Received: Added at production | Revised: Added at production | Accepted: Added at production

DOI: xxx/xxxx

RESEARCH ARTICLE

Reinforcement learning-based composite suboptimal control for Markov jump singularly perturbed systems with unknown dynamics

Wenqian Li | Yun Wang | Jiacheng Wu | Hao Shen

[1]School of Electrical and Information Engineering, Anhui University of Technology, Ma'anshan, China

Correspondence
Jiacheng Wu, School of Electrical and Information Engineering, Anhui University of Technology, Ma'anshan 243002, China.
Email: jc980507@163.com

Summary

In this article, a model-free parallel reinforcement learning method is proposed to solve the suboptimal control problem for the Markov jump singularly perturbed systems. First, since fast and slow dynamics coexist in Markov jump singularly perturbed systems, it may lead to ill-conditioned numerical problems during the controller design process. Therefore, the original system can be decomposed into independent subsystems at different time-scales by employing the reduced order method. Besides, a model-based parallel algorithm is designed to obtain the optimal controllers of the fast and slow subsystems respectively. Moreover, within the framework of reinforcement learning, the composite controller of the Markov jump singularly perturbed systems can be obtained without system dynamics. Finally, a numerical example is introduced to prove the effectiveness of proposed algorithms.

KEYWORDS:
Markov jump singularly perturbed systems, reinforcement learning, fast and slow decomposition, composite control

1 | INTRODUCTION

In the past several decades, singularly perturbed systems (SPSs) have attracted much attention due to their powerful capability to model multi-timescale phenomena[1,2,3]. Such as robot systems[4,5,6], energy and power systems[7,8,9,10], mechanical systems[11,12], engineering and physics systems[13,14]. Since some small parasitic parameters inevitably exist in the engineering field, it usually leads to the coexistence of slow and fast time-scales in SPSs, which may bring an ill-conditional numerical problem. In order to eliminate the above problem, the singularly perturbed theory (SPT) was introduced. One is the adopted time-scale separation technique (TSST), which decomposes the original system into pure fast and pure slow subsystems, then the composite controller for the original system was designed[15,16]. Compared with the conventional full-order method, this reduced-order method can save learning time[17]. Besides, in[18], the authors converted the original system algebraic Riccati equations (AREs) into two asymmetric fast and slow AREs to solve the optimal control for SPSs. Then, the authors in[19] provided a new idea for solving the optimal control problem of the SPSs based on the eigenvector method, which decomposed AREs by employing Newton iterative approximation of the original equation solution. However, due to the change of internal parameters and the influence of the external environment, the system dynamics will inevitably change. Subsequently, the Markov jump systems (MJSs) were developed to describe the system of state changes.

MJSs as a special stochastic system can be used to describe a class of systems with abrupt variations, such as power systems, economic systems, and communication systems[20,21,22]. Recently, MJSs have been extensively investigated and have yielded a

series of notable results[23,24,25,26,27]. The characteristic of MJSs is that the jumping between different modes obeys the Markov process[28,29]. To study the characteristics of the SPSs and the MJSs simultaneously, the Markov jump singularly perturbed systems (MJSPSs) have attracted considerable attention. In[30], a class of parallel algorithms was proposed to deal with optimal control problem MJSPSs with arbitrary order precision. Moreover, in[31,32], the fuzzy H_∞ control problem was solved for nonlinear MJSPSs with partial information. After that, in[33], the H_∞ control and filtering problems for nonlinear MJSPSs approximated by Takagi-Sugeno fuzzy models were addressed. However, the results mentioned above are acquired that the system dynamics are known or partially known. It is obvious that this condition is tough in many practical applications.

To overcome the limitation mentioned above, reinforcement learning (RL) was introduced to solve the optimal controller design with unknown or partially unknown system dynamics[34,35,36,37,38]. The authors in[39] developed an integral RL method to deal with the optimal control problem of continuous-time systems with partial system dynamics. On this basis, in[40], a model-free RL algorithm was proposed to remove the assumption that the dynamic information of the system is partially unknown. In addition, in[41], an adaptive composite control method was proposed for SPSs with unknown slow dynamics by using SPT and RL approach. Afterwards, in[42], a novel model-free off-policy learning algorithm for semi-coupled SPSs with completely unknown dynamics was proposed. Furthermore, in[43], a composite controller was designed with unknown slow dynamics for the MJSPSs by the RL method. It is worth noting that the system dynamics of the fast-subsystems in[43] are assumed to be known, which is not practical in some real-world situations. Inspired by the above discussions, this article focuses on solving the suboptimal control problem for the MJSPSs with unknown system dynamics.

The contributions of this paper can be mainly concluded as follows

(1) As the first attempt, the composite controller design problem of linear continuous-time MJSPSs with unknown system dynamics is solved by employing the RL method.

(2) Based on the reduced order method, the composite controller of MJSPSs can be approximated by combining the optimal controllers for subsystems of two time-scales.

(3) The accuracy of the composite control solution method proposed in this paper is also demonstrated.

This paper has the following specific organization. Section 2 introduces the system description, problem description as well as system decomposition of this paper. In section 3, the optimal controllers of fast-subsystems and slow-subsystems are designed by the proposed model-free algorithms, respectively. Besides, the composite controller is obtained by combining the above optimal controllers. In section 4, a numerical example is presented to prove the convergence and effectiveness of the proposed algorithms. Section 5, the conclusion of this paper is presented.

Notation : R^n indicates an n-dimensional real matrix. $E\{\cdot\}$ represents the mathematical expectation of stochastic processes. $\otimes$ stands for the Kronecker product. $\|\cdot\|$ represents the Euclidean norm of vectors. For $H \in R^{n\times m}$, $vec(H) = [h_1^T\ h_2^T\ \cdots\ h_m^T]^T$, where $h_i \in R^n$ means the ith column of the matrix H. For $B \in R^{m\times m}$, $B' = \left[b_{11}, 2b_{12}, b_{22}, \ldots, 2b_{m1}, 2b_{m2}, \ldots, b_{mm}\right]^T$. $I_p \in R^{p\times p}$ means an identity matrix.

2 | PROBLEM FORMULATION

2.1 | System Description

Consider a class of continuous-time MJSPSs modeled by

$$\dot{x}_1(\rho) = A_{11}(s(\rho))\,x_1(\rho) + A_{12}(s(\rho))\,x_2(\rho) \tag{1}$$

$$\gamma\dot{x}_2(\rho) = A_{22}(s(\rho))\,x_2(\rho) + B_2(s(\rho))\,u(\rho) \tag{2}$$

$$y(\rho) = C_1(s(\rho))\,x_1(\rho) + C_2(s(\rho))\,x_2(\rho) \tag{3}$$

with the transition probabilities

$$\Pr\{s(\rho+\Delta\rho) = \beta \mid s(\rho) = \alpha\} = \begin{cases} \pi_{\alpha\beta}\Delta\rho + o(\Delta\rho) & (\alpha \neq \beta) \\ 1 + \pi_{\alpha\alpha}\Delta\rho + o(\Delta\rho) & (\alpha = \beta) \end{cases}$$

where $x_1(\rho) \in R^{n_1}$ and $x_2(\rho) \in R^{n_2}$ are the slow and fast state vectors, respectively. $u(\rho) \in R^m$ represents the control input. $y(\rho) \in R^p$ denotes the control output. $0 < \gamma \ll 1$ stands for the singularly perturbed parameter (SPP). $A_{11}(s(\rho)) \in R^{n_1\times n_1}$, $A_{12}(s(\rho)) \in R^{n_1\times n_2}$, $A_{22}(s(\rho)) \in R^{n_2\times n_2}$, $B_2(s(\rho)) \in R^{n_2\times m_1}$, $C_1(s(\rho)) \in R^{p\times n_1}$ and $C_2(s(\rho)) \in R^{p\times n_2}$ are mode-independent constant matrices with appropriate dimensions. $\{s(\rho), \rho \geq 0\}$ represents the system model subject to Markov stochastic process, which takes values in a discrete set $M = \{1, 2, ..., N\}$. Furthermore, for transition probabilities, $\alpha, \beta \in M$,

$\Delta\rho > 0$, $lim_{\Delta\rho\to 0}\frac{(o(\Delta\rho))}{\Delta\rho} = 0$, and $\pi_{\alpha\beta} \geqslant 0$ represents the system transition rate from mode α to mode β at time $\rho \to \rho + \Delta\rho$, with $\pi_{\alpha\alpha} = -\sum_{\alpha\neq\beta}\pi_{\alpha\beta}$. For convenience, assuming that $s(\rho) = \alpha$, then $A_{11}(s(\rho))$, $A_{12}(s(\rho))$, $A_{22}(s(\rho))$, $B_2(s(\rho))$, $C_1(s(\rho))$, $C_2(s(\rho))$, $Q(s(\rho))$, $R(s(\rho))$ can be denoted as $A_{11\alpha}$, $A_{12\alpha}$, $A_{22\alpha}$, $B_{2\alpha}$, $C_{1\alpha}$, $C_{2\alpha}$, Q_α, R_α respectively.

2.2 | Problem Description

The mode-dependent optimal control policy for MJSPSs (1)-(3) is written as

$$u^*(\rho) = -K_\alpha^* x(\rho)$$

which can minimize the performance index in the following

$$J(x(\rho), u(\rho)) = E\left\{\int_\rho^\infty (x^T(\sigma) Q_\alpha x(\sigma) + u_\alpha^T(\sigma) R_\alpha u_\alpha(\sigma))d\sigma\right\}$$

where K_α^* refers to the optimal control gain for MJSPSs (1)-(3). $Q_\alpha \geqslant 0$ and $R_\alpha \geqslant 0$ are the mode-dependent positive-definite weighting matrices with appropriate dimensions.

Before the further presentation, some assumptions are given in below.

Assumption 1. The matrix $A_{22\alpha}$ is nonsingular.

Assumption 2. The matrices $A_{11\alpha}$, $A_{12\alpha}$, $A_{22\alpha}$, $B_{2\alpha}$ are unknown.

2.3 | System Decomposition

For MJSPSs (1)-(3), traditional methods for designing optimal controllers cannot be applied directly due to the existence of SPP, which may result in ill-conditioned numerical problems. Therefore, under Assumption 1 and Assumption 2, the SPT is employed to decompose the MJSPSs (1)-(3) into two subsystems based on different time-scales. One is the slow-subsystems and the other is the fast-subsystems. Then, the optimal controllers of the two subsystems are designed, respectively. Furthermore, the composite controller of MJSPSs (1)-(3) can be obtained by combining of the subsystems optimal controllers[41].

Under Assumption 1, set SPP $\gamma = 0$, then the slow-subsystems can be written as

$$\dot{x}_s(\rho) = A_{s\alpha} x_s(\rho) + B_{s\alpha} u_s(\rho) \tag{4}$$

$$y_s(\rho) = C_{s\alpha} x_s(\rho) + D_{s\alpha} u_s(\rho) \tag{5}$$

where $A_{s\alpha} = A_{11\alpha}$, $B_{s\alpha} = -A_{12\alpha}A_{22\alpha}^{-1}B_{2\alpha}$, $C_{s\alpha} = C_{1\alpha}$, $D_{s\alpha} = -C_{2\alpha}A_{22\alpha}^{-1}B_{2\alpha}$.

For the slow-subsystems, the mode-dependent optimal controller is designed as

$$u_{s\alpha}^*(\rho) = -K_{s\alpha}^* x_s(\rho) \tag{6}$$

which can minimize the performance index as follows

$$J_{s\alpha}\left(x_s(\rho), u_{s\alpha}(\rho)\right) = E\left\{\int_\rho^\infty (x_s^T(\sigma) C_{s\alpha}^T C_{s\alpha} x_s(\sigma) + 2u_{s\alpha}^T(\sigma) D_{s\alpha}^T C_{s\alpha} x_s(\sigma) + u_{s\alpha}^T(\sigma)\left(R_\alpha + D_{s\alpha}^T D_{s\alpha}\right) u_{s\alpha}(\sigma))d\sigma\right\}$$

where $K_{s\alpha}^*$ is the optimal control gain for slow-subsystems (4)-(5).

On the other hand, the fast-subsystems can be described as

$$\gamma\dot{x}_f(\rho) = A_{22\alpha} x_f(\rho) + B_{2\alpha} u_f(\rho) \tag{7}$$

$$y_f(\rho) = C_{2\alpha} x_f(\rho) \tag{8}$$

where $x_f(\rho) = x_2(\rho) + A_{22\alpha}^{-1}B_{2\alpha}u_s(\rho)$, $y_f(\rho) = y(\rho) - y_s(\rho)$.

Furthermore, the mode-dependent optimal controller design for the fast-subsystems is shown below

$$u_{f\alpha}^*(\rho) = -K_{f\alpha}^* x_f(\rho) \tag{9}$$

which can minimize the performance index in the following

$$J_{f\alpha}\left(x_f(\rho), u_{f\alpha}(\rho)\right) = E\left\{\int_{\rho}^{\infty}(x_f^T(\sigma)C_{2\alpha}^T C_{2\alpha}x_f(\sigma) + u_{f\alpha}^T(\sigma)R_\alpha u_{f\alpha}(\sigma))d\sigma\right\} \tag{10}$$

where $K_{f\alpha}^*$ is the optimal control gain for fast-subsystems (7)-(8).

Therefore, the composite controller of MJSPSs (1)-(3) has the following form

$$u_{c\alpha}^*(\rho) = u_{s\alpha}^*(\rho) + u_{f\alpha}^*(\rho) = -K_{s\alpha}^* x_s(\rho) - K_{f\alpha}^* x_f(\rho). \tag{11}$$

According to[41], Lemma 1 is given to explain the relationship between the original system and decomposit- ion subsystems.

Lemma 1. Assuming that the control policies (6), (9), and (11) are used for systems (4)-(5), (7)-(8), and (1)-(3), respectively. Moreover, if both $A_{11\alpha} - B_{2\alpha}K_{s\alpha}$ and $A_{22\alpha} - B_{2\alpha}K_{f\alpha}$ are asymptotically stable, then all of the following equations hold for $\rho \in [0, +\infty)$

$$\begin{aligned} x_1(\rho) &= x_s(\rho) + o(\gamma) \\ x_2(\rho) &= x_f(\rho) + A_{22\alpha}^{-1}B_{2\alpha}K_{s\alpha}x_s(\rho) + o(\gamma) \\ u_\alpha(\rho) &= u_{s\alpha}(\rho) + u_{f\alpha}(\rho) + o(\gamma) \\ y(\rho) &= y_s(\rho) + y_f(\rho) + o(\gamma). \end{aligned}$$

Next section, we will propose different learning methods to design the optimal controllers for the fast and slow subsystems based on the different characteristics, respectively. Thus, the composite controller of MJSPSs (1)-(3) will be obtained by (11).

3 | MAIN RESULTS

In this section, under Assumption 2, two off-policy algorithms are developed for designing optimal controllers for each subsystem. Furthermore, the convergence of the proposed methods is proved.

3.1 | Optimal Controller Design for Slow-subsystems

With unknown system dynamics, an adaptive dynamic programming technique was proposed in[41] that can obtain the optimal controller. However, it is notable that the performance index for the slow-subsystems is different from[41], which means that this method cannot be directly applied. Therefore, a novel learning method that can design the optimal controller of the slow-subsystems satisfying Assumption 1 is proposed below.

Before proceeding further in the analysis, the following conversion for subsequent calculation is given

$$\begin{aligned} v_s(\rho) &= u_s(\rho) + (R_\alpha + D_{s\alpha}^T D_{s\alpha})^{-1}E_{s\alpha}x_s(\rho) \\ \dot{x}_s(\rho) &= A_{ss\alpha}x_s(\rho) + B_{s\alpha}v_s(\rho) \end{aligned} \tag{12}$$

where $E_{s\alpha} = D_{s\alpha}^T C_{s\alpha}$, $A_{ss\alpha} = A_{s\alpha} - B_{s\alpha}(R_\alpha + D_{s\alpha}^T D_{s\alpha})^{-1}E_{s\alpha}$.

For (12), the optimal control policy has the form as

$$v_{s\alpha}^*(\rho) = -G_{s\alpha}^* x_s(\rho) = -(R_\alpha + D_{s\alpha}^T D_{s\alpha})^{-1}B_{s\alpha}^T P_{s\alpha}^* x_s(\rho)$$

which can minimize the following performance index

$$J_{vs\alpha}\left(x_s(\rho), v_{s\alpha}(\rho)\right) = \boldsymbol{E}\left\{\int_{\rho}^{\infty}[x_s^T(\sigma)Q_{ss\alpha}x_s(\sigma) + v_{s\alpha}^T(\sigma)(R_\alpha + D_{s\alpha}^T D_{s\alpha})v_{s\alpha}(\sigma)]d\sigma\right\} \tag{13}$$

where

$$Q_{ss\alpha} = C_{s\alpha}^T C_{s\alpha} - E_{s\alpha}^T R_{s\alpha}^{-1}E_{s\alpha} = C_{s\alpha}^T(I_p - D_{s\alpha}(R_\alpha + D_{s\alpha}^T D_{s\alpha})^{-1}D_{s\alpha}^T)C_{s\alpha}$$

and $P_{s\alpha}$ are the solutions of the following coupled algebraic Riccati equations (CAREs)

$$\hat{A}_{ss\alpha}^T P_{s\alpha} + P_{s\alpha}\hat{A}_{ss\alpha} + Q_{ss\alpha} + \sum_{\alpha\neq\beta,\beta=1}^{N} \pi_{\alpha\beta} P_{s\beta} - P_{s\alpha} B_{s\alpha}(R_\alpha + D_{s\alpha}^T D_{s\alpha})^{-1} B_{s\alpha}^T P_{s\alpha} = 0$$

where $\hat{A}_{ss\alpha} = A_{ss\alpha} + \frac{1}{2}\pi_{\alpha\alpha}$.

Therefore, the optimal controller design problem for (4) is equivalent to accessing the optimal controller of (12). By employing the value function $V\left(x_s(\rho)\right) = x_s^T P_{s\alpha} x_s$ and (13), the integral Bellman equation can be acquired as

$$\begin{aligned} x_{s,\rho+\Delta\rho}^T P_{s\alpha(\xi)} x_{s,\rho+\Delta\rho}^T - x_{s,\rho}^T P_{s\alpha(\xi)} x_{s,\rho} = & -\int_{\rho}^{\rho+\Delta\rho} x_s^T(\sigma)\,\bar{Q}_{ss\alpha(\xi)} x_s(\sigma)\,d\sigma + 2\int_{\rho}^{\rho+\Delta\rho} \left[(v_{s\alpha}(\sigma)\right. \\ & \left. + G_{s\alpha(\xi)} x_s(\sigma))^T (R_\alpha + D_{s\alpha}^T D_{s\alpha}) G_{s\alpha(\xi+1)} x_s(\sigma)\right] d\sigma \end{aligned} \tag{14}$$

where $\bar{Q}_{ss\alpha(\xi)} = Q_{ss\alpha} + \sum_{\alpha\neq\beta,\beta=1}^{N} \pi_{\alpha\beta} P_{s\beta(\xi-1)} + G_{s\alpha(\xi)}^T (R_\alpha + D_{s\alpha}^T D_{s\alpha}) G_{s\alpha(\xi)}$.

According to Assumption 2, the optimal controllers of fast and slow subsystems can not be directly acquired by solving CAREs. Therefore, an online off-policy parallel learning method is proposed in Algorithm 1. Furthermore, $x_1(\rho)$ is introduced in the data collection process to replace the state $x_s(\rho)$, which is virtual. Thus, rewriting the above mentioned $P_{s\alpha}$, $G_{s\alpha}$ into $\bar{P}_{s\alpha}$, $\bar{G}_{s\alpha}$ when using $x_1(\rho)$ as the actual data. For further analysis, some definitions are given in below

$$\begin{aligned} I_{\alpha x_1 x_1} &\triangleq \left[\int_{\rho_0}^{\rho_1} x_{1\alpha,\sigma}^T \otimes x_{1\alpha,\sigma}^T d\sigma, \int_{\rho_1}^{\rho_2} x_{1\alpha,\sigma}^T \otimes x_{1\alpha,\sigma}^T d\sigma, \ldots, \int_{\rho_{l-1}}^{\rho_l} x_{1\alpha,\sigma}^T \otimes x_{1\alpha,\sigma}^T d\sigma\right]^T \in R^{l\times n_1^2} \\ I_{\alpha x_1 v_s} &\triangleq \left[\int_{\rho_0}^{\rho_1} x_{1\alpha,\sigma}^T \otimes v_{s\alpha,\sigma}^T d\sigma, \int_{\rho_1}^{\rho_2} x_{1\alpha,\sigma}^T \otimes v_{s\alpha,\sigma}^T d\sigma, \ldots, \int_{\rho_{l-1}}^{\rho_l} x_{1\alpha,\sigma}^T \otimes v_{s\alpha,\sigma}^T d\sigma\right]^T \in R^{l\times m_1 n_1} \\ \delta_{\alpha x_1 x_1} &\triangleq \left[\hat{x}_{1\alpha}(\rho_1) - \hat{x}_{1\alpha}(\rho_0), \hat{x}_{1\alpha}(\rho_2) - \hat{x}_{1\alpha}(\rho_1), \ldots \hat{x}_{1\alpha}(\rho_l) - \hat{x}_{1\alpha}(\rho_{l-1})\right]^T \in R^{l\times \frac{n_1(n_1+1)}{2}} \\ \bar{P}_{s\alpha}' &\triangleq \left[p_{11\alpha}, 2p_{12\alpha}, ..., 2p_{1n_1\alpha}, p_{22\alpha}, 2p_{23\alpha}, ..., p_{n_1n_1\alpha}\right]^T \end{aligned}$$

where

$$\begin{aligned} x_{1\alpha} &\triangleq \left[x_{11\alpha}, x_{12\alpha}, x_{13\alpha}, \ldots x_{1n_1\alpha}\right]^T \\ \hat{x}_{1\alpha} &\triangleq x_{1\alpha}^T \otimes x_{1\alpha}^T = \left[x_{11\alpha}^2, x_{11\alpha}x_{12\alpha}, ...x_{11\alpha}x_{1n_1\alpha}, x_{12\alpha}^2, x_{12\alpha}x_{13\alpha}, ...x_{12\alpha}x_{1n_1\alpha}, ..., x_{1n_1\alpha}^2\right]^T \\ \bar{P}_{s\alpha} &\in R^{n_1\times n_1} \rightarrow \bar{P}_{s\alpha}' \in R^{\frac{n_1(n_1+1)}{2}}, x_{1\alpha} \in R^{n_1} \rightarrow \hat{x}_{1\alpha} \in R^{\frac{n_1(n_1+1)}{2}}. \end{aligned}$$

Thus, (14) can be described as

$$\Theta_{s\alpha(\xi)} \begin{bmatrix} \bar{P}_{s\alpha(\xi)}' \\ vec\left(\bar{G}_{s\alpha(\xi+1)}\right) \end{bmatrix} = \Xi_{s\alpha(\xi)} \tag{15}$$

where

$$\begin{aligned} \Theta_{s\alpha(\xi)} &= \left[\delta_{\alpha x_1 x_1} \; -2\left[I_{\alpha x_1 x_1}\left(I_{n_1} \otimes \bar{G}_{s\alpha(\xi)}^T (R_\alpha + D_{s\alpha}^T D_{s\alpha})\right) + I_{\alpha x_1 v_s}\left(I_{n_1} \otimes (R_\alpha + D_{s\alpha}^T D_{s\alpha})\right)\right]\right] \\ \Xi_{s\alpha(\xi)} &= -I_{\alpha x_1 x_1} vec\left(\bar{Q}_{ss\alpha(\xi)}\right) \end{aligned}$$

with $\Theta_{s\alpha(\xi)} \in \mathfrak{R}^{l\times\left[\frac{1}{2}n_1(n_1+1)+m_1n_1\right]}$, $\Xi_{s\alpha(\xi)} \in \mathfrak{R}^l$.

Moreover, the (15) can be solved if Lemma 2 is satisfied

$$\begin{bmatrix} \bar{P}_{s\alpha(\xi)}' \\ vec\left(\bar{G}_{s\alpha(\xi+1)}\right) \end{bmatrix} = \left(\Theta_{s\alpha(\xi)}^T \Theta_{s\alpha(\xi)}\right)^{-1} \Theta_{s\alpha(\xi)}^T \Xi_{s\alpha(\xi)}. \tag{16}$$

Lemma 2. In[40], to ensure that every step of the algorithm can be implemented online, matrix $\Theta_{s\alpha(\xi)}$ should satisfy

$$rank\left(\left[I_{\alpha x_1 x_1}, I_{\alpha x_1 v_s}\right]\right) = \frac{n_1(n_1+1)}{2} + m_1 n_1.$$

Algorithm 1 is used for obtaining the control law of the slow-subsystems, and the algorithm can solve $\bar{P}_{s\alpha(\xi)}$ and $\bar{G}_{s\alpha(\xi)}$, furthermore the slow-subsystems controller gains all can be acquired. Its form is as follows

$$\bar{K}_{s\alpha} = \bar{G}_{s\alpha(\xi)} + (R_\alpha + D_{s\alpha}^T D_{s\alpha})^{-1} E_{s\alpha}.$$

Algorithm 1: Off-policy Model-Free Parallel RL Algorithm for Slow-subsystems

Initialization
Give the initial stabilizing sequence $\left\{\bar{G}_{s1(0)}, \bar{G}_{s2(0)}, \bar{G}_{s3(0)} \ldots \bar{G}_{sN(0)}\right\}$, and select a threshold $\epsilon > 0$.
for $\alpha = 1 : N$ **do**
  Data collection:
  Employ initial control policies $v_{s\alpha} = -\bar{G}_{s\alpha(0)} x_1(\rho) + e_{s\alpha}$ in time interval $\left[\rho_0, \rho_l\right]$, where $e_{s\alpha}$ is the exploration noise.
  Compute $I_{\alpha x_1 x_1}, I_{\alpha x_1 v_s}$.
  while $\max\{||\bar{P}_{s\alpha(\xi+1)} - \bar{P}_{s\alpha(\xi)}||\} \geq \epsilon$ **do**
    Iterative computation:
    Parallel solve $\bar{G}_{s\alpha(\xi)}$ and $\bar{P}_{s\alpha(\xi)}$ from (16)
    $\xi \leftarrow \xi + 1$;
  end
end

Remark 1. By Algorithm 1, which is a model-free and off-policy parallel algorithm, the optimal control policy can be obtained for the slow-subsystems without system dynamics. It is worth mentioning that v_s derived here is not the same as the optimal control law u_s.

3.2 | Optimal Controller Design for Fast-subsystems

For the fast-subsystems, $P_{f\alpha}$ can be obtained by solving the following CAREs

$$A_{f\alpha}^T P_{f\alpha} + P_{f\alpha} A_{f\alpha} + Q_{f\alpha} + \sum_{\alpha\neq\beta,\beta=1}^{N} \pi_{\alpha\beta} P_{f\beta} - P_{f\alpha} B_{f\alpha} R_{f\alpha}^{-1} B_{f\alpha}^T P_{f\alpha} = 0$$

where $A_{f\alpha} = A_{22\alpha} + \frac{1}{2}\pi_{\alpha\alpha}$, $Q_{f\alpha} = C_{2\alpha}^T C_{2\alpha}$, $B_{f\alpha} = B_{2\alpha}$, $R_{f\alpha} = R_\alpha$.

Therefore, the optimal control policy can be shown below

$$u_{f\alpha}^*(\rho) = -K_{f\alpha}^* x_\rho(\rho) = -R_{f\alpha}^{-1} B_{f\alpha}^T P_{f\alpha}^* x_f(\rho).$$

However, due to the system dynamics needing to be known in advance, CAREs are usually difficult to be solved in practical applications. In order to overcome this limitation, a model-free parallel control scheme is presented in Algorithm 2. Moreover, the online implementation of Algorithm 2 is shown below.

Combining with the value function as $V\left(x_f(\rho)\right) = x_f^T P_{f\alpha} x_f$ and (10), we have

$$\begin{aligned} x_{f,\rho+\Delta\rho}^T P_{f\alpha(\xi)} x_{f,\rho+\Delta\rho}^T - x_{f,\rho}^T P_{f\alpha(\xi)} x_{f,\rho} = &- \int_\rho^{\rho+\Delta\rho} x_f^T(\sigma) \bar{Q}_{f\alpha(\xi)} x_f(\sigma)\, d\sigma \\ &+ 2 \int_\rho^{\rho+\Delta\rho} \left[\left(u_{f\alpha}(\sigma) + K_{f\alpha(\xi)} x_f(\sigma)\right)^T R_{f\alpha} K_{f\alpha(\xi+1)} x_f(\sigma)\right] d\sigma \end{aligned} \tag{17}$$

where $\bar{Q}_{f\alpha(\xi)} = Q_{f\alpha} + \sum_{\alpha\neq\beta,\beta=1}^{N} \pi_{\alpha\beta} P_{f\beta(\xi-1)} + K_{f\alpha(\xi)}^T R_{f\alpha} K_{f\alpha(\xi)}$.

Rewrite the above-mentioned $P_{f\alpha}$, $K_{f\alpha}$ into $\bar{P}_{f\alpha}$, $\bar{K}_{f\alpha}$. Before further analysis, some definitions are given in the following

$$I_{\alpha x_f x_f} \triangleq \left[\int_{\rho_0}^{\rho_1} x_{f\alpha,\sigma}^T \otimes x_{f\alpha,\sigma}^T d\sigma, \int_{\rho_1}^{\rho_2} x_{f\alpha,\sigma}^T \otimes x_{f\alpha,\sigma}^T d\sigma, \ldots, \int_{\rho_{l-1}}^{\rho_l} x_{f\alpha,\sigma}^T \otimes x_{f\alpha,\sigma}^T d\sigma\right]^T \in R^{l\times n_2^2}$$

$$I_{\alpha x_f u_f} \triangleq \left[\int_{\rho_0}^{\rho_1} x_{f\alpha,\sigma}^T \otimes u_{f\alpha,\sigma}^T d\sigma, \int_{\rho_1}^{\rho_2} x_{f\alpha,\sigma}^T \otimes u_{f\alpha,\sigma}^T d\sigma, \ldots, \int_{\rho_{l-1}}^{\rho_l} x_{f\alpha,\sigma}^T \otimes u_{f\alpha,\sigma}^T d\sigma\right]^T \in R^{l\times m_2 n_2}$$

$$\delta_{\alpha x_f x_f} \triangleq \left[\hat{x}_{f\alpha}(\rho_1) - \hat{x}_{f\alpha}(\rho_0), \hat{x}_{f\alpha}(\rho_2) - \hat{x}_{f\alpha}(\rho_1), \ldots, \hat{x}_{f\alpha}(\rho_l) - \hat{x}_{f\alpha}(\rho_{l-1})\right]^T \in R^{l\times \frac{n_2(n_2+1)}{2}}$$

$$\bar{P}'_{f\alpha} \triangleq \left[p_{11\alpha}, 2p_{12\alpha}, ..., 2p_{1n_2\alpha}, p_{22\alpha}, 2p_{23\alpha}, ..., p_{n_2n_2\alpha}\right]^T$$

where

$$x_{f\alpha} \triangleq \left[x_{11\alpha}, x_{12\alpha}, x_{13\alpha}, \ldots x_{1n_2\alpha}\right]^T$$

$$\hat{x}_{f\alpha} \triangleq x_{f\alpha}^T \otimes x_{f\alpha}^T = \left[x_{11\alpha}^2, x_{11\alpha}x_{12\alpha}, ...x_{11\alpha}x_{1n_2\alpha}, x_{12\alpha}^2, x_{12\alpha}x_{13\alpha}, ...x_{12\alpha}x_{1n_2\alpha}, ..., x_{1n_2\alpha}^2\right]^T$$

$$\bar{P}_{f\alpha} \in R^{n_2\times n_2} \rightarrow \bar{P}'_{f\alpha} \in R^{\frac{n_2(n_2+1)}{2}}, x_{f\alpha} \in R^{n_2} \rightarrow \hat{x}_{f\alpha} \in R^{\frac{n_2(n_2+1)}{2}}$$

Then (17) can be described as

$$\Theta_{f\alpha(\xi)} \begin{bmatrix} \bar{P}'_{f\alpha(\xi)} \\ vec\left(\bar{K}_{f\alpha(\xi+1)}\right) \end{bmatrix} = \Xi_{f\alpha(\xi)} \tag{18}$$

where

$$\Theta_{f\alpha(\xi)} = \left[\delta_{\alpha x_f x_f} \ -2\left[I_{\alpha x_f x_f}\left(I_{n_2} \otimes \bar{K}_{f\alpha(\xi)}^T R_{f\alpha}\right) + I_{\alpha x_f u_f}\left(I_{n_2} \otimes R_{f\alpha}\right)\right]\right]$$

$$\Xi_{f\alpha(\xi)} = -I_{\alpha x_f x_f} vec\left(\bar{Q}_{f\alpha(\xi)}\right)$$

with $\Theta_{f\alpha(\xi)} \in \Re^{l\times\left[\frac{1}{2}n_2(n_2+1)+m_2n_2\right]}$, $\Xi_{f\alpha(\xi)} \in \Re^l$.

Thus, under Lemma 2, the (18) can be solved by

$$\begin{bmatrix} \bar{P}'_{f\alpha(\xi)} \\ vec\left(\bar{K}_{f\alpha(\xi+1)}\right) \end{bmatrix} = \left(\Theta_{f\alpha(\xi)}^T \Theta_{f\alpha(\xi)}\right)^{-1} \Theta_{f\alpha(\xi)}^T \Xi_{f\alpha(\xi)}. \tag{19}$$

Algorithm 2: Off-policy Model-Free Parallel RL Algorithm for Fast-subsystems

Input: A initial sequence $\left\{\bar{K}_{f1(0)}, \bar{K}_{f2(0)}, \bar{K}_{f3(0)} \ldots \bar{K}_{fN(0)}\right\}$. A threshold $\epsilon > 0$.

Output: The optimal control policy of fast-subsystems as $u_{f\alpha}(\rho) = -\bar{K}_{f\alpha(\xi)} x_f(\rho)$.

```
for all α ∈ M do
    Data Collection:
    Employ initial control policies u_fα = −K̄_fα(0) x_f(ρ) + e_fα in [ρ_0, ρ_l], where e_fα denotes the exploration noise.
      Compute I_αx_f x_f, I_αx_f v_f.
    while max{||P̄_fα(ξ+1) − P̄_fα(ξ)||} ≥ ϵ do
        Iterative computation:
        Parallel learn K̄_fα(ξ) and P̄_fα(ξ) from (19)
        ξ ← ξ + 1;
    end
end
```

Remark 2. Different from previous studies, the dynamics of the fast-subsystems are unknown in this article. Moreover, a new model-free online parallel Algorithm 2 is designed to obtain the optimal control policy for the fast-subsystems.

3.3 | Composite Controller Design for MJSPSs

According to the proposed methods, the optimal control gains for each subsystems are both obtained. Thus, the composite controller gain has the following form

$$\bar{K}_{c\alpha} = \left[\left(I + \bar{K}_{f\alpha} A_{22\alpha}^{-1} B_{2\alpha} \right) \bar{K}_{s\alpha} \;\; \bar{K}_{f\alpha} \right]. \tag{20}$$

Furthermore, the convergence analysis of the proposed method is shown in Theorem 1.

Theorem 1. The relationship between performance index $J_{opt\alpha}\left(x(\rho), u_{opt\alpha}(\rho)\right)$ corresponding to the optimal controller $u_{opt\alpha}(\rho)$ and $J_{c\alpha}\left(x(\rho), u_{c\alpha}^{\infty}(\rho)\right)$ corresponding to the composite controller $u_{c\alpha}^{\infty}(\rho)$ satisfy

$$J_{c\alpha}\left(x(\rho), u_{c\alpha}^{\infty}(\rho)\right) = J_{opt\alpha}\left(x(\rho), u_{opt\alpha}(\rho)\right) + o(\varepsilon).$$

Proof. Due to the existence of SPP, the ideal data $x_s(\rho)$ is difficult to be measured directly, therefore, $x_1(\rho)$ takes place of $x_s(\rho)$. Then, the controller of the original system has the form as follows

$$\begin{aligned} u_{c\alpha}^{\infty}(\rho) &= \lim_{k\to\infty} u_{s\alpha(\xi)}(\rho) + u_{f\alpha}(\rho) + o(\gamma) \\ &= u_{s\alpha}^{*}(\rho) + o(\gamma) + u_{f\alpha}^{*}(\rho) + o(\gamma) \\ &= u_{opt\alpha}(\rho) + o(\gamma) \end{aligned}$$

where $u_{opt\alpha}(\rho) = -K_{opt\alpha}x(\rho) = -R_{\alpha}^{-1}B_{\alpha\gamma}^{T}P_{opt\alpha}x(\rho)$. $P_{opt\alpha}$ is the solution of the following CAREs

$$A_{\alpha\gamma}^{T}P_{opt\alpha} + P_{opt\alpha}A_{\alpha\gamma} + Q_{\alpha} + \sum_{\beta=1}^{N}\pi_{\alpha\beta}P_{opt\beta} - P_{opt\alpha}B_{\alpha\gamma}R_{\alpha}^{-1}B_{\alpha\gamma}^{T}P_{opt\alpha} = 0$$

where $A_{\alpha\gamma} = \begin{bmatrix} A_{11\alpha} & A_{12\alpha} \\ o & \gamma^{-1}A_{22\alpha} \end{bmatrix}$, $B_{\alpha\gamma} = \begin{bmatrix} o \\ \gamma^{-1}B_{2\alpha} \end{bmatrix}$ and $P_{opt\beta}$ represents the solution of CAREs for the MJSPSs (1)-(3) in mode β.

Thus, $J_{c\alpha}\left(x(\rho), u_{c\alpha}^{\infty}(\rho)\right)$ and $J_{opt\alpha}\left(x(\rho), u_{opt\alpha}(\rho)\right)$ can be defined as

$$J_{opt\alpha}\left(x(\rho), u_{opt\alpha}(\rho)\right) = \int_{0}^{\infty} (x^{T}(\sigma)Q_{\alpha}x(\sigma) + u_{opt\alpha}^{T}(\sigma)R_{\alpha}u_{opt\alpha}(\sigma))d\sigma \tag{21}$$

$$J_{c\alpha}\left(x(\rho), u_{c\alpha}^{\infty}(\rho)\right) = \int_{0}^{\infty} (x^{T}(\sigma)Q_{\alpha}x(\sigma) + \left(u_{c\alpha}^{\infty}(\sigma)\right)^{T}R_{\alpha}u_{c\alpha}^{\infty}(\sigma))d\sigma. \tag{22}$$

Comparing (21) with (22), we have

$$J_{c\alpha}\left(x(\rho), u_{c\alpha}^{\infty}(\rho)\right) = J_{opt\alpha}\left(x(\rho), u_{opt\alpha}(\rho)\right) + o(\gamma).$$

This ends the proof.

Remark 3. According to two model-free parallel algorithms proposed above, we can obtain the optimal controller of (4)-(5) and (7)-(8) respectively, so as to derive the composite controller of the original system, which reduces the computational complexity.

4 | AN ILLUSTRATIVE EXAMPLE

In this section, a numerical example is presented to verify the validity of the above-proposed algorithms. To show the applicability of Algorithm 1 and Algorithm 2, consider the following MJSPSs with two jump modes described as

$$A_1 = \begin{bmatrix} -5.2 & -1 \\ 0 & -52 \end{bmatrix}, A_2 = \begin{bmatrix} -5 & -1 \\ 0 & -50 \end{bmatrix}, B_1 = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix}, B_2 = \begin{bmatrix} 0 \\ 0.5 \end{bmatrix},$$

with the transition probabilities as

$$\Pi = \begin{bmatrix} -3 & 3 \\ 1.5 & -1.5 \end{bmatrix}.$$

The weighting matrices in the performance index can be expressed as

$$Q_1 = Q_2 = \begin{bmatrix} 1 & 1 \\ 1 & 1 \end{bmatrix}, R_1 = R_2 = I,$$

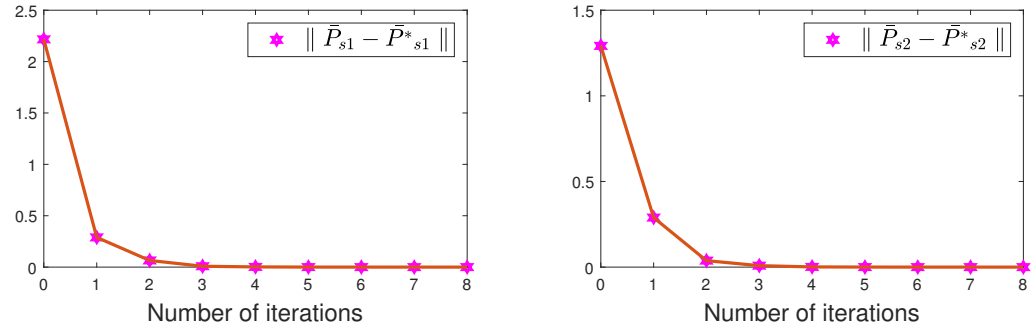

FIGURE 1 Convergence of $\bar{P}_{s1}$ and $\bar{P}_{s2}$ for slow-subsystems.

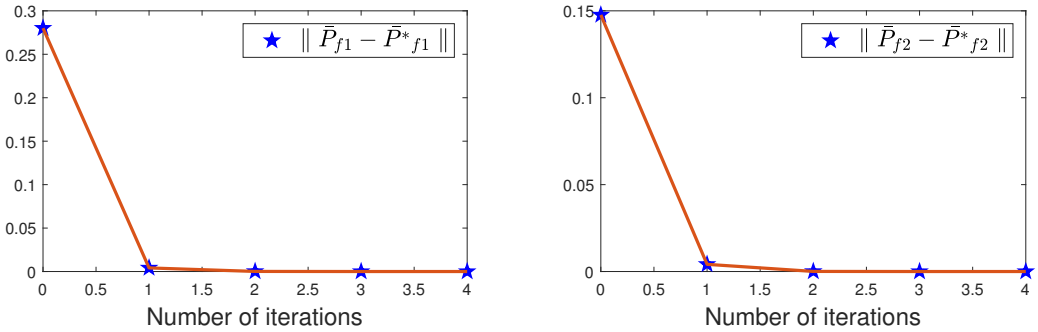

FIGURE 2 Convergence of $\bar{P}_{f1}$ and $\bar{P}_{f2}$ for fast-subsystems.

in the simulation, let $\gamma = 0.1$, the error value $\epsilon = 1 \times 10^{-5}$ and the initial stabilizing feedback gains of the slow-subsystems and fast-subsystems are viewed as $\bar{G}_{s\alpha(0)}$, $\bar{K}_{f\alpha(0)}$.

In the time interval $[0, 2]$s, the controllers are described as following

$$\begin{aligned} v_{s\alpha} &= -\bar{G}_{s\alpha(0)} x_{s\alpha} + e_{s\alpha} \\ u_{f\alpha} &= -\bar{K}_{f\alpha(0)} x_{f\alpha} + e_{f\alpha} \end{aligned}$$

where $e_{s\alpha} = e_{f\alpha} = 100 \sum_{n=1}^{100} sin\left(\omega_n \rho\right)$ refer to the exploration noises, and the ω_n represents a constant randomly choosing from $[-500, 500]$.

By employing Algorithm 1 and Algorithm 2, the optimal results for fast-subsystems and slow-subsystems can be calculated respectively after 9 iterations and 5 iterations as follows

$$\begin{aligned} \bar{P}_{s1(9)} &= 0.0969, \bar{P}_{s2(9)} = 0.0996, \bar{G}_{s1(9)} = -9.3183 \times 10^{-4}, \bar{G}_{s2(9)} = -9.9582 \times 10^{-4}, \\ \bar{P}_{f1(5)} &= 0.0096, \bar{P}_{f2(5)} = 0.0100, \bar{K}_{f1(5)} = 0.0048, \bar{K}_{f2(5)} = 0.0050. \end{aligned}$$

Based on the (20), the composite control feedback gains can be acquired as below

$$K_{c1} = \begin{bmatrix} 0.0087 & 0.0048 \end{bmatrix}, K_{c2} = \begin{bmatrix} 0.0090 & 0.0050 \end{bmatrix},$$

and by employing the CAREs, the optimal control feedback gains are presented as

$$K_{opt1} = \begin{bmatrix} 0.0088 & 0.0048 \end{bmatrix}, K_{opt2} = \begin{bmatrix} 0.0090 & 0.0050 \end{bmatrix}.$$

In Figure 1, the matrices $\bar{P}_{s1}$ and $\bar{P}_{s2}$ converge to the optimal value by Algorithm 1, which means that the optimal controllers of slow-subsystems are obtained. Similarly, Figure 2 exhibits the convergence of $\bar{P}_{f1}$ and $\bar{P}_{f2}$ Algorithm 2. It can be noted that these two parallel algorithms can obtain the optimal controllers of fast-subsystems and slow-subsystems without the system dynamics. Furthermore, the convergence of the matrices $\bar{K}_{f1}$ and $\bar{K}_{f2}$ for the fast-subsystems as well as the matrices $\bar{G}_{s1}$ and $\bar{G}_{s2}$ for the slow-subsystems are shown in Figures 3-4. Thus, based on Theorem 1, the composite controller gain can be obtained

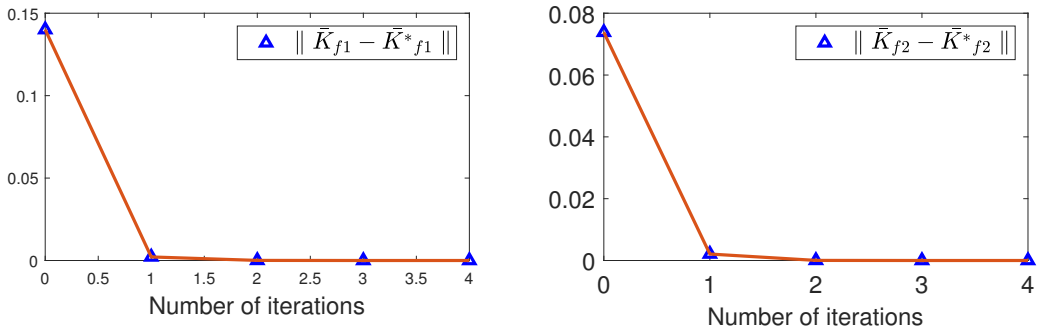

FIGURE 3 Convergence of $\bar{K}_{f1}$ and $\bar{K}_{f2}$ for fast-subsystems.

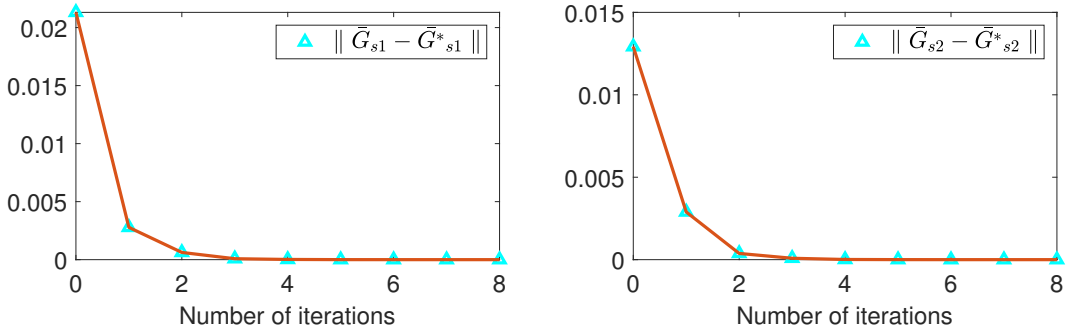

FIGURE 4 Convergence of $\bar{G}_{s1}$ and $\bar{G}_{s2}$ for slow-subsystems.

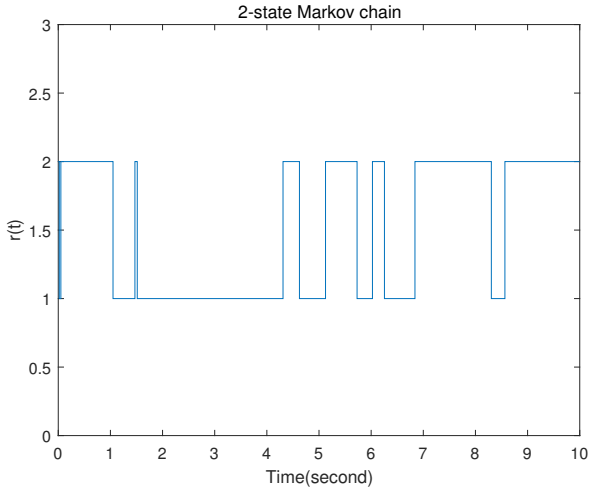

FIGURE 5 Evolution of the Markov chain.

by (20). Moreover, Figure 5 shows the mode evolution of the MJSPSs, which describes the changes of the two modes in a given time interval.

5 | CONCLUSION

This paper solved the adaptive suboptimal controller design problem for MJSPSs without either fast-subsystems dynamics or slow-subsystems dynamics. Under the TSST, the MJSPSs (1)-(3) can be decomposed into fast and slow subsystems. Thus, the suboptimal controller of MJSPSs can be obtained by compositing the optimal controllers of fast-subsystems and slow-subsystems. Furthermore, an online parallel RL learning method without system dynamics was proposed to design the optimal controllers for subsystems. Moreover, the simulation results show the practicality of the designed algorithms.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China under Grants 62273006, 62173001, 61873002, 61703004.

Conflict of interest

The authors declare no potential conflict of interests.

References

1. Aarthika K, Shanthi V, Ramos H. A computational approach for a two-parameter singularly perturbed system of partial differential equations with discontinuous coefficients. *Appl. Math. Comput.* 2022; 434: 127409.

2. Guo Y, Li J, Duan R. Extended dissipativity-based control for persistent dwell-time switched singularly perturbed systems and its application to electronic circuits. *Appl. Math. Comput.* 2021; 402: 126114.

3. Cheng J, Yan H, Park JH, Zong G. Output-feedback control for fuzzy singularly perturbed systems: a nonhomogeneous stochastic communication protocol approach. *IEEE Trans. Cybern.* 2023; 53(1): 76–87.

4. D'Andrea-Novel B, Campion G, Bastin G. Control of wheeled mobile robots not satisfying ideal velocity constraints: a singular perturbation approach. *Int. J. Robust Nonlinear Control* 1995; 5(4): 243–267.

5. Sun T, Liang D, Song Y. Singular-perturbation-based nonlinear hybrid control of redundant parallel robot. *IEEE Trans. Ind. Electron.* 2018; 65(4): 3326–3336.

6. Kim J, Croft EA. Full-state tracking control for flexible joint robots with singular perturbation techniques. *IEEE Trans. Control Syst. Technol.* 2019; 27(1): 63–73.

7. Abolvafaei M, Ganjefar S. Two novel approaches to capture the maximum power from variable speed wind turbines using optimal fractional high-order fast terminal sliding mode control. *Eur. J. Control* 2021; 60: 78–94.

8. Shen F, Ju P, Shahidehpour M, Li Z, Wang C, Shi X. Singular perturbation for the dynamic modeling of integrated energy systems. *IEEE Trans. Power Syst.* 2020; 35(3): 1718–1728.

9. Wang Y, Xie X, Chadli M, Xie S, Peng Y. Sliding-mode control of fuzzy singularly perturbed descriptor systems. *IEEE Trans. Fuzzy Syst.* 2021; 29(8): 2349–2360.

10. Vian JL, Sawan ME. H_∞ control for a singularly perturbed aircraft model. *Optim. Control Appl. Methods* 1994; 15(4): 277–289.

11. Shi Z, Wang Z. Optimal control for a class of complex singular system based on adaptive dynamic programming. *IEE/CAA J. Automatica Sinica* 2019; 6(1): 188–197.

12. Zhao J, Yang C, Dai W, Gao W. Reinforcement learning-based composite optimal operational control of industrial systems with multiple unit devices. *IEEE Trans. Ind. Informat.* 2022; 18(2): 1091–1101.

13. Kadalbajoo MK, Gupta V. A brief survey on numerical methods for solving singularly perturbed problems. *Appl. Math. Comput.* 2010; 217(8): 3641–3716.

14. Naidu DS, Calise AJ. Singular perturbations and time scales in guidance and control of aerospace systems: A survey. *J. Guid. Control Dyn.* 2012; 24(6): 1057–1078.

15. Chow J, Kokotovic P. A decomposition of near-optimum regulators for systems with slow and fast modes. *IEEE Trans. Autom. Control* 1976; 21(5): 701–705.

16. Xu Y, Wang X, Wang L, Wang K, Ma L. Learning Control for Flexible Manipulators with Varying Loads: A Composite Method with Robust Adaptive Dynamic Programming and Robust Sliding Mode Control. *Electronics* 2022; 11(6): 956.

17. Mukherjee S, Bai H, Chakrabortty A. Reduced-dimensional reinforcement learning control using singular perturbation approximations. *Automatica* 2021; 126: 109451.

18. Su WC, Gajic Z, Shen XM. The exact slow-fast decomposition of the algebraic Ricatti equation of singularly perturbed systems. *IEEE Trans. Autom. Control* 1992; 37(9): 1456–1459.

19. Kecman V, Bingulac S, Gajic Z. Eigenvector approach for order-reduction of singularly perturbed linear-quadratic optimal control problems. *Automatica* 1999; 35(1): 151–158.

20. Ugrinovskii* V, Pota HR. Decentralized control of power systems via robust control of uncertain Markov jump parameter systems. *Int. J. Control* 2005; 78(9): 662–677.

21. Shi P, Li F. A survey on Markovian jump systems: modeling and design. *Int. J. Control, Autom.* 2015; 13: 1–16.

22. Wu ZG, Shi P, Shu Z, Su H, Lu R. Passivity-based asynchronous control for Markov jump systems. *IEEE Trans. Autom. Control* 2017; 62(4): 2020–2025.

23. Shi P, Boukas EK, Agarwal RK. Kalman filtering for continuous-time uncertain systems with Markovian jumping parameters. *IEEE Trans. Autom. Control* 1999; 44(8): 1592–1597.

24. Luan X, Zhao S, Liu F. H_∞ control for discrete-time Markov jump systems with uncertain transition probabilities. *IEEE Trans. Autom. Control* 2013; 58(6): 1566–1572.

25. Xin X, Tu Y, Stojanovic V, et al. Online reinforcement learning multiplayer non-zero sum games of continuous-time Markov jump linear systems. *Appl. Math. Comput.* 2022; 412: 126537.

26. Liang T, Shi S, Ma Y. Asynchronous sliding mode control of continuous-time singular markov jump systems with time-varying delay under event-triggered strategy. *Appl. Math. Comput.* 2023; 448: 127947.

27. Qi W, Sha M, Park JH, Yan H, Xie X. Asynchronous stabilization for discrete hidden semi-Markov jumping power models with cyber attacks. *IEEE Trans. Circuits Syst. II, Exp. Briefs*; in press, doi: 10.1109/TCSII.2023.3244936.

28. Shi P, Xia Y, Liu G, Rees D. On designing of sliding-mode control for stochastic jump systems. *IEEE Trans. Autom. Control* 2006; 51(1): 97–103.

29. Qu H, Hu J, Song Y, Yang T. Mean square stabilization of discrete-time switching Markov jump linear systems. *Optim. Control Appl. Methods* 2019; 40(1): 141-151.

30. Borno I, Gajic Z. Parallel algorithms for optimal control of weakly coupled and singularly perturbed jump linear systems. *Automatica* 1995; 31(7): 985–988.

31. Li F, Xu S, Shen H. Fuzzy-Model-Based H_∞ Control for Markov Jump Nonlinear Slow Sampling Singularly Perturbed Systems With Partial Information. *IEEE Trans. Fuzzy Syst.* 2019; 27(10): 1952–1962.

32. Guo X, Yang G. Reliable H_∞ filter design for a class of discrete-time nonlinear systems with time-varying delay. *Optim. Control Appl. Methods* 2010; 31(4): 303–322.

33. Wang Y, Ahn CK, Yan H, Xie S. Fuzzy control and filtering for nonlinear singularly perturbed Markov jump systems. *IEEE Trans. Cybern.* 2021; 51(1): 297–308.

34. Luo B, Liu D, Huang T, Wang D. Model-free optimal tracking control via critic-only Q-learning. *IEEE Trans. Neural Netw. Learn. Syst.* 2016; 27(10): 2134–2144.

35. Wei Q, Shi G, Song R, Liu Y. Adaptive dynamic programming-based optimal control scheme for energy storage systems with solar renewable energy. *IEEE Trans. Ind. Electron.* 2017; 64(7): 5468–5478.

36. Li J, Ji L, Li H. Optimal consensus control for unknown second-order multi-agent systems: Using model-free reinforcement learning method. *Appl. Math. Comput.* 2021; 410: 126451.

37. Wong WC, Lee JH. A reinforcement learning-based scheme for direct adaptive optimal control of linear stochastic systems. *Optim. Control Appl. Methods* 2010; 31(4): 365–374.

38. Liang Y, Zhang H, Zhang J, Luo Y. Integral reinforcement learning-based guaranteed cost control for unknown nonlinear systems subject to input constraints and uncertainties. *Appl. Math. Comput.* 2021; 408: 126336.

39. Vrabie D, Pastravanu O, Abu-Khalaf M, Lewis FL. Adaptive optimal control for continuous-time linear systems based on policy iteration. *Automatica* 2009; 45(2): 477–484.

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

41. Yang C, Zhong S, Liu X, Dai W, Zhou L. Adaptive composite suboptimal control for linear singularly perturbed systems with unknown slow dynamics. *Int. J. Robust Nonlinear Control* 2020; 30(7): 2625–2643.

42. Zhao J, Yang C, Zhou L, Gao W. Model-Free Composite Control for a Class of Semi-Coupled Two-Time-Scale Systems: An Adaptive Dynamic Programming Approach. *IFAC-PapersOnLine* 2021; 54(14): 364–369.

43. Wang J, Peng C, Park JH, Shen H, Shi K. Reinforcement Learning-Based Near Optimization for Continuous-Time Markov Jump Singularly Perturbed Systems. *IEEE Trans. Circuits Syst. II, Exp. Briefs*; in press, doi: 10.1109/TCSII.2022.3233790.