 1, pp. 635–645, Jan. 2023.
[2] C. Yang, Y. Xu, L. Zhou, and Y. Sun, "Model-free composite control of flexible manipulators based on adaptive dynamic programming," *Complexity*, vol. 2018, Oct. 2018, Art. no. 9720309.
[3] J. Zhao, C. Yang, W. Dai, and W. Gao, "Reinforcement learning-based composite optimal operational control of industrial systems with multiple unit devices," *IEEE Trans. Ind. Informat.*, vol. 18, no. 2, pp. 1091–1101, Feb. 2022.
[4] Y. Wang, P. Shi, and H. Yan, "Reliable control of fuzzy singularly perturbed systems and its application to electronic circuits," *IEEE Trans. Circuits Syst. I., Reg. Papers*, vol. 65, no. 10, pp. 3519–3528, Oct. 2018.
[5] P. Kokotovic, H. Khalil, and J. O'Reilly, *Singular Perturbation Methods in Control: Analysis and Design*. Philadelphia, PA, USA: SIAM, 1999.
[6] J. Zhao, C. Yang, and W. Gao, "Reinforcement learning based optimal control of linear singularly perturbed systems," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 69, no. 3, pp. 1362–1366, Mar. 2022.
[7] D. Wang, J. Wu, J. Ren, and J. Qiao, "Online value iteration for intelligent discounted tracking design of constrained systems," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 69, no. 9, pp. 3829–3833, Sep. 2022.
[8] T. Bian and Z.-P. Jiang, "Value iteration and adaptive dynamic programming for data-driven adaptive optimal control design," *Automatica*, vol. 71, pp. 348–360, Sep. 2016.
[9] D. Liu, S. Xue, B. Zhao, B. Luo, and Q. Wei, "Adaptive dynamic programming for control: A survey and recent advances," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 51, no. 1, pp. 142–160, Jan. 2021.
[10] W. Xue, J. Fan, V. G. Lopez, Y. Jiang, T. Chai, and F. L. Lewis, "Off-policy reinforcement learning for tracking in continuous-time systems on two time scales," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 32, no. 10, pp. 4334–4346, Oct. 2021.
[11] S. Mukherjee, H. Bai, and A. Chakrabortty, "Reduced-dimensional reinforcement learning control using singular perturbation approximations," *Automatica*, vol. 126, Apr. 2021, Art. no. 109451.
[12] S. Mukherjee, H. Bai, and A. Chakrabortty, "On robust model-free reduced-dimensional reinforcement learning control for singularly perturbed systems," in *Proc. Amer. Control Conf.*, 2020, pp. 3914–3919.
[13] C. Yang, S. Zhong, X. Liu, W. Dai, and L. Zhou, "Adaptive composite suboptimal control for linear singularly perturbed systems with unknown slow dynamics," *Int. J. Robust Nonlinear Control*, vol. 30, no. 7, pp. 2625–2643, 2020.
[14] L. Zhou, J. Zhao, L. Ma, and C. Yang, "Decentralized composite suboptimal control for a class of two-time-scale interconnected networks with unknown slow dynamics," *Neurocomputing*, vol. 382, pp. 71–79, Mar. 2020.
[15] J. Wang, C. Peng, J. Park, H. Shen, and K. Shi, "Reinforcement learning-based near optimization for continuous-time Markov jump singularly perturbed systems," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, early access, Jan. 2, 2023, doi: 10.1109/TCSII.2022.3233790.
[16] X. Liu, C. Yang, B. Luo, and W. Dai, "Suboptimal control for nonlinear slow-fast coupled systems using reinforcement learning and Takagi–Sugeno fuzzy methods," *Int. J. Adapt. Control Signal Process.*, vol. 35, no. 6, pp. 1017–1038, 2021.
[17] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal Control*. Hoboken, NJ, USA: Wiley, 2012.
[18] 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, Dec. 2016.
[19] T. H. Cormen, C. Leiserson, R. Rivest, and C. Stein, *Introduction to Algorithms*. Cambridge, MA, USA: MIT Press, 2009.