Reinforcement Learning-Based Near Optimization for Continuous-Time Markov Jump Singularly Perturbed Systems

Jing Wang, Chuanjun Peng, Ju H. Park, *Senior Member, IEEE*, Hao Shen, *Member, IEEE*, and Kaibo Shi, *Member, IEEE*

***Abstract*—The design of a suboptimal controller for continuous-time Markov jump singularly perturbed systems with partially unknown dynamics is studied in this brief. With fast and slow decomposition technique, the original Markov jump singularly perturbed systems are decomposed into fast and slow subsystems as a new attempt. On this basis, an offline parallel Kleinman algorithm and an online parallel integral reinforcement learning algorithm are presented to cope with the different subsystems, respectively. Meanwhile, the controllers obtained by the above two algorithms are used to design the suboptimal controllers for original systems. Furthermore, the suboptimality of the proposed controllers is also discussed. Finally, an example of the electric circuit model is shown to illustrate the applicability of the proposed method.**

***Index Terms*—Markov jump systems, fast and slow decomposition technique, singularly perturbed systems, reinforcement learning.**

I. Introduction

THE PAST few decades have witnessed that two-time-scale phenomenon receives much attention due to its frequent occurrence in various fields such as mechanical systems, electrical networks and mobile robots [1], [2], [3]. To describe the phenomenon of the two-time-scale, singularly perturbed systems (SPSs) are introduced as one of an important strategy. In [4], the authors adopted a time-scale separation technique to decompose the SPSs. However, in this brief, the random jumps in practical systems caused by unexpected events and attacks are not considered [5]. Thus, how to capture the inevitable stochastic switching phenomenon in SPSs has become a hot topic.

Markov jump systems (MJSs) have been widely investigated due to their ability to describe random switching phenomena between different subsystems [6], [7], [8], [9]. The switching between modes is subject to the Markov process and the switching rules are described by the transition rate [10]. Furthermore, the combination of those two systems, namely, Markov jump singularly perturbed systems (MJSPSs), have emerged. However, the existing results about MJSPSs are obtained by offline calculation, which requires accurate system matrices information [11]. There is no doubt that this is a harsh condition in practical application. Therefore, it has become a challenging problem to obtain controllers through data and has attracted a large number of scholars to conduct relevant research.

On the other hand, integral reinforcement learning (IRL), as a learning mechanism in which an agent constantly updates its policy in the process of interacting with the environment to obtain the maximum reward, is similar to finding the optimal policy in optimal control to minimize the performance index to some extents [12]. For example, a novel IRL approach is developed to find a solution for multiplayer non-zero sum games in MJSs with completely unknown dynamics [13]. Thereinto, the design of optimal controller for MJSPSs with unknown dynamics had been a largely under explored domain, which also arouses our curiosity about it.

Summarizing above considerations, the issue of the design of controller for MJSPSs with partially unknown dynamics is investigated. The main contributions are summarized as follows

(1) On the basis of singularly perturbed theory (SPT), the linear continuous-time MJSPSs with unknown dynamics of the slow subsystem are analyzed for the first time, and the dependence on singularly perturbed parameter (SPP) in design of controller is eliminated.

(2) The time-scale separation technique is applied to the MJSPSs, so the controller design problem of the original MJSPSs is transformed into two subproblems related to fast and slow subsystems. An offline algorithm is used to obtain the optimal controller for the fast subsystem, while a model-free RL algorithm is used to obtain the optimal controller for the slow subsystem.

Manuscript received 11 September 2022; revised 23 November 2022; accepted 28 December 2022. Date of publication 2 January 2023; date of current version 8 June 2023. This work was supported in part by the National Natural Science Foundation of China under Grant 62273006 and Grant 62173001, and in part by the Natural Science Foundation for Distinguished Young Scholars of Higher Education Institutions of Anhui Province under Grant 2022AH020034. The work of Ju H. Park was supported by the National Research Foundation of Korea (NRF) Grant funded by the Korean Government (Ministry of Science and ICT) under Grant 2019R1A5A8080290. This brief was recommended by Associate Editor H. Yu. *(Corresponding authors: Ju H. Park; Hao Shen.)*

Jing Wang, Chuanjun Peng, and Hao Shen are with the Anhui Province Key Laboratory of Special Heavy Load Robot and the School of Electrical and Information Engineering, Anhui University of Technology, Maanshan 243032, China (e-mail: jingwang08@126.com; chuanjunpeng_1210@163.com; haoshen10@gmail.com).

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

Kaibo Shi is with the School of Information Science and Engineering, Chengdu University, Chengdu 610106, China (e-mail: skbs111@163.com).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TCSII.2022.3233790.

Digital Object Identifier 10.1109/TCSII.2022.3233790

(3) Different from obtaining the controller of original systems directly, the optimal controller corresponding to the fast and slow subsystems are used to reconstruct the composite controller, and the suboptimality of the controller proposed in this brief is proved.

Notation: $\mathbb{R}^m$ refers to m-dimensional real matrix. $\otimes$ denotes the Kronecker product. $\|\cdot\|$ indicates the Euclidean norm for vector or spectral norm for matrix. For matrix A, $A > 0$ indicates that A is a positive definite matrix. For $B \in \mathbb{R}^{m\times n}$, $\text{vec}(B) = [b_{11}, b_{12}, \ldots, b_{1n}, \ldots, b_{2n}, \ldots, b_{mn}]^T$. $C \in \mathbb{R}^{m\times m}$, $\widetilde{C} = [c_{11}, 2c_{21}, c_{22}, \ldots, 2c_{m1}, 2c_{m2}, \ldots, c_{mm}]^T$. $\boldsymbol{E}\{\cdot\}$ means the mathematical expectation. I_n means the n-dimensional identity matrix.

II. Problem Formulation

Consider a class of linear continuous-time MJSPSs described by

$$\dot{x}_{1(t)} = A_{11}\big(\partial_{(t)}\big)x_{1(t)} + A_{12}\big(\partial_{(t)}\big)x_{2(t)} + B_1\big(\partial_{(t)}\big)u_{(t)} \tag{1}$$

$$\varepsilon\dot{x}_{2(t)} = A_{21}\big(\partial_{(t)}\big)x_{1(t)} + A_{22}\big(\partial_{(t)}\big)x_{2(t)} + B_2\big(\partial_{(t)}\big)u_{(t)} \tag{2}$$

$$y_{(t)} = C_1\big(\partial_{(t)}\big)x_{1(t)} + C_2\big(\partial_{(t)}\big)x_{2(t)} \tag{3}$$

where $x_{1(t)} \in \mathbb{R}^{n1}$ is the slow state and $x_{2(t)} \in \mathbb{R}^{n2}$ is the fast state; $x_{(t)} = [x_{1(t)}^T \;\; x_{2(t)}^T]^T \in \mathbb{R}^n$ refers to the system state; $u_{(t)} \in \mathbb{R}^m$, $y_{(t)} \in \mathbb{R}^p$ represent the control input and output, respectively; $0 < \varepsilon \ll 1$ refers to SPP. $A_{11}(\partial_{(t)})$, $A_{12}(\partial_{(t)})$, $A_{21}(\partial_{(t)})$, $A_{22}(\partial_{(t)})$, $B_1(\partial_{(t)})$, $B_2(\partial_{(t)})$, $C_1(\partial_{(t)})$ and $C_2(\partial_{(t)})$ are mode-dependent matrices with appropriate dimensions. For convenience, they are labeled as A_{11a}, A_{12a}, A_{21a}, A_{22a}, B_{1a}, B_{2a}, C_{1a}, C_{2a}, respectively. $\{\partial_{(t)}, t \geq 0\}$ is the Markov chain and get values from a finite set $\mathbf{M} \triangleq \{1, 2, \ldots, M\}$, where M means the total number of modes. The Markov process is considered with the following transition rates

$$P_r\big\{\partial_{(t+\Delta t)} = b \mid \partial_{(t)} = a\big\} = \begin{cases} \pi_{ab}\Delta t + o(\Delta t) & b \neq a \\ 1 + \pi_{aa}\Delta t + o(\Delta t) & b = a \end{cases}$$

where $\Delta t > 0$, $\lim_{\Delta t\to 0}(o(\Delta t)/\Delta t) = 0$; $a, b \in \mathbf{M}$, $\pi_{ab} \geq 0 (b \neq a)$ is the jumping rate, which means jumping from mode a at time t to mode b at time $t + \Delta t$ and $\pi_{aa} = -\sum_{b\in\mathbf{M}, b\neq a} \pi_{ab}$.

Then, for the further analysis, some assumptions are given.

Assumption 1: A_{11a}, A_{12a} and B_{1a} are unknown whereas A_{21a}, A_{22a} and B_{2a} are known.

Assumption 2: A_{22a} is nonsingular.

According to [4], the systems (1)-(3) can be rewritten into the following forms without fast state

$$\begin{aligned} \dot{\bar{x}}_{1(t)} &= A_{11a}\bar{x}_{1(t)} + A_{12a}\bar{x}_{2(t)} + B_{1a}\bar{u}_{(t)} \\ 0 &= A_{21a}\bar{x}_{1(t)} + A_{22a}\bar{x}_{2(t)} + B_{2a}\bar{u}_{(t)} \\ \bar{y}_{(t)} &= C_{1a}\bar{x}_{1(t)} + C_{2a}\bar{x}_{2(t)} \end{aligned}$$

where $\bar{x}_{1(t)}$, $\bar{x}_{2(t)}$, $\bar{u}_{(t)}$ and $\bar{y}_{(t)}$ represent the slow state, fast state, control input and output within the slow state.

Under Assumption 2, the slow subsystems (4)-(5) can be obtained as

$$\dot{x}_{s(t)} = A_{sa}x_{s(t)} + B_{sa}u_{s(t)} \tag{4}$$

$$y_{s(t)} = C_{sa}x_{s(t)} + D_{sa}u_{s(t)} \tag{5}$$

where $x_{s(t)} = \bar{x}_{1(t)}$, $y_{s(t)} = \bar{y}_{(t)}$, $u_{s(t)} = \bar{u}_{(t)}$, $A_{sa} = A_{11a} - A_{12a}A_{22a}^{-1}A_{21a}$, $B_{sa} = B_{1a} - A_{12a}A_{22a}^{-1}B_{2a}$, $C_{sa} = C_{1a} - C_{2a}A_{22a}^{-1}A_{21a}$ and $D_{sa} = -C_{2a}A_{22a}^{-1}B_{2a}$.

When analyzing the fast subsystems, all variables related to the slow subsystems can be treated as a constant. Therefore, we can get the following equation

$$0 = \varepsilon\dot{\bar{x}}_{2(t)} = A_{21a}\bar{x}_{1(t)} + A_{22a}\bar{x}_{2(t)} + B_{2a}\bar{u}_{(t)}. \tag{6}$$

Making the difference between the (6) and (2), the fast subsystems (7)-(8) can be obtained as

$$\varepsilon\dot{x}_{f(t)} = A_{22a}x_{f(t)} + B_{2a}u_{f(t)} \tag{7}$$

$$y_{f(t)} = C_{2a}x_{f(t)} \tag{8}$$

where $x_{f(t)} = x_{2(t)} - \bar{x}_{2(t)}$, $u_{f(t)} = u_{(t)} - \bar{u}_{(t)}$, $y_{f(t)} = y_{(t)} - y_{s(t)}$.

In the next section, we will discuss the design of controllers for subsystems separately according to their different characteristics.

III. Main Results

In this section, the design method of composite controller for MJSPSs is introduced. And the suboptimality of composite controller is also discussed.

For fast subsystems, the performance index can be expressed as

$$J_{fa}\big(x_{f(t)}, u_{fa(t)}\big) = \boldsymbol{E}\Big\{\int_t^\infty \Big(x_{f(\tau)}^T C_{2a}^T C_{2a} x_{f(\tau)} + u_{fa(\tau)}^T R_a u_{fa(\tau)}\Big)\Big\}.$$

In order to minimize the performance index, the optimal policy is given as

$$u_{fa(t)}^* = -R_a^{-1}B_{2a}^T P_{fa}^* x_{f(t)}$$

where $R_a > 0$ is a weighting matrix, P_{fa}^* is the solution of following coupled algebraic Riccati equations (CAREs)

$$0 = A_{22a}^T P_{fa} + P_{fa}A_{22a} - P_{fa}B_{2a}R_a^{-1}B_{2a}^T P_{fa} + \sum_{b=1}^{M} \pi_{ab}P_{fb} + C_{2a}^T C_{2a}. \tag{9}$$

However, it is difficult to obtain P_{fa} from (9) directly. In order to obtain the solutions of (9), an offline parallel algorithm using subsystem transformation method from [14] is presented in the following. According to [15], we assume the initial stabilizing gains are known.

In order to obtain the optimal policy for slow subsystems, the following conversion is made in advance for subsequent calculation

$$w_{s(t)} = u_{s(t)} + \big(R_a + D_{sa}^T D_{sa}\big)^{-1} D_{sa}^T C_{sa} x_{s(t)}. \tag{10}$$

Combine (4) and (10), it follows that

$$\dot{x}_{s(t)} = A_{ssa}x_{s(t)} + B_{sa}w_{s(t)} \tag{11}$$

where $A_{ssa} = A_{sa} - B_{sa}(R_a + D_{sa}^T D_{sa})^{-1}D_{sa}^T C_{sa}$.

For (11), the optimal control policy is designed as

$$\begin{aligned} w^*_{sa(t)} &= -G^*_{sa}x_{s(t)} \\ &= -\left(R_a + D^T_{sa}D_{sa}\right)^{-1}B^T_{sa}P^*_{Gsa}x_{s(t)} \end{aligned} \tag{12}$$

where P^*_{Gsa} can be obtained by solving following CAREs

$$\begin{aligned} 0 = {} & A^T_{ssa}P_{Gsa} + P_{Gsa}A_{ssa} + Q_{ssa} + \sum_{b=1}^{M}\pi_{ab}P_{Gsb} \\ & - P_{Gsa}B_{sa}\left(R_a + D^T_{sa}D_{sa}\right)^{-1}B^T_{sa}P_{Gsa} \end{aligned}$$

where $Q_{ssa} = C^T_{sa}(I_{P1} - D_{sa}(R_a + D^T_{sa}D_{sa})^{-1}D^T_{sa})C_{sa}$.

With (12), the following performance index can be minimized

$$\begin{aligned} J_{wsa}\left(x_{s(t)}, w_{sa(t)}\right) = \boldsymbol{E}\{\int_t^{\infty} & (w^T_{sa(\tau)}\left(R_a + D^T_{sa}D_{sa}\right)w_{sa(\tau)} \\ & + x^T_{s(\tau)}Q_{ssa}x_{s(\tau)})d\tau\}. \end{aligned} \tag{13}$$

Although the authors in [16] proposed an adaptive dynamic programming technique that does not rely on prior knowledge of system matrices to obtain optimal controller, this technique can not be directly used to design controller due to the performance index corresponding to the slow subsystems in this brief are different from [16]. Now, the optimal control problem of the original slow subsystems can be transformed into the optimal control problem of (11). According to $V(x_{s(t)}) = x^T_{s(t)}P_{Gsa}x_{s(t)}$ and (13), the integral Bellman equation are obtained as

$$\begin{aligned} & x^T_{s(t+\delta t)}P^{(k)}_{Gsa}x_{s(t+\delta t)} - x^T_{s(t)}P^{(k)}_{Gsa}x_{s(t)} \\ & = -\int_t^{t+\delta t} x^T_{s(\tau)}\bar{Q}^{(k)}_{ssa}x_{s(\tau)}d\tau \\ & \quad + 2\int_t^{t+\delta t} x^T_{s(\tau)}\left(G^{(k)}_{sa}\right)^T R_a G^{(k+1)}_{sa}x_{s(\tau)}d\tau \\ & \quad + 2\int_t^{t+\delta t} x^T_{s(\tau)}\left(G^{(k)}_{sa}\right)^T D^T_{sa}D_{sa}G^{(k+1)}_{sa}x_{s(\tau)}d\tau \\ & \quad + 2\int_t^{t+\delta t} w^T_{sa(\tau)}\left(R_a + D^T_{sa}D_{sa}\right)G^{(k+1)}_{sa}x_{s(\tau)}d\tau \end{aligned}$$

where δt means a small time interval and $\bar{Q}^{(k)}_{ssa} = Q_{ssa} + \sum_{b=1}^{M}\pi_{ab}P^{(k-1)}_{Gsb} + (G^{(k)}_{sa})^T(R_a + D^T_{sa}D_{sa})G^{(k)}_{sa}$.

Since the state $x_{s(t)}$ in (11) is virtual, we choose $x_{1(t)}$ to replace $x_{s(t)}$ in the data collection process. For the purpose of distinguishing between using ideal data and actual data, when using $x_{1(t)}$, rewrite the above mentioned P_{Gsa} and G_{sa} into P'_{Gsa} and G'_{sa}. To facilitate subsequent analysis, subsystem transformation method is also used and the following definitions are given as

$$\begin{aligned} \mathfrak{B}_{ax_1x_1} &\triangleq \left[\varrho_1, \quad \ldots \quad \varrho_j\right]^T \\ \mathfrak{A}_{ax_1x_1} &\triangleq \left[\int_{t_0}^{t_1} x_{1a}\otimes x_{1a}d\tau, \quad \ldots \quad \int_{t_{j-1}}^{t_j} x_{1a}\otimes x_{1a}d\tau\right]^T \\ \mathfrak{A}_{ax_1w_s} &\triangleq \left[\int_{t_0}^{t_1} x_{1a}\otimes w_{sa}d\tau, \quad \ldots \quad \int_{t_{j-1}}^{t_j} x_{1a}\otimes w_{sa}d\tau\right]^T \end{aligned}$$

where $\varrho_j = (\hat{x}_{1(t_j)} - \hat{x}_{1(t_{j-1})})$, $x_{1a} = [x_{11a}, \ x_{12a}, \ \ldots \ x_{1n_1a}]^T$, with $\hat{x}_1 \triangleq [x^2_{11a}, \ x_{11a}x_{12a}, \ \ldots \ x_{12a}x_{1n_1a}, \ \ldots \ x^2_{1n_1a}]^T$.

With the above preparations, a compact form is presented to acquire G'_{sa}

$$\begin{bmatrix} \widetilde{P}'^{(k)}_{Gsa} \\ vec\left(G'^{(k+1)}_{sa}\right) \end{bmatrix} = \left(\left(\varpi^{(k)}_{sa}\right)^T\varpi^{(k)}_{sa}\right)^{-1}\left(\varpi^{(k)}_{sa}\right)^T\vartheta^{(k)}_{sa} \tag{14}$$

where $\varpi^{(k)}_{sa} = [\mathfrak{B}_{ax_1x_1}, \ \Lambda^{(k)}]$, $\vartheta^{(k)}_{sa} = -\mathfrak{A}_{ax_1x_1}vec(\bar{Q}'^{(k)}_{ssa})$, with $\Lambda^{(k)} = -2\mathfrak{A}_{ax_1x_1}(I_{n_1}\otimes(G'^{(k)}_{sa})^T(R_a + D^T_{sa}D_{sa})) - 2\mathfrak{A}_{ax_1w_s}(I_{n_1}\otimes(R_a + D^T_{sa}D_{sa}))$, $\bar{Q}'^{(k)}_{ssa} = Q_{ssa} + \sum_{b=1}^{M}\pi_{ab}P'^{(k-1)}_{Gsb} + (G'^{(k)}_{sa})^T(R_a + D^T_{sa}D_{sa})G'^{(k)}_{sa}$.

Now, a novel parallel Algorithm 2 is presented to get the optimal solutions of revised slow subsystems.

When optimal control policies are obtained by using Algorithm 2, according to (10), the optimal gain of the slow subsystems with state variable $x_{1(t)}$ can also be obtained at the same time, and its form presented as following

$$K'_{sa} = G'_{sa} + \left(R_a + D^T_{sa}D_{sa}\right)^{-1}D^T_{sa}C_{sa}.$$

When the optimal gains corresponding to the subsystems are obtained, composite controller gain can be defined as

$$K'_{ca} = \left[\left(I_{p2} + K_{fa}A^{-1}_{22a}B_{2a}\right)K'_{sa} + K_{fa}A^{-1}_{22a}A_{21a}, \quad K_{fa}\right]. \tag{15}$$

Then, the composite controller can be given as

$$u_{ca(t)} = -K'_{ca}x_{(t)}.$$

Theorem 1: The performance index $J_{ca}(x_{(t)}, u^{\infty}_{ca(t)})$ with composite controller $u^{\infty}_{ca(t)}$ and $J_{opta}(x_{(t)}, u_{opta(t)})$ with optimal controller satisfy

$$J_{ca}\left(x_{(t)}, u^{\infty}_{ca(t)}\right) = J_{opta}\left(x_{(t)}, u_{opta(t)}\right) + o(\varepsilon).$$

Proof: Since $x_{s(t)}$ can not be measured directly, $x_{1(t)}$ is used to replace $x_{s(t)}$ in Algorithm 2. Meanwhile, according to [17], $x_{s(t)}$ and $x_{1(t)}$ differ by a small constant related to SPP. Then, it can be deduced that

$$\begin{aligned} u^{\infty}_{ca(t)} &= \lim_{k\to\infty} u^{(k)}_{sa(t)} + u_{fa(t)} + o(\varepsilon) \\ &= u^*_{sa(t)} + o(\varepsilon) + u^*_{fa(t)} + o(\varepsilon) \\ &= u_{opta(t)} + o(\varepsilon) \end{aligned}$$

where $u_{opta(t)} = -K_{opta}x_{(t)} = R^{-1}_a B^T_{a\varepsilon}P_{opta}x_{(t)}$, thereinto P_{opta} is the solution of the following CAREs

$$\begin{aligned} 0 = {} & A^T_{a\varepsilon}P_{opta} + P_{opta}A_{a\varepsilon} + Q_a \\ & - P_{opta}B_{a\varepsilon}R^{-1}_a B^T_{a\varepsilon}P_{opta} + \sum_{b=1}^{M}\pi_{ab}P_{optb} \end{aligned}$$

where $A_{a\varepsilon} \triangleq \begin{bmatrix} A_{11a} & A_{12a} \\ \varepsilon^{-1}A_{21a} & \varepsilon^{-1}A_{22a} \end{bmatrix}$, $B_{a\varepsilon} \triangleq \begin{bmatrix} B_{1a} \\ \varepsilon^{-1}B_{2a} \end{bmatrix}$.

Then, $J_{opta}(x_{(t)}, u_{opta(t)})$ and $J_{ca}(x_{(t)}, u^{\infty}_{ca(t)})$ can be defined as

$$\begin{aligned} J_{opta}\left(x_{(t)}, u_{opta(t)}\right) = \boldsymbol{E}\{\int_0^{\infty} & (x^T_{(\tau)}Q_a x_{(\tau)} \\ & + u^T_{opta(\tau)}R_a u_{opta(\tau)})d\tau\} \end{aligned} \tag{16}$$

Algorithm 1 An Offline Parallel Model-Based Algorithm

Step I: Give a set of initial stabilizing gain matrices $\left\{K_{f1}^{(0)}, K_{f2}^{(0)}, \ldots, K_{fM}^{(0)}\right\}$;

Step II: Solve P_{fa}^{*} from M parallel decoupled algebraic Lyapunov equations with $\hat{A}_{22a} = A_{22a} + \frac{\pi_{aa}}{2} I_{n_2}$

$$\begin{aligned} &P_{fa}^{(k+1)}\left(\hat{A}_{22a} - B_{2a}K_{fa}^{(k)}\right) \\ &+ \left(\hat{A}_{22a} - B_{2a}K_{fa}^{(k)}\right)^{T} P_{fa}^{(k+1)} \\ &= -\left(K_{fa}^{(k)}\right)^{T} R_a K_{fa}^{(k)} - Q_{fa}^{(k+1)} \end{aligned}$$

where $Q_{fa}^{(k+1)} = C_{2a}^{T} C_{2a} + \sum_{b=1, b\neq a}^{M} \pi_{ab} P_{fb}^{(k)}$;

Step III: Update $K_{fa}^{(k+1)}$ by $K_{fa}^{(k+1)} = R_a^{-1} B_{2a}^{T} P_{fa}^{(k)}$;

Step IV: Repeat Step II and Step III with $k = k+1$ until $\left\| P_{fa}^{(k+1)} - P_{fa}^{(k)} \right\| \leq \epsilon_1$, where $\epsilon_1 > 0$ is a predefined small threshold;

Step V: Obtain the approximated optimal control policy as $u_{fa(t)} = -K_{fa}^{(k)} x_{fa(t)}$.

Algorithm 2 An Online Parallel Model-Free Algorithm

Step I: Give a set of initial stabilizing gain matrices $\left\{G_{s1}^{\prime(0)}, G_{s2}^{\prime(0)}, \ldots G_{sM}^{\prime(0)}\right\}$ and employ $w_{sa(t)} = -G_{sa}^{\prime(0)} x_{1a(t)} + e_a$ as the control input during learning stage, where e_a is the exploration noise;

Step II: Solve $P_{Gsa}^{\prime(k)}$ and $G_{sa}^{\prime(k+1)}$ from the equation (14);

Step III: Let $k = k+1$ and repeat Step II until $\left\| P_{Gsa}^{\prime(k+1)} - P_{Gsa}^{\prime(k)} \right\| \leq \epsilon_2$, $a \in \mathbf{M}$, where $\epsilon_2 > 0$ is a predefined small threshold;

Step IV: Obtain the approximated optimal control policy as $w_{sa(t)} = -G_{sa}^{\prime(k)} x_{1a(t)}$.

$$\begin{aligned} J_{ca}\left(x_{(t)}, u_{ca(t)}^{\infty}\right) = \boldsymbol{E}\{&\int_{0}^{\infty} (x_{(\tau)}^{T} Q_a x_{(\tau)} \\ &+ \left(u_{ca(\tau)}^{\infty}\right)^{T} R_a u_{ca(\tau)}^{\infty}) d\tau\}. \end{aligned} \tag{17}$$

By comparing (16) and (17), one can obtain that

$$J_{ca}\left(x_{(t)}, u_{ca(t)}^{\infty}\right) = J_{opta}\left(x_{(t)}, u_{opta(t)}\right) + o(\varepsilon).$$

The proof is completed. ■

Remark 1: When the fast state changes quickly and it can not be measured, the controller of the original systems can only be obtained from the slow subsystems after the fast and slow decomposition. As a special case in designing composite controller, (15) can be rewritten as $K_{ra}' = [\, K_{sa}' \;\; 0\,]$, where K_{ra}' is regarded as the reduced order gain.

Now, the reduced controller can be given as

$$u_{ra(t)} = -K_{ra}' x_{(t)}.$$

Theorem 2: The performance index $J_{ra}(x_{(t)}, u_{ra(t)}^{\infty})$ with reduced order controller $u_{ra(t)}^{\infty}$ and $J_{opta}(x_{(t)}, u_{opta(t)})$ with optimal controller satisfy

$$J_{ra}\left(x_{(t)}, u_{ra(t)}^{\infty}\right) = J_{opta}\left(x_{(t)}, u_{opta(t)}\right) + o(\varepsilon).$$

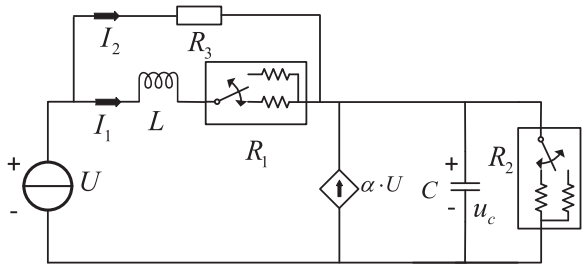

Fig. 1. An electronic circuit with jumping characteristics.

Proof: The proof process is similar to that of Theorem 1, so it is omitted here. ■

IV. Simulation

In this section, a practical electronic circuit which is modified from [18] is shown to confirm the practicability of proposed methods. The circuit schematic diagram is given in Fig. 1. And the circuit can be modeled as MJSPSs and state equations can be given in the following

$$\begin{cases} L\dot{I}_{1(t)} = U - u_{c(t)} - I_{1(t)} R_1 \\ C\dot{u}_{c(t)} = \alpha U + I_{1(t)} - \frac{u_{c(t)}}{R_2} + I_{2(t)} \end{cases} \tag{18}$$

where coefficient $\alpha = 0.5$, $C = 100$mF means the capacitance, $L = 1$H represents the inductance, $\varepsilon = 0.1$; $I_{2(t)} = 9I_{1(t)}$, both $I_{1(t)}$ and $I_{2(t)}$ mean the current. The ideal voltage source is taken as control input $u_{(t)} = U$. $x_{1(t)} = I_{1(t)}$ and $x_{2(t)} = u_{c(t)}$ refer to state vectors. Consider that the circuit operates in model a, $R_1 = 5\Omega$, $R_2 = 0.05\Omega$, and in model b, $R_1 = 10\Omega$, $R_2 = 0.2\Omega$. In Fig. 1, R_1, R_2 and R_3 represent resistance. The transition rate matrix is considered as $\pi = \begin{bmatrix} -3 & 3 \\ 1.5 & -1.5 \end{bmatrix}$. Thus, the system equation (18) can be modeled as

$$\Pi_{\varepsilon} \dot{x}_{(t)} = A_i x_{(t)} + B_i u_{(t)}$$

where

$$A_a = \begin{bmatrix} -5 & -1 \\ 10 & -20 \end{bmatrix}, \; A_b = \begin{bmatrix} -10 & -1 \\ 10 & -5 \end{bmatrix},$$
$$\Pi_{\varepsilon} = diag\{1, \varepsilon\}, \;\; B_a = B_b = \begin{bmatrix} 1 \\ 0.5 \end{bmatrix}.$$

The weighting matrices are selected as $Q_a = \begin{bmatrix} 1 & 0.1 \\ 0.1 & 1 \end{bmatrix}$, $Q_b = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}$, $R_a = R_b = 1$.

For fast subsystems, set $\epsilon_1 = 1 \times 10^{-5}$ and the optimal controller gains are obtained after 7th iterations by using Algorithm 1

$$K_{fa}^{(7)} = 0.0032, \;\; K_{fb}^{(7)} = 0.0438.$$

Next, Algorithm 2 is used to acquire optimal controllers for slow subsystems with unknown dynamics. Set the initial feedback gains $G_{sa}^{\prime(0)} = G_{sb}^{\prime(0)} = 10$. The two exploration noises for subsystems in Algorithm 2 are set as $e_a = e_b = \sum_{i=1}^{100} (sin(0.5t) + sin(0.01t))$, respectively. Select predefined threshold as $\epsilon_2 = 1 \times 10^{-5}$. Then, the approximate optimal

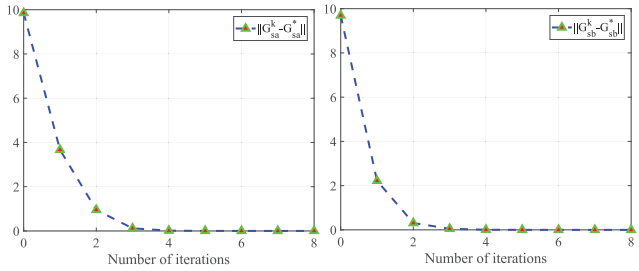

Fig. 2. Convergence of $G_{sa}^{\prime(k)}$ and $G_{sb}^{\prime(k)}$ for subsystems.

TABLE I
THE VALUE OF PERFORMANCE INDEXES WITH DIFFERENT CONTROLLER IN TWO MODES

	J_{opt}	J_c	J_r
Mode a	10.42549	10.43313	10.42757
Mode b	33.86826	33.86845	33.88098

control gains are given as follows after 8th iterations by executing Algorithm 2

$$G_{sa}^{\prime(8)} = 0.1478, \quad G_{sb}^{\prime(8)} = 0.3095.$$

In Fig. 2, as the number of iterations increase, the difference between the value obtained by Algorithm 2 and the optimal value obtained when the system dynamic is known gradually approaches zero.

Now, the composite controller and the reduced order controller under the two modes can be given directly

$$K_{ca}^{\prime(8)} = \begin{bmatrix} 0.1488 & 0.0032 \end{bmatrix}, K_{cb}^{\prime(8)} = \begin{bmatrix} 0.5215 & 0.0438 \end{bmatrix},$$
$$K_{ra}^{\prime(8)} = \begin{bmatrix} 0.1504 & 0 \end{bmatrix}, K_{rb}^{\prime(8)} = \begin{bmatrix} 0.6065 & 0 \end{bmatrix}.$$

In order to verify the suboptimality of the designed controllers, these controllers will be compared with the optimal controllers. The optimal controller gains are shown as

$$K_{opta} = \begin{bmatrix} 0.1466 & 0.0007 \end{bmatrix}, K_{optb} = \begin{bmatrix} 0.4221 & 0.0726 \end{bmatrix}.$$

Choose the initial states as $x_0 = [1 \quad 0.1]^T$, then the specific value for various performance indexes for infinite time is given in the Table I. Comparing the data presented in Table I, it can be obtained that the performance loss with composite controller and reduced order controller are 0.073%, 0.02% in mode a, and 0.00056%, 0.037% in mode b.

V. CONCLUSION

In this brief, the fast and slow decomposition method has been used for the first time to study continuous-time SPSs with random jump parameters with partially unknown dynamics. Firstly, SPT has been used to decompose the original systems into fast subsystems and slow subsystems. Then, the optimal controllers have been designed respectively for the two subsystems with different characteristics. Furthermore, the optimal controllers of each subsystem have been applied to construct the composite controller and the reduced order controller for original systems. At the same time, the effectiveness of the presented results have been illustrated by a electronic circuit model.

REFERENCES

[1] 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.
[2] 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.
[3] J. Wang, C. Yang, J. Xia, Z.-G. Wu, and H. Shen, "Observer-based sliding mode control for networked fuzzy singularly perturbed systems under weighted try-once-discard protocol," *IEEE Trans. Fuzzy Syst.*, vol. 30, no. 6, pp. 1889–1899, Jun. 2022.
[4] J. Chow and P. Kokotovic, "A decomposition of near-optimum regulators for systems with slow and fast modes," *IEEE Trans. Autom. Control*, vol. AC-21, no. 5, pp. 701–705, Oct. 1976.
[5] Z.-W. Liu, Y.-L. Shi, H. Yan, B.-X. Han, and Z.-H. Guan, "Secure consensus of multiagent systems via impulsive control subject to deception attacks," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 70, no. 1, pp. 166–170, Jan. 2023, doi: 10.1109/TCSII.2022.3196042.
[6] H. Wan, X. Luan, H. R. Karimi, and F. Liu, "High-order moment filtering for Markov jump systems in finite frequency domain," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 66, no. 7, pp. 1217–1221, Jul. 2019.
[7] W. Zou, C. K. Ahn, and Z. Xiang, "Fuzzy-approximation-based distributed fault-tolerant consensus for heterogeneous switched nonlinear multiagent systems," *IEEE Trans. Fuzzy Syst.*, vol. 29, no. 10, pp. 2916–2925, Oct. 2021.
[8] L. Wang, Z.-G. Wu, and Y. Shen, "Asynchronous mean stabilization of positive jump systems with piecewise-homogeneous Markov chain," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 68, no. 10, pp. 3266–3270, Oct. 2021.
[9] H. Shen, X. Hu, J. Wang, J. Cao, and W. Qian, "Non-fragile H_∞ synchronization for Markov jump singularly perturbed coupled neural networks subject to double-layer switching regulation," *IEEE Trans. Neural Netw. Learn. Syst.*, early access, Sep. 6, 2021, doi: 10.1109/TNNLS.2021.3107607.
[10] X. Ding and H. Li, "Finite-time time-variant feedback stabilization of logical control networks with Markov jump disturbances," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 67, no. 10, pp. 2079–2083, Oct. 2020.
[11] I. Borno and Z. Gajic, "Parallel algorithms for optimal control of weakly coupled and singularly perturbed jump linear systems," *Automatica*, vol. 31, no. 7, pp. 985–988, 1995.
[12] H. Modares and F. L. Lewis, "Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning," *Automatica*, vol. 50, no. 7, pp. 1780–1792, 2014.
[13] X. Xin et al., "Online reinforcement learning multiplayer non-zero sum games of continuous-time Markov jump linear systems," *Appl. Math. Comput.*, vol. 412, Jan. 2022, Art. no. 126537.
[14] S. He, J. Song, Z. Ding, and F. Liu, "Online adaptive optimal control for continuous-time Markov jump linear systems using a novel policy iteration algorithm," *IET Control Theory Appl.*, vol. 9, no. 10, pp. 1536–1543, 2015.
[15] Y. Jiang and Z.-P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, no. 10, pp. 2699–2704, 2012.
[16] S. He, M. Zhang, H. Fang, F. Liu, X. Luan, and Z. Ding, "Reinforcement learning and adaptive optimization of a class of Markov jump systems with completely unknown dynamic information," *Neural Comput. Appl.*, vol. 32, no. 18, pp. 14311–14320, 2020.
[17] 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.
[18] T.-H. Li and K.-J. Lin, "Stabilization of singularly perturbed fuzzy systems," *IEEE Trans. Fuzzy Syst.*, vol. 12, no. 5, pp. 579–595, Oct. 2004.