SCIENCE CHINA
Information Sciences

July 2023, Vol. 66 170201:1–170201:25
https://doi.org/10.1007/s11432-022-3796-2

• RESEARCH PAPER •
Special Topic: Entanglement of Control and Optimization

Reinforcement learning for optimal tracking of large-scale systems with multitime scales

Jinna LI[1], Hao NIE[1], Tianyou CHAI[2*] & Frank L. LEWIS[3]

[1]*School of Information and Control Engineering, Liaoning Petrochemical University, Fushun* 113001*, China;*
[2]*State Key Laboratory of Synthetical Automation for Process Industries, Northeastern University, Shenyang* 110819*, China;*
[3]*UTA Research Institute, University of Texas at Arlington, Arlington* 76118*, USA*

Received 28 August 2022/Revised 16 March 2023/Accepted 10 May 2023/Published online 29 June 2023

Abstract This paper aims to solve an optimal tracking control (OTC) problem of large-scale systems with multitime scales and coupled subsystems using singular perturbation (SP) theory and reinforcement learning (RL) techniques. A considerable contribution of this paper is the development of a data-driven SP-based RL method for the OTC of unknown large-scale systems with multitime scales. To achieve this, a multitime scale tracking problem was decomposed into a linear quadratic tracker problem for slow subsystems and a dynamical game problem for fast subsystems using the SP theory. Then, the distributed composite feedback controllers were found using a distributed off-policy integral RL algorithm that uses only measured data from the system in real time. Thus, the operational index can follow its prescribed target value via an approximately optimal approach. Theoretical analysis and proof are presented to demonstrate that the sum of the performances of reduced-order subsystems is approximately equal to the performance of the original large-scale system. Finally, numerical and practical examples are provided to validate the effectiveness of the proposed method.

Keywords reinforcement learning, singular perturbation, multitime scales, data-driven control

Citation Li J N, Nie H, Chai T Y, et al. Reinforcement learning for optimal tracking of large-scale systems with multitime scales. Sci China Inf Sci, 2023, 66(7): 170201, https://doi.org/10.1007/s11432-022-3796-2

1 Introduction

With the development of society, increasing productivity, and continuous advancement of science and technology, large-scale systems in the field of modern engineering, such as electric power systems, robotic systems, communication networks, economic systems, traffic networks, and industrial process control systems [1–3] are becoming increasingly common. Large-scale systems refer to systems that comprise several interconnected local systems that may be coupled in some way to achieve a common performance goal. Conventional centralized control schemes [2] are unsuitable for such large-scale systems because of their high dimension of state variables, high computational complexity, and high search complexity of action space.

Solving the optimal control problem of complex large-scale systems is challenging due to the different time-scale characteristics exhibited by local subsystems, resulting in a multitime scale system [3–6]. In the practical process operation of large-scale industrial systems, plant-wide performance indicators often comprise a unit equipment layer with fast time scales and an operation indicator layer with slow time scales to form a global system with two time scales or more. It is desirable to find optimal set-points for the unit processes to ensure that the operational indices remain within their target ranges or at their desired target values when all units in the equipment layer adhere to the set-points [3, 4, 7]. To meet this requirement, the singular perturbation (SP) theory has been a prime candidate for analyzing and modeling systems with two time scales or more, after which the composite controller comprising the controllers of the subsystems can be designed using various control methods [8–14].

* Corresponding author (email: tychai@mail.neu.edu.cn)

It should be noted that existing SP-based control methods still have limitations, such as a lack of consideration for performance optimization of systems with multitime scales and a heavy reliance on accurate models of controlled systems. These constraints severely impede the performance enhancement of large-scale systems, and the existing methods [8–14] even fail to work for unknown large-scale systems with more than two time scales. In practical applications, the large scale, complex coupling relations, and multitime scales between subsystems make it impossible to model accurate system dynamics. Thus, developing data-driven SP-based optimal control methods to meet the desire for performance optimization by overcoming these limitations is crucial and is the main motivation of this paper.

Recently, the integration of case-based-reasoning intelligent control and reinforcement learning (RL) methods has demonstrated a promising prospect for studying data-driven optimal control for large-scale complex systems with multitime scales [15]. It is well known that RL is proven to be a powerful tool for determining optimal control for systems with unknown dynamics (see [16,17] and the references therein). More RL methods have been utilized to achieve optimal control of large-scale systems with multitime scales [18–22] without the need for system dynamics information. For optimal operational control (OOC) of industrial processes with two time scales, the RL algorithms merged with the SP theory were developed in [18,19] to identify the optimal control policies from the perspective of discrete-time and continuous-time domains using only data. In [20], the off-policy RL was used to address the OOC problem for nonlinear industrial operational processes. Refs. [21,22] considered a class of industrial systems comprising multiple unit devices and an unknown operational process, and the noncascade decentralized composite control methods were developed.

Notably, in the preceding RL-based research for control of multitime scale systems using data, some of them can work only for systems with one fast process and one slow process [18–20], and the others simply designed decentralized composite controllers without the concern of coupling among subsystems, although they exist in practice [21, 22]. To the best of our knowledge, data-driven distributed RL algorithms dedicated to achieving optimal tracking control (OTC) have received little attention. However, finding an effective way to solve this problem from a practical application standpoint is a pressing matter, even though it is considerably challenging due to the existence of mutual coupling among systems, multitime scales, and unknown system dynamics in large-scale systems.

The purpose of this paper is to combine SP and RL techniques to solve the OTC problem of large-scale systems with multitime scales, inner coupling relations, and unknown models. Motivated by the desire to solve this problem in an efficient and completely data-driven manner, a novel distributed composite controller design method is created using only measurable data.

The main contributions of this paper are summarized below.

(1) In contrast to [18–22], which only concern two time scales and decentralized control, this paper develops a novel distributed SP-based RL method for solving OTC problems of multitime-scale large-scale systems. The off-policy integral RL (IRL), the minmax strategy, and the actor-critic neural network (NN) structure are integrated for the first time with some mathematical manipulation, allowing the composite controller for performance optimization to be found without requiring the knowledge of systems dynamics.

(2) In fact, the longer the time scale, the more difficult it is to solve the OTC problem. Note that the data-driven SP method for large-scale systems with more than two time scales is currently unavailable. This paper introduces mathematical manipulations for successfully removing the obstacles caused by multitime scales and coupling relations among systems when performing SP decomposition.

(3) Theoretical analysis and proof that the designed composite controllers are capable of achieving OTC of multitime scale systems are presented.

The remainder of the paper is organized as follows. Section 2 defines the SP decomposition of large-scale systems with multitime scales. Section 3 describes the optimal control problems for global multitime scale systems as well as fast and slow separated subsystems. Based on Section 3, the RL technique is used to solve optimal control problems. Section 4 develops a data-driven distributed RL algorithm using the actor-critic NN architecture to find approximately optimal composite controllers capable of tracking the operational index using only measured data. Section 5 shows the results of two illustrative simulations. Finally, Section 6 concludes this article.

Notation. Π denotes a concatenated multiplicative relation. $\otimes$ represents the Kronecker product. $\|\cdot\|$ represents Euclidean norms. $\mathrm{Re}(\cdot)$ denotes the real part of the number, and U denotes a compact set $\{u_1, \ldots, u_N\}$. $\mathrm{Diag}(\cdot)$ denotes a diagonal matrix.

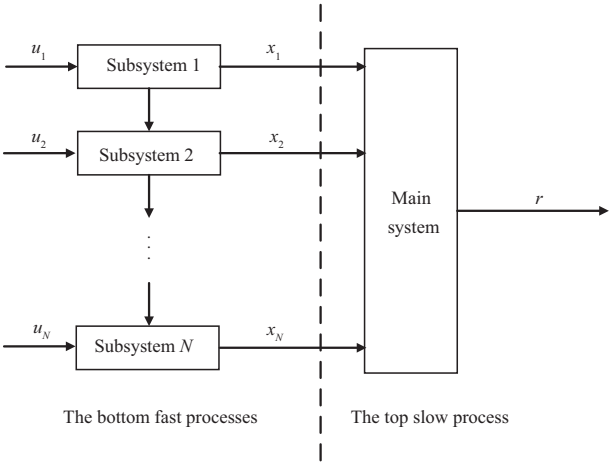

Figure 1 Structure diagram of a multitime-scale system.

2 Singular perturbation formulation for multitime scale systems

In this section, the multitime scale problem of large-scale systems is addressed. The SP theory is used to divide the systems into two parts—fast and slow subsystems.

2.1 Model of large-scale systems with multitime scales

Multitime scales operation is a common feature of practical industrial processes. The industrial operational processes comprise multiple unit device processes that are linked together and operate on different fast time scales of seconds. It is preferred that the operational indies adhere to desired target values for hours or longer.

Because unit devices in practical operational processes typically operate at some steady states, their nonlinear dynamics can be linearized near the steady operating points. Thus, the large-scale system with coupled multiple unit devices is defined as follows:

$$\begin{cases} \dot{x}_1(t) = A_{11}x_1(t) + B_{11}u_1(t), \\ \vdots \qquad\qquad\qquad\qquad\qquad\qquad\qquad\qquad i = 2, \ldots, N, \\ \dot{x}_i(t) = A_{ii}x_i(t) + A_{i,i-1}x_{i-1}(t) + B_{ii}u_i(t), \end{cases} \tag{1}$$

where $x_i(t) \in \mathbb{R}^{n_x}$ and $u_i(t) \in \mathbb{R}^{n_u}$ are the state and input of the ith subsystem, respectively. The fast state matrices A_{ii} and $A_{i,i-1}$ and the input matrices B_{ii} are assumed to be unknown with appropriate dimensions. N denotes the number of subsystems.

The linear dynamics of the operational index is given as follows:

$$\begin{cases} \dot{y}(t) = A_0 y(t) + \sum_{i=1}^{N} A_{0i}x_i(t), \\ r(t) = C_0 y(t), \end{cases} \tag{2}$$

where $y(t) \in \mathbb{R}^{n_y}$ is the state vector of the operational processes and $r(t) \in \mathbb{R}^{n_r}$ represents the operational index. The slow state matrix A_0, input matrices A_{0i}, and output matrix C_0 are assumed to be unknown with appropriate dimensions.

Remark 1. As illustrated in Figure 1, the large-scale system under consideration in this paper comprises a main operating system and multiple unit device subsystems. Unit devices are generally cascaded and run on different time scales in practical industrial processes, such as multiple reactors with cascade coupling in nonferrous metallurgy processes and cascade coupling screening, grinding, and magnetic separation units in mining processes. Furthermore, the operational index generally changes at a slower rate than the states of unit device processes [23, 24].

2.2 Singular perturbation for multitime scale separation

Since multitime scale characteristics of the global system (1) and (2) imply that the variables $x_i(t)$ change faster than the variable $y(t)$, there is a so-called gap in changing rate between the multiple unit devices and the operational processes. Here, we introduce small time-scale parameters ε_i and define $x_i(t) = \varepsilon_i z_i(t)$. These parameters ε_i could be small time constants, inertias, or masses in practice, and they explicitly demonstrate the different changing rates of the slow process (2). Substituting these new variables into the global system (1) and (2) yields a new singularly perturbed system as follows:

$$\begin{cases} \varepsilon_1 \dot{z}_1(t) = \hat{A}_{11} z_1(t) + B_{11} u_1(t), \\ \vdots \qquad\qquad\qquad\qquad\qquad\qquad i = 2, \ldots, N, \\ \varepsilon_i \dot{z}_i(t) = \hat{A}_{ii} z_i(t) + \hat{A}_{i,i-1} z_{i-1}(t) + B_{ii} u_i(t), \end{cases} \tag{3}$$

$$\begin{cases} \dot{y}(t) = A_0 y(t) + \sum_{i=1}^{N} \hat{A}_{0i} z_i(t), \\ r(t) = C_0 y(t), \end{cases} \tag{4}$$

where $\hat{A}_{ii} = \varepsilon_i A_{ii}$, $\hat{A}_{i,i-1} = \varepsilon_{i-1} A_{i,i-1}$, and $\hat{A}_{0i} = \varepsilon_i A_{0i}$.

Using classical SP theory, we will decompose the singularly perturbed systems (3) and (4) as the approximate fast and slow subsystems, respectively. Therefore, $u_i(t)$ and $z_i(t)$ can be defined as $u_i(t) = u_{is}(t) + u_{if}(t)$ and $z_i(t) = z_{is}(t) + z_{if}(t)$, where $u_{is}(t)$ and $z_{is}(t)$ are the slow components of the system variables, while $u_{if}(t)$ and $z_{if}(t)$ denote the fast components of the system variables. To separate the kth ($k \neq i$ and $k \neq 1$) subsystem from system (3), setting $\varepsilon_i = 0$ ($i = 1, 2, \ldots, k-1, k+1, \ldots, N$) yields

$$\varepsilon_k \dot{z}_k(t) = \hat{A}_{kk} z_k(t) + \hat{A}_{k,k-1} z_{k-1}(t) + B_{kk} u_k(t), \tag{5a}$$

$$0 = \hat{A}_{ii} z_i(t) + \hat{A}_{i,i-1} z_{i-1}(t) + B_{ii} u_i(t), \tag{5b}$$

where $z_0(t) = 0$ and $\hat{A}_{10} = I$ (I is the identity matrix). Eq. (5b) can be specifically represented as follows:

$$\begin{cases} 0 = \hat{A}_{11} z_1(t) + B_{11} u_1(t), \\ 0 = \hat{A}_{22} z_2(t) + \hat{A}_{21} z_1(t) + B_{22} u_2(t), \\ 0 = \hat{A}_{33} z_3(t) + \hat{A}_{32} z_2(t) + B_{33} u_3(t), \\ \vdots \\ 0 = \hat{A}_{k-1,k-1} z_{k-1}(t) + \hat{A}_{k-1,k-2} z_{k-2}(t) \\ \qquad + B_{k-1,k-1} u_{k-1}(t), \\ 0 = \hat{A}_{k+1,k+1} z_{k+1}(t) + \hat{A}_{k+1,k} z_k(t) \\ \qquad + B_{k+1,k+1} u_{k+1}(t), \\ \vdots \\ 0 = \hat{A}_{NN} z_N(t) + \hat{A}_{N,N-1} z_{N-1}(t) + B_{NN} u_N(t). \end{cases} \tag{6}$$

The following general assumption is expressed like [25].

Assumption 1. Matrices A_{ii}, $i = 1, \ldots, N$ are nonsingular.

Now, the quasi-steady-state values of the various unit devices can be calculated using (6) as follows:

$$z_1(t) = -\hat{A}_{11}^{-1} B_{11} u_1(t), \tag{7}$$

$$\begin{aligned} z_2(t) =& -\hat{A}_{22}^{-1} [\hat{A}_{21} (-\hat{A}_{11}^{-1} B_{11} u_1(t)) + B_{22} u_2(t)] \\ =& \hat{A}_{22}^{-1} \hat{A}_{21} \hat{A}_{11}^{-1} B_{11} u_1(t) - \hat{A}_{22}^{-1} B_{22} u_2(t), \end{aligned} \tag{8}$$

$$\begin{aligned} z_3(t) =& -\hat{A}_{33}^{-1} (\hat{A}_{32} z_2(t) + B_{33} u_3(t)) \\ =& -\hat{A}_{33}^{-1} \hat{A}_{32} \hat{A}_{22}^{-1} \hat{A}_{21} \hat{A}_{11}^{-1} B_{11} u_1(t) \end{aligned}$$

$$+\hat{A}_{33}^{-1}\hat{A}_{32}\hat{A}_{22}^{-1}B_{22}u_2(t)-\hat{A}_{33}^{-1}B_{33}u_3(t), \tag{9}$$

$$\vdots$$

$$\begin{aligned}z_{k-1}(t)=&(-1)^{k-1}\prod_{i=k-1}^{1}(\hat{A}_{ii}^{-1}\hat{A}_{i,i-1})B_{11}u_1(t)\\&+(-1)^{k-2}\prod_{i=k-1}^{2}(\hat{A}_{ii}^{-1}\hat{A}_{i,i-1})B_{22}u_2(t)\\&+\cdots+(-1)^1\hat{A}_{k-1,k-1}^{-1}B_{k-1,k-1}u_{k-1}(t).\end{aligned} \tag{10}$$

Then, Eq. (10) can be rewritten as follows:

$$z_{k-1}(t)=\sum_{j=1}^{k-1}\bar{B}_{k-1,j}u_j(t), \tag{11}$$

where $\bar{B}_{k-1,j}=(-1)^{k-j}\prod_{i=k-1}^{j}(\hat{A}_{ii}^{-1}\hat{A}_{i,i-1})\hat{A}_{j,j-1}^{-1}B_{jj}, j\leqslant k-1$.

Similarly, it follows

$$z_{k+1}(t)=-\hat{A}_{k+1,k+1}^{-1}\hat{A}_{k+1,k}z_k(t)-\hat{A}_{k+1,k+1}^{-1}B_{k+1,k+1}u_{k+1}(t), \tag{12}$$

$$\begin{aligned}z_{k+2}(t)=&\hat{A}_{k+2,k+2}^{-1}\hat{A}_{k+2,k+1}\hat{A}_{k+1,k+1}^{-1}\hat{A}_{k+1,k}z_k(t)\\&+\hat{A}_{k+2,k+2}^{-1}\hat{A}_{k+2,k+1}\hat{A}_{k+1,k+1}^{-1}B_{k+1,k+1}u_{k+1}(t)\\&-\hat{A}_{k+2,k+2}^{-1}B_{k+2,k+2}u_{k+2}(t),\end{aligned} \tag{13}$$

$$\vdots$$

$$\begin{aligned}z_N(t)=&(-1)^{N-k}\prod_{i=N}^{k+1}(\hat{A}_{ii}^{-1}\hat{A}_{i,i-1})z_k(t)\\&+(-1)^{N-k}\prod_{i=N}^{k+1}(\hat{A}_{ii}^{-1}\hat{A}_{i,i-1})\hat{A}_{k+1,k}^{-1}B_{k+1,k+1}u_{k+1}(t)\\&+\cdots-\hat{A}_{NN}^{-1}B_{NN}u_N(t).\end{aligned} \tag{14}$$

Then, there are the following general forms:

$$z_p(t)=(-1)^{p-k}\prod_{i=p}^{k+1}(\hat{A}_{ii}^{-1}\hat{A}_{i,i-1})z_k(t)+\sum_{j=k+1}^{p}\bar{N}_{p,j}u_j(t), \tag{15}$$

where $\bar{N}_{p,j}=(-1)^{p-j+1}\prod_{i=p}^{j}(\hat{A}_{ii}^{-1}\hat{A}_{i,i-1})\hat{A}_{j,j-1}^{-1}B_{jj}(j\leqslant p)$, $p=k+1,k+2,\ldots,N$.

Substituting (11) into (5a) yields

$$\varepsilon_k\dot{z}_k(t)=\hat{A}_{kk}z_k(t)+\hat{A}_{k,k-1}\sum_{i=1}^{k-1}\bar{B}_{k-1,i}u_i(t)+B_{kk}u_k(t). \tag{16}$$

Like [8,10,21], letting $\varepsilon_k=0$ in (16) yields

$$z_{ks}(t)=-\hat{A}_{kk}^{-1}\left(\hat{A}_{k,k-1}\sum_{i=1}^{k-1}\bar{B}_{k-1,i}u_{is}(t)+B_{kk}u_{ks}(t)\right), \tag{17}$$

where matrix A_{kk} is invertible as assumed in Assumption 1. Substituting (11), (15), and (17) into (4) produces the kth reduced-order slow subsystem dynamics as follows:

$$\dot{y}_{ks}(t)=A_0y_{ks}(t)-\tilde{A}_{0k}\hat{A}_{kk}^{-1}B_{kk}u_{ks}(t)+\sum_{j=1}^{k-1}(\tilde{B}_{0j}-\tilde{A}_{0k}\hat{A}_{kk}^{-1}\hat{A}_{k,k-1}\bar{B}_{k-1,j})u_{js}(t)$$

$$+\sum_{j=k+1}^{N}\tilde{C}_{0j}u_{js}(t),\quad k\neq 1, \tag{18}$$

where $\tilde{B}_{0j}=(\sum_{i=k-1}^{1}\hat{A}_{0i}\bar{B}_{i,j})$, $\tilde{C}_{0j}=(\sum_{i=N}^{k+1}\hat{A}_{0i}\bar{N}_{ij})$, and $\tilde{A}_{0k}=(\hat{A}_{0k}+\sum_{i=k+1}^{N}\hat{A}_{0i}((-1)^{i-k}\prod_{w=i}^{k+1}(\hat{A}_{ww}^{-1}\cdot\hat{A}_{w,w-1})))$.

Similarly, the derivation of the kth ($k=1$) slow subsystem can be derived as follows:

$$\dot{y}_{ks}(t)=A_0y_{ks}(t)-\tilde{A}_{01}\hat{A}_{11}^{-1}B_{11}u_{1s}(t)+\sum_{i=2}^{N}\tilde{C}_{0i}u_{is}(t),\quad k=1. \tag{19}$$

Based on the donations of parameters $\tilde{A}_{0k},\hat{A}_{kk}^{-1},\hat{A}_{k,k-1},B_{kk},\tilde{B}_{0i},\bar{B}_{k-1,i}$, and $\tilde{C}_{0i}$ in (18) and (19), one can observe that the dynamics is the same dynamics regardless of the taken value of k. Thus, the dynamics of the slow subsystem is as follows:

$$\begin{cases}\dot{y}_s(t)=A_0y_s(t)-\tilde{A}_{01}\hat{A}_{11}^{-1}B_{11}u_{1s}(t)+\sum_{i=2}^{N}\tilde{C}_{0i}u_{is}(t),\\ r_s(t)=C_0y_s(t).\end{cases} \tag{20}$$

For the kth fast subsystem, the slow variable z_{ks} in the fast time scale s_k is actually a constant, where $s_k=\frac{t}{\varepsilon_k}$. As a result, it follows $\dot{z}_{ks}(t)=0$. By (16), the dynamics of the kth fast subsystem is given as follows:

$$\begin{aligned}\varepsilon_k\dot{z}_k&=\frac{\mathrm{d}z_k(t)}{\mathrm{d}s_k}=\frac{\mathrm{d}(z_{kf}(t)+z_{ks}(t))}{\mathrm{d}s_k}=\frac{\mathrm{d}z_{kf}(t)}{\mathrm{d}s_k}\\&=\hat{A}_{kk}z_{kf}+\hat{A}_{k,k-1}\sum_{i=1}^{k-1}\bar{B}_{k-1,i}u_{if}+B_{kk}u_{kf},\end{aligned} \tag{21}$$

which is equivalent to the following form:

$$\dot{z}_{kf}(s_k)=\hat{A}_{kk}z_{kf}(s_k)+\hat{A}_{k,k-1}\sum_{i=1}^{k-1}\bar{B}_{k-1,i}u_{if}(s_k)+B_{kk}u_{kf}(s_k). \tag{22}$$

Remark 2. When compared to two-time scale systems [18–22], the complexity of (1) and (2) lies in the inner coupling among subsystems except for more than two time scales. The above decomposition method, combined with mathematical manipulations, results in the separation of fast and slow variables, resulting in reduced-order subsystems.

3 Reinforcement learning for optimal tracking control

In this section, a linear command generator system is presented for generating a reference trajectory. Then, we convert the OTC problem of multitime-scale large-scale systems to a dynamical game and the linear quadratic tracker (LQT) problems of reduced-order subsystems. Furthermore, the performance of the global system is explicitly analyzed by the composite controllers. Finally, an off-policy IRL algorithm is developed to find the best composite controllers.

3.1 Formulation of the optimal tracking control problem

The reference trajectory is assumed to be described by a linear command generator as follows:

$$\dot{r}^*(t)=Fr^*(t), \tag{23}$$

where $r^*(t)\in\mathbb{R}^{n_r}$ is the dynamical trajectory vector and F is a constant square matrix of appropriate dimension. We generally assume that Eq. (23) is unstable [26].

The goal of this paper is to design the optimal policies $\{u_1,u_2,\ldots,u_N\}$ for the original system (1) and (2) under which the operational index r can follow the prescribed value r^* via an approximately optimal approach. To this end, Problem 1 is formulated below.

Problem 1. OTC problem: For the global system comprising (1) and (2), it is desirable to determine the tuple of control policies u_k^* $(k = 1, \ldots, N)$ such that the following performance indices can be minimized in a finite horizon, that is

$$\begin{aligned} J_k(y, r^*, x_k) = \min_{u_k} \int_t^{tf} \Bigg[& (x_k - x_{ks})^{\mathrm{T}} Q_{kf} (x_k - x_{ks}) + (r - r^*)^{\mathrm{T}} Q_0 (r - r^*) \\ & + \sum_{i=1}^{k-1} u_i^{\mathrm{T}} R_i u_i + u_k^{\mathrm{T}} R_k u_k + \sum_{i=k+1}^{N} u_{is}^{\mathrm{T}} R_{is} u_{is} \Bigg] \mathrm{d}\tau + \Psi(y(tf), r^*(tf), x_k(tf)), \\ \text{s.t.} \quad & (1) \text{ and } (2), \end{aligned} \tag{24}$$

where $u_k = [u_{kf}^{\mathrm{T}} \ u_{ks}^{\mathrm{T}}]^{\mathrm{T}}$, $R_k = \mathrm{diag}\ \{R_{kf}, R_{ks}\}$, $Q_0 \geqslant 0$, and $Q_{kf} \geqslant 0$ are all symmetric matrices, $J_k(y(tf), r^*(tf), x_k(tf)) = \Psi(y(tf), r^*(tf), x_k(tf)) \geqslant 0$ represents the terminal constraint, and tf represents the terminal time instant.

It is worth noting that Problem 1 is written using the global systems (1) and (2). Since the SP theory is used and the fast and slow subsystems are derived (see (20) and (22)), Problems 2 and 3 are presented such that solving Problem 1 can be replaced by solving Problems 2 and 3.

Problem 2. Game problem: For the kth $(k = 1, 2, \ldots, N)$ fast subsystem, it is desirable to identify the control policy u_{kf}^* such that the following performance index can be minimized in a finite horizon; that is

$$\begin{aligned} J_{kf}(z_{kf}) = \min_{u_{kf}} \int_{s_k}^{s_{ktf}} & \left(z_{kf}^{\mathrm{T}} \hat{Q}_{kf} z_{kf} + \sum_{i=1}^{k-1} u_{if}^{\mathrm{T}} \hat{R}_{if} u_{if} + u_{kf}^{\mathrm{T}} \hat{R}_{kf} u_{kf} \right) \mathrm{d}\tau_{kf}' \\ & + \Psi_{kf}(z_{kf}(s_{ktf})), \\ \text{s.t.} \quad & (22), \end{aligned} \tag{25}$$

where $\hat{Q}_{kf} = \varepsilon_k^3 Q_{kf}$, $\hat{R}_{kf} = \varepsilon_k R_{kf}$, $\hat{R}_{if} = \varepsilon_i R_{if}$, $\tau_{kf}' \varepsilon_k = \tau$, $s_{ktf} = \frac{tf}{\varepsilon_k}$, and the terminal performance $J_{kf}(z_{kf}(s_{ktf})) = \Psi_{kf}(z_{kf}(s_{ktf}))$.

Problem 3. LQT problem: For the slow subsystem, it is desirable to find the control policies $\{u_{1s}^*, \ldots, u_{Ns}^*\}$, such that the following performance index can be minimized in a finite horizon; that is

$$\begin{aligned} J_s(y_s, r^*) = \min_{\{u_{1s}, u_{2s}, \ldots, u_{Ns}\}} \int_t^{tf} & \left[(C_0 y_s - r^*)^{\mathrm{T}} Q_0 (C_0 y_s - r^*) + \sum_{i=1}^{N} u_{is}^{\mathrm{T}} R_{is} u_{is} \right] \mathrm{d}\tau \\ & + \Psi_s(y_s(tf), r^*(tf)), \\ \text{s.t.} \quad & (20), \end{aligned} \tag{26}$$

with the terminal performance constraint $J_s(y_s(tf), r^*(tf)) = \Psi_s(y_s(tf), r^*(tf))$.

When Eqs. (20) and (23) are combined, one obtains the augmented dynamics of the reduced-order slow subsystem given as follows:

$$\begin{cases} \dot{\bar{Y}}(t) = A\bar{Y}(t) + \sum_{i=1}^{N} B_{is} u_{is}(t) = A\bar{Y}(t) + B_s u_s(t), \\ r_s(t) = C_s \bar{Y}(t), \end{cases} \tag{27}$$

where $\bar{Y} = [y_s^{\mathrm{T}}, r^{*\mathrm{T}}]^{\mathrm{T}}$, $\bar{Y} \in \mathbb{R}^{n_{\bar{Y}}}$ with $n_{\bar{Y}} = n_y + n_r$, $C_s = [C_0\ 0]$, $B_s = [B_{1s} \ \cdots \ B_{Ns}]$, $u_s = [u_{1s}^{\mathrm{T}} \ \cdots \ u_{Ns}^{\mathrm{T}}]^{\mathrm{T}}$,

$$A = \begin{bmatrix} A_0 & 0 \\ 0 & F \end{bmatrix}, \quad B_{is} = \begin{bmatrix} \wedge_{is} \\ 0 \end{bmatrix}, \quad \wedge_{is} = \begin{cases} \tilde{B}_{0i} - \tilde{A}_{01} \hat{A}_{11}^{-1} \bar{B}_{0i}, & i = 1, \\ \tilde{C}_{0i}, & 2 \leqslant i \leqslant N. \end{cases}$$

Thus, Eq. (26) can be rewritten as follows:

$$\begin{aligned} J_s(\bar{Y}) = \min_{\{u_{1s}, u_{2s}, \ldots, u_{Ns}\}} \int_t^{tf} & \left[\bar{Y}^{\mathrm{T}} Q_s \bar{Y} + \sum_{i=1}^{N} u_{is}^{\mathrm{T}} R_{is} u_{is} \right] \mathrm{d}\tau + \bar{\Psi}_s(\bar{Y}(tf)), \\ \text{s.t.} \quad & (27), \end{aligned} \tag{28}$$

where the terminal performance $J_s(\bar{Y}(tf)) = \bar{\Psi}_s(\bar{Y}(tf))$ and $Q_s = [C_0 \ -I]^{\mathrm{T}} Q_0 [C_0 \ -I]$ is a positive semidefinite matrix.

Remark 3. Notice that Problem 2 is particularly a multiagent dynamical game problem with each fast subsystem acting as an agent attempting to minimize its own performance. However, the performance of each agent is affected by other agents due to their intercoupling relation. The optimization problem (28) is specifically an LQT problem with multiple inputs, which means that all control policies $\{u_{1s}, u_{2s}, \ldots, u_{Ns}\}$ aim to drive the operational index r to the desired value r^*. This is an extension of the LQT problem [27, 28] in which a single control policy works for tracking the desired trajectory.

The following result demonstrates that the solutions to Problems 2 and 3 provide the solution to Problem 1 for the global system depicted in (1) and (2). Tikhonov's theorem in [8] yields

$$\begin{cases} z_k(t) = z_{kf}(s_k) + z_{ks}(t) + o(\varepsilon_k), \\ u_k(t) = u_{ks}(t) + u_{kf}(s_k) + o(\varepsilon_k), \\ y(t) = y_s(t) + o(\varepsilon), \end{cases} \tag{29}$$

where $z_{ks}(t)$ is defined in (17) and $o(\varepsilon)$ is an infinitesimal of ε ($\varepsilon = \max_{\varepsilon_k}\{\varepsilon_1, \varepsilon_2, \ldots, \varepsilon_N\}$); then Eqs. (20), (22), and (29) give

$$\begin{aligned} x_k(t) =& \varepsilon_k z_{kf}(s_k) + \varepsilon_k z_{ks}(t) + o(\varepsilon_k) \\ =& x_{kf}(s_k) + x_{ks}(t) + o(\varepsilon_k), \end{aligned} \tag{30}$$

$$\begin{aligned} r(t) =& C_0 y_s(t) + o(\varepsilon) \\ =& r_s(t) + o(\varepsilon). \end{aligned} \tag{31}$$

Thus, when the slow (20) and fast subsystems (22) are stable, the global systems (1) and (2) are stable. This indicates that the composite controllers $u_{kc} = u_{ks} + u_{kf}$ can make the operational index r follow the reference trajectory r^*, if u_{ks} and u_{kf} can track the target and stabilize the reduced-order subsystems, respectively.

Theorem 1. For the cost function (24) of the global system, the cost function (25) of the fast boundary layer subsystem, and the cost function (28) of the reduced-order slow subsystem, the relation $J_k = J_{kf} + J_s + o(\varepsilon)$ holds.

Proof. Considering $s_k \varepsilon_k = t$, $\tau'_{kf} \varepsilon_k = \tau$, and (29)–(31),

$$\begin{aligned} J_{kf} =& \int_t^{tf} \left[z_{kf}(\tau)^{\mathrm{T}} \frac{\hat{Q}_{kf}}{\varepsilon_k} z_{kf}(\tau) + \sum_{i=1}^{k-1} u_{if}^{\mathrm{T}}(\tau) R_{if} u_{if}(\tau) \right. \\ & \left. + u_{kf}^{\mathrm{T}}(\tau) R_{kf} u_{kf}(\tau) \right] \mathrm{d}\tau + \Psi_{kf}(z_{kf}(\tau)) \\ =& \int_t^{tf} \left[(x_k - x_{ks})^{\mathrm{T}} Q_{kf} (x_k - x_{ks}) + \sum_{i=1}^{k-1} u_{if}^{\mathrm{T}} R_{if} u_{if} \right. \\ & \left. + u_{kf}^{\mathrm{T}} R_{kf} u_{kf} \right] \mathrm{d}\tau + \Psi_{kf}(x_{kf}(tf)) - o(\varepsilon). \end{aligned} \tag{32}$$

Then, one has

$$\begin{aligned} J_s =& \int_t^{tf} \left[\bar{Y}(\tau)^{\mathrm{T}} Q_s \bar{Y}(\tau) + \sum_{i=1}^{N} u_{is}^{\mathrm{T}}(\tau) R_{is} u_{is}(\tau) \right] \mathrm{d}\tau + \bar{\Psi}_s(\bar{Y}(tf)) \\ =& \int_t^{tf} \left[(C_0 y(\tau) - r^*)^{\mathrm{T}} Q_0 (C_0 y(\tau) - r^*) + \sum_{i=1}^{N} u_{is}^{\mathrm{T}} R_{is} u_{is} \right] \mathrm{d}\tau \\ & + \Psi_s(y(tf), r^*(tf)) - o(\varepsilon), \end{aligned} \tag{33}$$

where $\Psi_{kf}(x_{kf}(tf)) + \Psi_s(y(tf), r^*(tf)) = \Psi(y(tf), r^*(tf), x_i(tf))$. Thus, it follows

$$J_{kf} + J_s = J_k - o(\varepsilon), \tag{34}$$

which gives $J_k = J_{kf} + J_s + o(\varepsilon)$. This completes the proof.

3.2 Algorithm design of off-policy minmax reinforcement learning

Now, we are going to solve Problems 2 and 3. To solve Problem 2, the value functions are defined in terms of (25) as follows:

$$V_{kf}(z_{kf}, s_k) = \int_{s_k}^{s_{ktf}} \Bigg[z_{kf}^{\mathrm{T}} \hat{Q}_{kf} z_{kf} + u_{kf}^{\mathrm{T}} \hat{R}_{kf} u_{kf} + \sum_{i=1}^{k-1} u_{if}^{\mathrm{T}} \hat{R}_{if} u_{if} \Bigg] \mathrm{d}\tau'_{kf} + \Psi_{kf}(z_{kf}(s_{ktf})). \tag{35}$$

An infinitesimal comparable to (35) is given by depicting [29] as follows:

$$\frac{\partial V_{kf}}{\partial s_k} + \frac{\partial V_{kf}^{\mathrm{T}}}{\partial z_{kf}} \left(\hat{A}_{kk} z_{kf} + \hat{A}_{k,k-1} \sum_{i=1}^{k-1} \bar{B}_{k-1,i} u_{if} + B_{kk} u_{kf} \right) + r_{kf} = 0, \tag{36}$$

where $r_{kf} = z_{kf}^{\mathrm{T}} \hat{Q}_{kf} z_{kf} + u_{kf}^{\mathrm{T}} \hat{R}_{kf} u_{kf} + \sum_{i=1}^{k-1} u_{if}^{\mathrm{T}} \hat{R}_{if} u_{if}$.

Subsequently, the following Hamiltonian functions are given:

$$\begin{aligned} H_{kf} =& r_{kf} + \frac{\partial V_{kf}^{\mathrm{T}}}{\partial z_{kf}} \frac{\mathrm{d}z_{kf}}{\mathrm{d}s_k} + \frac{\partial V_{kf}}{\partial s_k} \\ =& r_{kf} + \frac{\partial V_{kf}^{\mathrm{T}}}{\partial z_{kf}} \left(\hat{A}_{kk} z_{kf} + \hat{A}_{k,k-1} \sum_{i=1}^{k-1} \bar{B}_{k-1,i} u_{if} + B_{kk} u_{kf} \right) + \frac{\partial V_{kf}}{\partial s_k}. \end{aligned} \tag{37}$$

Implementing $\frac{\partial H_{kf}}{\partial u_{kf}} = 0$ yields

$$u_{kf}^* = -\frac{1}{2} \hat{R}_{kf}^{-1} B_{kk}^{\mathrm{T}} \frac{\partial V_{kf}^*}{\partial z_{kf}}. \tag{38}$$

Substituting (38) into (36) yields the coupled Hamilton-Jacobi-Bellman (HJB) equations as follows:

$$\begin{aligned} &\left(\frac{\partial V_{kf}^*}{\partial z_{kf}} \right)^{\mathrm{T}} \hat{A}_{kk} z_{kf} - \frac{1}{2} \left(\frac{\partial V_{kf}^*}{\partial z_{kf}} \right)^{\mathrm{T}} \hat{A}_{k,k-1} \sum_{i=1}^{k-1} \bar{B}_{k-1,i} \hat{R}_{if}^{-1} B_{ii}^{\mathrm{T}} \frac{\partial V_{if}^*}{\partial z_{if}} + z_{kf}^{\mathrm{T}} \hat{Q}_{kf} z_{kf} \\ &\quad + \frac{1}{4} \sum_{i=1}^{k-1} \left(\frac{\partial V_{if}^*}{\partial z_{if}} \right)^{\mathrm{T}} B_{ii} \hat{R}_{if}^{-1} B_{ii}^{\mathrm{T}} \frac{\partial V_{if}^*}{\partial z_{if}} - \frac{1}{4} \left(\frac{\partial V_{kf}^*}{\partial z_{kf}} \right)^{\mathrm{T}} B_{kk} \hat{R}_{kf}^{-1} B_{kk}^{\mathrm{T}} \frac{\partial V_{kf}^*}{\partial z_{kf}} + \frac{\partial V_{kf}^*}{\partial s_k} = 0, \\ &V_{kf}^*(z_{kf}, s_{ktf}) = \Psi_{kf}(z_{kf}(s_{ktf})). \end{aligned} \tag{39}$$

Note that the coupled HJB equations are partial differential equations (PDE), and there exists a mutual coupling between V_{kf}^* and V_{if}^*. Notably, there may not exist solutions V_{kf}^* to the coupled HJB equations (39), because agent k cannot make its best response without the information z_{if} $(i = 1, 2, \ldots, k-1)$ of its neighbors [30].

To solve Problem 2 using a model-free approach, a novel off-policy minmax-based RL algorithm is developed. To achieve this, the performance index (25) is modified for formulating a zero-sum game in which agent k pretends its neighbors to be adverse.

$$J_{kf}(z_{kf}) = \min_{u_{kf}} \max_{\{u_{1f}, \ldots, u_{k-1f}\}} \int_{s_k}^{s_{ktf}} \Bigg[z_{kf}^{\mathrm{T}} \hat{Q}_{kf} z_{kf} + u_{kf}^{\mathrm{T}} \hat{R}_{kf} u_{kf} - \gamma_k^2 \sum_{i=1}^{k-1} u_{if}^{\mathrm{T}} \hat{R}_{if} u_{if} \Bigg] \mathrm{d}\tau'_{kf} + \Psi_{kf}(z_{kf}(s_{ktf})), \tag{40}$$

where γ_k is a positive scalar.

Based on (40), consider the value function of a fast subsystem as follows:

$$V_{kf}(z_{kf}, s_k) = \min_{u_{kf}} \max_{\{u_{1f}, \ldots, u_{k-1f}\}} \int_{s_k}^{s_{ktf}} \Bigg[z_{kf}^{\mathrm{T}} \hat{Q}_{kf} z_{kf} + u_{kf}^{\mathrm{T}} \hat{R}_{kf} u_{kf}$$

$$- \gamma_k^2 \sum_{i=1}^{k-1} u_{if}^{\mathrm{T}} \hat{R}_{if} u_{if} \Bigg] \mathrm{d}\tau'_{kf} + \Psi_{kf}(z_{kf}(s_{ktf})). \tag{41}$$

The Hamiltonian function related to the cost index (41) is defined as follows:

$$\begin{aligned} \hat{H}_{kf} =& \hat{r}_{kf} + \frac{\partial V_{kf}^{\mathrm{T}}}{\partial z_{kf}} \frac{\mathrm{d}z_{kf}}{\mathrm{d}s_k} + \frac{\partial V_{kf}}{\partial s_k} \\ =& \hat{r}_{kf} + \frac{\partial V_{kf}^{\mathrm{T}}}{\partial z_{kf}} \left(\hat{A}_{kk} z_{kf} + \hat{A}_{k,k-1} \sum_{i=1}^{k-1} \bar{B}_{k-1,i} u_{if} + B_{kk} u_{kf} \right) + \frac{\partial V_{kf}}{\partial s_k}, \end{aligned} \tag{42}$$

where $\hat{r}_{kf} = z_{kf}^{\mathrm{T}} \hat{Q}_{kf} z_{kf} + u_{kf}^{\mathrm{T}} \hat{R}_{kf} u_{kf} - \gamma_k^2 \sum_{i=1}^{k-1} u_{if}^{\mathrm{T}} \hat{R}_{if} u_{if}$.

The worst-case policy of the neighbors of agent k can be calculated using the stationary condition $\frac{\partial \hat{H}_{kf}}{\partial u_{if}} = 0$ as follows:

$$v_{if}^* = \frac{1}{2\gamma_k^2} \hat{R}_{if}^{-1} \bar{B}_{k-1,i}^{\mathrm{T}} \hat{A}_{k,k-1}^{\mathrm{T}} \frac{\partial V_{kf}^*}{\partial z_{kf}}. \tag{43}$$

Note that v_{if}^* is not always the actual control policy u_{if} used by agent i.

Substituting policies (38) and (43) into (42) yields the new HJB equations as follows:

$$\begin{aligned} &\left(\frac{\partial V_{kf}^*}{\partial z_{kf}} \right)^{\mathrm{T}} \hat{A}_{kk} z_{kf} + \frac{1}{4\gamma_k^2} \left(\frac{\partial V_{kf}^*}{\partial z_{kf}} \right)^{\mathrm{T}} \hat{A}_{k,k-1} \sum_{i=1}^{k-1} \bar{B}_{k-1,i} \hat{R}_{if}^{-1} B_{k-1,i}^{\mathrm{T}} \hat{A}_{k,k-1}^{\mathrm{T}} \frac{\partial V_{kf}^*}{\partial z_{kf}} \\ &\quad + z_{kf}^{\mathrm{T}} \hat{Q}_{kf} z_{kf} - \frac{1}{4} \left(\frac{\partial V_{kf}^*}{\partial z_{kf}} \right)^{\mathrm{T}} B_{kk} \hat{R}_{kf}^{-1} B_{kk}^{\mathrm{T}} \frac{\partial V_{kf}^*}{\partial z_{kf}} + \frac{\partial V_{kf}^*}{\partial s_k} = 0, \\ &V_{kf}(z_{kf}, s_{ktf}) = \Psi_{kf}(z_{kf}(s_{ktf})). \end{aligned} \tag{44}$$

Remark 4. Equations in the form of (44) are known to have positive definite solutions V_{kf}^*, where (A_{kk}, B_{kk}) are assumed to be stabilizable, $(A_{kk}, \sqrt{\hat{Q}_{kk}})$ are observable, and γ_k are large enough. Notably, the subsystems (22) are L_2 stable under minmax policies (38) if V_{kf}^* satisfy the HJB equations (44), as demonstrated in [30].

Then, we present the auxiliary variables $v_{if}^{(j_f)}$ $(i = 1, 2, \ldots, k-1)$ and $u_{kf}^{(j_f)}$ into the kth fast subsystem (22) and the following is obtained:

$$\begin{aligned} \dot{z}_{kf} =& \hat{A}_{kk} z_{kf} + \sum_{i=1}^{k-1} \hat{A}_{k,k-1} \bar{B}_{k-1,i} v_{if}^{(j_f)} + B_{kk} u_{kf}^{(j_f)} \\ &+ \sum_{i=1}^{k-1} \hat{A}_{k,k-1} \bar{B}_{k-1,i} (u_{if} - v_{if}^{(j_f)}) + B_{kk} (u_{kf} - u_{kf}^{(j_f)}), \end{aligned} \tag{45}$$

where u_{if} and u_{kf} are the behavior policies used to generate data and $v_{if}^{(j_f)}$ and $u_{kf}^{(j_f)}$ are viewed as the target policies that need to be updated. By separating $V_{kf}^{(j_f+1)}$ from the dynamics of the kth fast subsystem (22) and considering (36), one obtains

$$\begin{aligned} \dot{V}_{kf}^{(j_f+1)} &= \left(\frac{\partial V_{kf}^{(j_f+1)}}{\partial z_{kf}} \right)^{\mathrm{T}} \frac{\mathrm{d}z_{kf}}{\mathrm{d}s_k} + \frac{\partial V_{kf}^{(j_f+1)}}{\partial s_k} \\ &= \left(\frac{\partial V_{kf}^{(j_f+1)}}{\partial z_{kf}} \right)^{\mathrm{T}} \left(\hat{A}_{kk} z_{kf} + \hat{A}_{k,k-1} \sum_{i=1}^{k-1} \bar{B}_{k-1,i} v_{if}^{(j_f)} + B_{kk} u_{kf}^{(j_f)} \right) + \frac{\partial V_{kf}^{(j_f+1)}}{\partial s_k} \\ &= - r_{kf}^{(j_f)}. \end{aligned} \tag{46}$$

Furthermore, integrating both sides of (46) from $\frac{t}{\varepsilon_k}$ to $\frac{t+T}{\varepsilon_k}$ yields the following equality based on (45):

$$\int_{\frac{t}{\varepsilon_k}}^{\frac{t+T}{\varepsilon_k}} \dot{V}_{kf}^{(j_f+1)} \mathrm{d}\tau'_{kf} = \int_{\frac{t}{\varepsilon_k}}^{\frac{t+T}{\varepsilon_k}} \Bigg[\left(\frac{\partial V_{kf}^{(j_f+1)}}{\partial z_{kf}} \right)^{\mathrm{T}} \left(\frac{\mathrm{d}z_{kf}}{\mathrm{d}s_k} - \sum_{i=1}^{k-1} \hat{A}_{k,k-1} \bar{B}_{k-1,i} (u_{if} - v_{if}^{(j_f)}) \right.$$

$$- B_{kk}(u_{kf} - u_{kf}^{(j_f)})\Bigg) + \frac{\partial V_{kf}^{(j_f+1)}}{\partial s_k}\Bigg] \mathrm{d}\tau'_{kf}. \tag{47}$$

Then, one obtains

$$\begin{aligned} &V_{kf}^{(j_f+1)}\left(z_{kf}, \frac{t+T}{\varepsilon_k}\right) - V_{kf}^{(j_f+1)}\left(z_{kf}, \frac{t}{\varepsilon_k}\right) \\ &\quad = -\int_{\frac{t}{\varepsilon_k}}^{\frac{t+T}{\varepsilon_k}} r_{kf}^{(j_f)} \mathrm{d}\tau'_{kf} - \int_{\frac{t}{\varepsilon_k}}^{\frac{t+T}{\varepsilon_k}} \left(\frac{\partial V_{kf}^{(j_f+1)}}{\partial z_{kf}}\right)^{\mathrm{T}} B_{kk}(u_{kf} - u_{kf}^{(j_f)}) \mathrm{d}\tau'_{kf} \\ &\qquad - \int_{\frac{t}{\varepsilon_k}}^{\frac{t+T}{\varepsilon_k}} \left(\frac{\partial V_{kf}^{(j_f+1)}}{\partial z_{kf}}\right)^{\mathrm{T}} \sum_{i=1}^{k-1} \hat{A}_{k,k-1} \bar{B}_{k-1,i}(u_{if} - v_{if}^{(j_f)}) \mathrm{d}\tau'_{kf}, \\ &V_{kf}^{(j_f+1)}(z_{kf}, s_{ktf}) = \Psi_{kf}(z_{kf}(s_{ktf})). \end{aligned} \tag{48}$$

Here, solving the Bellman equations (48) requires knowledge of system matrices. We will use the minmax strategy of multiplayer games [30, 31] to eliminate this requirement. The virtual control inputs v_{if}^* satisfy the below equation:

$$\frac{\partial V_{kf}^{*\,\mathrm{T}}}{\partial z_{kf}} \hat{A}_{k,k-1} \bar{B}_{k-1,i} = 2\gamma_k^2 v_{if}^{*\,\mathrm{T}} \hat{R}_{if}. \tag{49}$$

Thus, a model-free off-policy IRL Algorithm 1 is created by substituting (49) into the Bellman equations (48).

Algorithm 1 Model-free off-policy IRL algorithm for Problem 2

1: Start with arbitrary stabilizing behavior control policies u_{kf} and u_{if} which are used to collect data, and select initial admissible control policies $u_{kf}^{(0)}$ and $v_{if}^{(0)}(i = 1, 2, \ldots, k-1)$, where the iteration index $j_f = 0$;

2: Solve the Bellman equations for $(V_{kf}^{(j_f+1)}, u_{kf}^{(j_f+1)}, v_{if}^{(j_f+1)})$:

$$\begin{aligned} &V_{kf}^{(j_f+1)}\left(z_{kf}, \frac{t+T}{\varepsilon_k}\right) - V_{kf}^{(j_f+1)}\left(z_{kf}, \frac{t}{\varepsilon_k}\right) \\ &\quad = -\int_{\frac{t}{\varepsilon_k}}^{\frac{t+T}{\varepsilon_k}} \bar{r}_{kf}^{(j_f)} \mathrm{d}\tau'_{kf} + 2\int_{\frac{t}{\varepsilon_k}}^{\frac{t+T}{\varepsilon_k}} \left(u_{kf}^{(j_f+1)}\right)^{\mathrm{T}} \hat{R}_{kf}\left(u_{kf} - u_{kf}^{(j_f)}\right) \mathrm{d}\tau'_{kf} \\ &\qquad - 2\gamma_k^2 \sum_{i=1}^{k-1} \int_{\frac{t}{\varepsilon_k}}^{\frac{t+T}{\varepsilon_k}} \left(v_{if}^{(j_f+1)}\right)^{\mathrm{T}} \hat{R}_{if}\left(u_{if} - v_{if}^{(j_f)}\right) \mathrm{d}\tau'_{kf}, \\ &V_{kf}^{(j_f+1)}(z_{kf}, s_{ktf}) = \Psi_{kf}(z_{kf}(s_{ktf})), \end{aligned} \tag{50}$$

where $\bar{r}_{kf}^{(j_f)} = z_{kf}^{\mathrm{T}} \hat{Q}_{kf} z_{kf} + (u_{kf}^{(j_f)})^{\mathrm{T}} \hat{R}_{kf} u_{kf}^{(j_f)} - \gamma_k^2 \sum_{i=1}^{k-1} (v_{if}^{(j_f)})^{\mathrm{T}} \hat{R}_{if} v_{if}^{(j_f)}$;

3: Stop when $\| V_{kf}^{(j_f+1)} - V_{kf}^{(j_f)} \| \leqslant \varepsilon_f$ for all k with a small constant ε_f $(\varepsilon_f > 0)$; otherwise set $j_f = j_f + 1$ and go back to step 2.

The convergence of Algorithm 1 can be demonstrated in similar ways to [29, 31]. In Algorithm 1, the virtual control policies v_{if} $(i = 1, 2, \ldots, k-1)$ are not the actual policies used by agent i. They only serve to assist agent k in learning its control policy. Furthermore, the truth is that while learned control policies $\{u_{1f}, u_{2f}, \ldots, u_{Nf}\}$ cannot force agents to reach the Nash equilibrium, all agents can prepare themselves for the worst-case behavior of their individual neighbors when each agent strives to minimize its cost [30, 31]. The highlight of Algorithm 1 is that no model parameters of systems (1) and (2) are required, resulting in a completely data-driven off-policy RL approach. However, the data of z_{kf} is not available, which makes it difficult to implement Algorithm 1. Section 4 will present a solution to this problem.

For Problem 3, we will find $\{u_{1s}, u_{2s}, \ldots, u_{Ns}\}$ for minimizing (28) subject to (27) using the off-policy IRL method. The value function of the slow subsystem is defined as follows:

$$V_s(\bar{Y}, t) = \int_t^{tf} \left[\bar{Y}^{\mathrm{T}} Q_s \bar{Y} + \sum_{i=1}^{N} u_{is}^{\mathrm{T}} R_{is} u_{is}\right] \mathrm{d}\tau + \bar{\Psi}_s(\bar{Y}(tf)). \tag{51}$$

Comparably, an infinitesimal that corresponds to (51) is presented as follows:

$$\frac{\partial V_s}{\partial t}+\frac{\partial V_s^{\mathrm{T}}}{\partial \bar{Y}}\left(A\bar{Y}+\sum_{i=1}^{N}B_{is}u_{is}\right)+r_s=0, \tag{52}$$

where $r_s=\bar{Y}^{\mathrm{T}}Q_s\bar{Y}+\sum_{i=1}^{N}u_{is}^{\mathrm{T}}R_{is}u_{is}$. Then the Hamiltonian is

$$\begin{aligned}H_s=&r_s+\frac{\partial V_s^{\mathrm{T}}}{\partial \bar{Y}}\frac{\mathrm{d}\bar{Y}}{\mathrm{d}t}+\frac{\partial V_s}{\partial t}\\ =&r_s+\frac{\partial V_s^{\mathrm{T}}}{\partial \bar{Y}}\left(A\bar{Y}+\sum_{i=1}^{N}B_{is}u_{is}\right)+\frac{\partial V_s}{\partial t}.\end{aligned} \tag{53}$$

Implementing $\frac{\partial H_s}{\partial u_{is}}=0$ yields

$$u_{is}^{*}=-\frac{1}{2}R_{is}^{-1}B_{is}^{\mathrm{T}}\frac{\partial V_s^{*}}{\partial \bar{Y}}. \tag{54}$$

Substituting (54) into (52) yields the HJB equation as follows:

$$\begin{aligned}&\left(\frac{\partial V_s^{*}}{\partial \bar{Y}}\right)^{\mathrm{T}}A\bar{Y}+\bar{Y}^{\mathrm{T}}Q_s\bar{Y}-\frac{1}{4}\left(\frac{\partial V_s^{*}}{\partial \bar{Y}}\right)^{\mathrm{T}}B_{is}R_{is}^{-1}B_{is}^{\mathrm{T}}\frac{\partial V_s^{*}}{\partial \bar{Y}}+\frac{\partial V_s^{*}}{\partial t}=0,\\ &V_s(\bar{Y},tf)=\bar{\Psi}_s(\bar{Y}(tf)).\end{aligned} \tag{55}$$

Similarly, the HJB equation (55) is a PDE. To address (55), the classical Algorithm 2 can be developed by (52) and (54). However, it requires slow subsystem (27) information, which is currently unknown.

Algorithm 2 Model-based PI RL algorithm for Problem 3

1: Choose initial admissible control policies $u_{is}^{(0)}$ $(i=1,\ldots,k,\ldots,N)$, $j_s=0$;
2: Solve the following Bellman equation for $V_s^{(j_s+1)}$:

$$\begin{aligned}&\frac{\partial V_s^{(j_s+1)}}{\partial t}+\left(\frac{\partial V_s^{(j_s+1)}}{\partial \bar{Y}}\right)^{\mathrm{T}}\left(A\bar{Y}+\sum_{i=1}^{N}B_{is}u_{is}^{(j_s)}\right)+r_s^{(j_s)}=0,\\ &V_s^{(j_s+1)}(\bar{Y},tf)=\bar{\Psi}_s(\bar{Y}(tf)),\end{aligned} \tag{56}$$

where $r_s^{(j_s)}=\bar{Y}^{\mathrm{T}}Q_s\bar{Y}+\sum_{i=1}^{N}(u_{is}^{(j_s)})^{\mathrm{T}}R_{is}u_{is}^{(j_s)}$;
3: Update the control policies as

$$u_{is}^{(j_s+1)}=-\frac{1}{2}R_{is}^{-1}B_{is}^{\mathrm{T}}\frac{\partial V_s^{(j_s+1)}}{\partial \bar{Y}}; \tag{57}$$

4: Stop when $\| V_s^{(j_s+1)}-V_s^{(j_s)} \|\leqslant \varepsilon_f$ for all k with a small constant ε_s $(\varepsilon_s>0)$; otherwise set $j_s=j_s+1$ and go back to step 2.

To derive a model-free RL algorithm, we present the auxiliary variables $u_{is}^{(j_s)}$ into the reduced-order slow subsystem (27). Thus, one has

$$\dot{\bar{Y}}=A\bar{Y}+\sum_{i=1}^{N}B_{is}u_{is}^{(j_s)}+\sum_{i=1}^{N}B_{is}\left(u_{is}-u_{is}^{(j_s)}\right). \tag{58}$$

Differentiating $V_s^{(j_s+1)}$ in combination with the augmented dynamics of the reduced-order slow subsystem (27) and considering (52), one obtains

$$\begin{aligned}\dot{V}_s^{(j_s+1)}=&\left(\frac{\partial V_s^{(j_s+1)}}{\partial \bar{Y}}\right)^{\mathrm{T}}\frac{\mathrm{d}\bar{Y}}{\mathrm{d}t}+\frac{\partial V_s^{(j_s+1)}}{\partial t}\\ =&\left(\frac{\partial V_s^{(j_s+1)}}{\partial \bar{Y}}\right)^{\mathrm{T}}\left(A\bar{Y}+\sum_{i=1}^{N}B_{is}u_{is}^{(j_s)}\right)+\frac{\partial V_s^{(j_s+1)}}{\partial t}\\ =&-r_s^{(j_s)}.\end{aligned} \tag{59}$$

Furthermore, integrating both sides of (59) from t to $t+T$ yields the following equality based on (58):

$$\begin{aligned}\int_t^{t+T}\dot{V}_s^{(j_s+1)}\mathrm{d}\tau&=\int_t^{t+T}\left[\left(\frac{\partial V_s^{(j_s+1)}}{\partial\bar{Y}}\right)^{\mathrm{T}}\left(A\bar{Y}+\sum_{i=1}^{N}B_{is}u_{is}^{(j_s)}\right)+\frac{\partial V_s^{(j_s+1)}}{\partial t}\right]\mathrm{d}\tau\\&=\int_t^{t+T}\left[\left(\frac{\partial V_s^{(j_s+1)}}{\partial\bar{Y}}\right)^{\mathrm{T}}\left(\frac{\mathrm{d}\bar{Y}}{\mathrm{d}t}-\sum_{i=1}^{N}B_{is}(u_{is}-u_{is}^{(j_s)})\right)+\frac{\partial V_s^{(j_s+1)}}{\partial t}\right]\mathrm{d}\tau.\end{aligned}\tag{60}$$

Therefore, one has

$$\begin{cases}V_s^{(j_s+1)}(\bar{Y}(t+T),t+T)-V_s^{(j_s+1)}(\bar{Y}(t),t)=-\displaystyle\int_t^{t+T}r_s^{(j_s)}\mathrm{d}\tau\\\qquad\qquad\qquad\qquad\qquad\qquad-\displaystyle\int_t^{t+T}\left[\left(\frac{\partial V_s^{(j_s+1)}}{\partial\bar{Y}}\right)^{\mathrm{T}}\sum_{i=1}^{N}B_{is}(u_{is}-u_{is}^{(j_s)})\right]\mathrm{d}\tau,\\V_s^{(j_s+1)}(\bar{Y},tf)=\bar{\Psi}_s(\bar{Y}(tf)).\end{cases}\tag{61}$$

Substituting (57) into the second term of the right-hand sides of the Bellman equation (61) yields

$$-\int_t^{t+T}\left[\left(\frac{\partial V_s^{(j_s+1)}}{\partial\bar{Y}}\right)^{\mathrm{T}}\sum_{i=1}^{N}B_{is}(u_{is}-u_{is}^{(j_s)})\right]\mathrm{d}\tau=2\int_t^{t+T}\sum_{i=1}^{N}(u_{is}^{(j_s+1)})^{\mathrm{T}}R_{is}(u_{is}-u_{is}^{(j_s)})\mathrm{d}\tau.\tag{62}$$

Algorithm 3 is designed to solve Problem 3 by solving the Bellman equation (63).

Algorithm 3 Model-free off-policy IRL algorithm for Problem 3

1: Start with arbitrary stabilizing behavior control policies u_{is} to collect data, and select initial control policies $u_{is}^{(0)}$, where the iteration index $j_s=0$;
2: Solve the Bellman equation for $(V_s^{(j_s+1)},u_{is}^{(j_s+1)})$:

$$\begin{aligned}V_s^{(j_s+1)}(\bar{Y}(t+T),t+T)-V_s^{(j_s+1)}(\bar{Y}(t),t)=&-\int_t^{t+T}r_s^{(j_s)}\mathrm{d}\tau\\&+2\int_t^{t+T}\sum_{i=1}^{N}(u_{is}^{(j_s+1)})^{\mathrm{T}}R_{is}(u_{is}-u_{is}^{(j_s)})\mathrm{d}\tau\end{aligned}\tag{63}$$

with $V_s^{(j_s+1)}(\bar{Y},tf)=\bar{\Psi}_s(\bar{Y}(tf))$;
3: Stop when $\|V_s^{(j_s+1)}-V_s^{(j_s)}\|\leqslant\varepsilon_s$ with a small constant ε_s ($\varepsilon_s>0$); otherwise set $j_s=j_s+1$ and go back to step 2.

Remark 5. $u_{is}^{(j_s)}$ in Algorithm 3 can converge to u_{is}^* that drives the output of the slow subsystem to the reference trajectory r^*, and it can be easily proven like [26]. Thus, the composite controllers $u_k=u_{kf}^{(j_f)}+u_{ks}^{(j_s)}$ converge to $u_{kc}^*=u_{kf}^*+u_{ks}^*$ when $j_f\to\infty$ and $j_s\to\infty$. Additionally, one can find that the control policies $u_{if}^{(j_f)}$ and $u_{is}^{(j_s)}$ are distributed in the sense that they rely upon their individual value functions during the learning.

4 Data-driven RL for multitime scales

In this section, we focus on developing data-driven algorithms to learn the optimal control policies by combining Algorithms 1 and 3.

4.1 Data-driven IRL algorithm design

It should be noted that z_{kf} in (22) should be replaced by other variables because they cannot be directly measured from the global system (1).

Given (17), (29), and $x_k=\varepsilon_k z_k$, the unmeasurable z_{kf} can be approximated by $\hat{z}_{kf}$ as follows:

$$\hat{z}_{kf}=z_k-z_{ks}=M_{kf}\eta_{kf},\tag{64}$$

where $M_{kf} = [\frac{1}{\varepsilon_k} I\ \tilde{B}_{k-1}\ \hat{A}_{kk}^{-1} B_{kk}]$, $\tilde{B}_{k-1} = [\hat{A}_{kk}^{-1}\hat{A}_{k,k-1}\bar{B}_{k-1,1}\ \hat{A}_{kk}^{-1}\hat{A}_{k,k-1}\bar{B}_{k-1,2}\ \cdots\ \hat{A}_{kk}^{-1}\hat{A}_{k,k-1}\bar{B}_{k-1,k-1}]$, $\eta_{kf} = [x_k^{\mathrm{T}}\ \tilde{u}_{k-1,s}^{\mathrm{T}}\ u_{ks}^{\mathrm{T}}]^{\mathrm{T}}$, and $\tilde{u}_{k-1,s} = [u_{1s}^{\mathrm{T}}\ u_{2s}^{\mathrm{T}}\ \cdots\ u_{k-1,s}^{\mathrm{T}}]^{\mathrm{T}}$.

For the kth fast subsystem, the off-policy Bellman equations (50) in Algorithm 1 can be rewritten as follows:

$$\begin{aligned} &V_{kf}^{(j_f+1)}(z_{kf}, t+T) - V_{kf}^{(j_f+1)}(z_{kf}, t) \\ &\quad = 2\int_t^{t+T} (u_{kf}^{(j_f+1)})^{\mathrm{T}} R_{kf}(u_{kf} - u_{kf}^{(j_f)})\mathrm{d}\tau_- \int_t^{t+T} \bar{r}_{kf}^{'(j_f)}\mathrm{d}\tau \\ &\qquad - 2\gamma_k^2 \sum_{i=1}^{k-1}\int_t^{t+T} (v_{if}^{(j_f+1)})^{\mathrm{T}} R_{if}(u_{if} - v_{if}^{(j_f)})\mathrm{d}\tau, \\ &V_{kf}^{(j_f+1)}(z_{kf}, \ldots, z_{1f}, tf) = \Psi_{kf}(\eta_{kf}(tf)), \end{aligned} \tag{65}$$

where $\bar{r}_{kf}^{'(j_f)} = \dfrac{1}{\varepsilon_k}\eta_{kf}^{\mathrm{T}} M_{kf}^{\mathrm{T}}\hat{Q}_{kf} M_{kf}\eta_{kf} + (u_{kf}^{(j_f)})^{\mathrm{T}} R_{kf} u_{kf}^{(j_f)} - \gamma_k^2 \sum_{i=1}^{k-1}(v_{if}^{(j_f)})^{\mathrm{T}} R_{if} v_{if}^{(j_f)}$.

Remark 6. Following variable substitution, one discovers that $\bar{r}_{kf}^{'(j_f)}$ in (65) requires prior knowledge of the system model of (1). Taking (17) and (29) into consideration, the term $[\tilde{B}_{k-1}\ \hat{A}_{kk}^{-1}B_{kk}]$ can be approximated using $\dfrac{1}{\varepsilon_k}x_k = -\hat{A}_{kk}^{-1}(\hat{A}_{k,k-1}\sum_{i=1}^{k-1}\bar{B}_{k-1,i}u_{is} + B_{kk}u_{ks})$. Thus, M_{kf} can be estimated as follows:

$$M_{kf} := \frac{1}{\varepsilon_k}[I\ -x_k[\tilde{u}_{k-1,s}^{\mathrm{T}}\ u_{ks}^{\mathrm{T}}]([\tilde{u}_{k-1,s}^{\mathrm{T}}\ u_{ks}^{\mathrm{T}}]^{\mathrm{T}}[\tilde{u}_{k-1,s}^{\mathrm{T}}\ u_{ks}^{\mathrm{T}}])^{-1}]. \tag{66}$$

This estimation method appears to be feasible if and only if the fast behavior strategies u_{kf} are sufficiently small, and $\omega_k = ([\tilde{u}_{k-1,s}^{\mathrm{T}}\ u_{ks}^{\mathrm{T}}]^{\mathrm{T}}[\tilde{u}_{k-1,s}^{\mathrm{T}}\ u_{ks}^{\mathrm{T}}])$ are invertible while the global system reaches steady state [19]. As such, $\bar{r}_{kf}^{'(j_f)}$ are estimated as follows:

$$\bar{r}_{kf}^{'(j_f)} := \eta_{kf}^{\mathrm{T}}\xi^{\mathrm{T}} Q_{kf}\xi\eta_{kf} + (u_{kf}^{(j_f)})^{\mathrm{T}} R_{kf} u_{kf}^{(j_f)} - \gamma_k^2\sum_{i=1}^{k-1}(v_{if}^{(j_f)})^{\mathrm{T}} R_{if} v_{if}^{(j_f)}, \tag{67}$$

where $\xi = [I\ -x_k[\tilde{u}_{k-1,s}^{\mathrm{T}}\ u_{ks}^{\mathrm{T}}]\omega_k^{-1}]$. As shown in (65) and (67), it is not necessary to know the system model (1) or the time-scale parameters ε_k $(k = 1, 2, \ldots, N)$ when calculating $V_{kf}^{(j_f+1)}$. Furthermore, y is actually y_s in the slow subsystem. Thus, $\hat{Y} = [y^{\mathrm{T}}\ r^{*\mathrm{T}}]^{\mathrm{T}}$ is obtained. As a result, the off-policy Bellman equation (63) of Algorithm 3 can be rewritten as follows:

$$\begin{cases} V_s^{(j_s+1)}(\hat{Y}, t+T) - V_s^{(j_s+1)}(\hat{Y}, t) = -\displaystyle\int_t^{t+T} r_s^{'(j_s)}\mathrm{d}\tau \\ \qquad\qquad + 2\displaystyle\sum_{i=1}^{N}\int_t^{t+T}(u_{is}^{(j_s+1)})^{\mathrm{T}} R_{is}(u_{is} - u_{is}^{(j_s)})\mathrm{d}\tau, \\ V_s^{(j_s+1)}(\hat{Y}, tf) = \bar{\Psi}_s(\hat{Y}(tf)), \end{cases} \tag{68}$$

where $r_s^{'(j_s)} = \hat{Y}^{\mathrm{T}} Q_s \hat{Y} + \sum_{i=1}^{N}(u_{is}^{(j_s)})^{\mathrm{T}} R_{is} u_{is}^{(j_s)}$.

Algorithm 4 is designed to find the approximately optimal composite control policies u_{kc}, such that the solution to Problem 1, the primary research objective, can be found without requiring the knowledge of the global system (1) and (2).

4.2 NN-based approximation

Now, we will use the actor-critic structure with the NN estimation to find the composite controllers u_{kc}^* using only data.

For Problem 2, when z_{kf} is approximated by (64), the value functions and the control policies can be represented by [29]

$$V_{kf}(\eta_{kf}, t) = W_{ckf}^{\mathrm{T}}\varphi_{ckf}(\eta_{kf}, t) + \epsilon_{ckf}(\eta_{kf}, t), \tag{69}$$

Algorithm 4 Data-driven off-policy IRL algorithm for Problem 1

1: Choose an admissible behavior control policy $u(t)$ to collect data, give initial admissible control policies $u_{kf}^{(0)}$ and $u_{ks}^{(0)}$, and set the iteration indices with $j_f = 0$ and $j_s = 0$;
Fast subsystems learning:
2: Solve the HJB equations (65) for $(V_{kf}^{(j_f+1)}, u_{kf}^{(j_f+1)})$ using collected data;
3: Make $j_f = j_f + 1$ and go back to step 2; stop when $\| V_{kf}^{(j_f+1)} - V_{kf}^{(j_f)} \| \leqslant \varepsilon_f$ with a small constant $\varepsilon_f > 0$;
Slow subsystems learning:
4: Solve the HJB equations (68) for $(V_s^{(j_s+1)}, u_{ks}^{(j_s+1)})$ using collected data;
5: Make $j_s = j_s + 1$ and go back to step 4; stop when $\| V_s^{(j_s+1)} - V_s^{(j_s)} \| \leqslant \varepsilon_s$ with a small constant $\varepsilon_s > 0$;
6: Compute the composite control inputs $u_{kc} = u_{kf}^{(j_f+1)} + u_{ks}^{(j_s+1)}$ as the best control inputs.

$$u_{kf}(\eta_{kf}, t) = W_{akf}^{\mathrm{T}} \phi_{akf}(\eta_{kf}, t) + \epsilon_{akf}(\eta_{kf}, t), \tag{70}$$

$$v_{if}(\eta_{kf}, t) = W_{aif}^{\mathrm{T}} \phi_{aif}(\eta_{kf}, t) + \epsilon_{aif}(\eta_{kf}, t), \tag{71}$$

where $W_{ckf} \in \mathbb{R}^{l_1}$, $W_{akf} \in \mathbb{R}^{m \times l_2}$, and $W_{aif} \in \mathbb{R}^{m \times l_2}$ are the appropriate weight for the kth fast critic NN, the kth fast actor NN, and the virtual actor NN, respectively. l_1 and l_2 are the numbers of hidden-layer neurons, $\varphi_{ckf} \in \mathbb{R}^{l_1}$, $\phi_{akf} \in \mathbb{R}^{l_2}$, and $\phi_{aif} \in \mathbb{R}^{l_2}$ represent the time-varying activation functions for the kth fast critic NN, the kth fast actor NN, and the virtual actor NN, respectively. ϵ_{ckf}, ϵ_{akf}, and ϵ_{aif} are NN reconstruction errors.

Let the estimations of the kth fast value functions V_{kf}, the kth fast control inputs u_{kf}, and the virtual control inputs v_{if} be

$$\hat{V}_{kf}(\eta_{kf}, t) = \hat{W}_{ckf}^{\mathrm{T}} \varphi_{ckf}(\eta_{kf}, t), \tag{72}$$

$$\hat{u}_{kf}(\eta_{kf}, t) = \hat{W}_{akf}^{\mathrm{T}} \phi_{akf}(\eta_{kf}, t), \tag{73}$$

$$\hat{v}_{if}(\eta_{kf}, t) = \hat{W}_{aif}^{\mathrm{T}} \phi_{aif}(\eta_{kf}, t), \tag{74}$$

where $\hat{W}_{ckf}$, $\hat{W}_{akf}$, and $\hat{W}_{aif}$ are the estimated values of the suitable weights W_{ckf}, W_{akf}, and W_{aif}.

The terminal estimations of the fast value functions become

$$\hat{V}_{kf}(\eta_{kf}, tf) = \hat{W}_{ckf}^{\mathrm{T}} \varphi_{ckf}(\eta_{kf}, tf). \tag{75}$$

Putting the estimations of V_{kf}, u_{kf}, and v_{if} described as (72)–(74) into (65) in Algorithm 4 produces the fast residual errors as follows:

$$\begin{aligned} e_{rkf} =& \hat{W}_{ckf}^{\mathrm{T}} (\varphi_{ckf}(\eta_{kf}, t) - \varphi_{ckf}(\eta_{kf}, t+T)) \\ & - 2\gamma_k^2 \sum_{i=1}^{k-1} \int_t^{t+T} (\hat{W}_{aif}^{\mathrm{T}} \phi_{aif})^{\mathrm{T}} R_{if} (u_{if} - \hat{v}_{if}^{(j_f)}) \mathrm{d}\tau \\ & + 2 \int_t^{t+T} (\hat{W}_{akf}^{\mathrm{T}} \phi_{akf})^{\mathrm{T}} R_{kf} (u_{kf} - \hat{u}_{kf}^{(j_f)}) \mathrm{d}\tau - \int_t^{t+T} \bar{r}_{kf}^{'(j_f)} \mathrm{d}\tau. \end{aligned} \tag{76}$$

Note that

$$V_{kf}(\eta_{kf}, tf) = \Psi_{kf}(\eta_{kf}(tf)). \tag{77}$$

For the finite-horizon control issue, both the time-varying property of the fast value functions and the terminal condition must be considered. Therefore, it is reasonable to design NN weight update laws that minimize both the residual errors (76) and the following terminal term's errors:

$$e_{ckf} = \hat{W}_{ckf}^{\mathrm{T}} \varphi_{ckf}(\eta_{kf}, tf) - \Psi_{kf}(\eta_{kf}(tf)). \tag{78}$$

Then, the fast residual errors e_{rkf} are rewritten as follows:

$$e_{rkf} = -M_{1kf} \hat{W}_{ckf} + \sum_{i=1}^{k-1} M_{2if} \mathrm{vec}(\hat{W}_{aif}) + M_{2kf} \mathrm{vec}(\hat{W}_{akf}) - N_{kf}, \tag{79}$$

where

$$M_{1kf} = \Delta \varphi_{ckf}^{\mathrm{T}} \otimes I,$$

$$\begin{aligned}
&\Delta\varphi_{ckf} = \varphi_{ckf}(\eta_{kf}, t+T) - \varphi_{ckf}(\eta_{kf}, t),\\
&M_{2if} = -2\gamma_k^2 \int_t^{t+T} ((u_{if} - \hat{v}_{if}^{(j_f)})^{\mathrm{T}} R_{if} \otimes \phi_{aif}^{\mathrm{T}}) \mathrm{d}\tau,\\
&M_{2kf} = 2 \int_t^{t+T} ((u_{kf} - \hat{u}_{kf}^{(j_f)})^{\mathrm{T}} R_{kf} \otimes \phi_{akf}^{\mathrm{T}}) \mathrm{d}\tau,\\
&N_{kf} = \int_t^{t+T} \bar{r}_{kf}^{'(j_f)} \mathrm{d}\tau.
\end{aligned} \tag{80}$$

The NN weights are tuned by the gradient descent algorithm [32],

$$\begin{aligned}
&\dot{\hat{W}}_{ckf} = -\alpha_{1f}(e_{rkf} \cdot -M_{1kf}^{\mathrm{T}} + e_{ckf} \cdot \varphi_{ckf}(\eta_{kf}, tf)),\\
&\mathrm{vec}(\dot{\hat{W}}_{aif}) = -\alpha_{2f}(e_{rkf} \cdot M_{2if}^{\mathrm{T}}),\\
&\mathrm{vec}(\dot{\hat{W}}_{akf}) = -\alpha_{3f}(e_{rkf} \cdot M_{2kf}^{\mathrm{T}}),
\end{aligned} \tag{81}$$

where α_{1f}, α_{2f}, and α_{3f} are the learning rate.

For Problem 3, the slow value functions and the slow control policies can be defined as follows:

$$V_s(\hat{Y}(t), t) = W_{cs}^{\mathrm{T}} \varphi_{cs}(\hat{Y}(t), t) + \epsilon_{cs}(\hat{Y}(t), t), \tag{82}$$

$$u_{is}(\hat{Y}(t), t) = W_{ais}^{\mathrm{T}} \phi_{ais}(\hat{Y}(t), t) + \epsilon_{ais}(\hat{Y}(t), t), \tag{83}$$

where $W_{cs} \in \mathbb{R}^{l_3}$ and $W_{ais} \in \mathbb{R}^{n \times l_4}$ are the optimum weights for the slow critic NN and the slow actor NN, respectively. l_3 and l_4 are the numbers of hidden-layer neurons, and $\varphi_{cs} \in \mathbb{R}^{l_3}$ and $\phi_{ais} \in \mathbb{R}^{l_4}$ are the time-varying activation functions for the slow critic NN and the slow actor NN, respectively. ϵ_{cs} and ϵ_{ais} are NN reconstruction errors.

Let the estimation of the function V_s and u_{is} be

$$\hat{V}_s(\hat{Y}(t), t) = \hat{W}_{cs}^{\mathrm{T}} \varphi_{cs}(\hat{Y}(t), t), \tag{84}$$

$$\hat{u}_{is}(\hat{Y}(t), t) = \hat{W}_{ais}^{\mathrm{T}} \phi_{ais}(\hat{Y}(t), t), \tag{85}$$

where $\hat{W}_{cs}$ and $\hat{W}_{ais}$ are the estimated values of the optimum weights W_{cs} and W_{ais}, respectively.

The terminal estimation of the value function changes to

$$\hat{V}_s(\hat{Y}(tf), tf) = \hat{W}_{cs}^{\mathrm{T}} \varphi_{cs}(\hat{Y}(tf), tf). \tag{86}$$

Inserting (85) into the second term of the right-hand sides of (68) produces

$$\begin{aligned}
\sum_{i=1}^{N} \hat{u}_{is}^{\mathrm{T}}(t) R_{is}(u_{is}(t) - \hat{u}_{is}^{(j_s)}(t)) &= \sum_{i=1}^{N} (\hat{W}_{ais}^{\mathrm{T}}(t) \phi_{ais})^{\mathrm{T}} R_{is}(u_{is}(t) - \hat{u}_{is}^{(j_s)}(t))\\
&= \sum_{i=1}^{N} (((u_{is}(t) - \hat{u}_{is}^{(j_s)}(t))^{\mathrm{T}} R_{is}) \otimes \phi_{ais}^{\mathrm{T}}) \mathrm{vec}(\hat{W}_{ais}).
\end{aligned} \tag{87}$$

Let $\Delta\varphi_{cs} = \varphi_{cs}(\hat{Y}(t+T), t+T) - \varphi_{cs}(\hat{Y}(t), t)$; then the left-hand sides of (68) is defined as follows:

$$\hat{V}_s(\hat{Y}, t+T) - \hat{V}_s(\hat{Y}, t) = \hat{W}_{cs}^{\mathrm{T}} \Delta\varphi_{cs} = (\Delta\varphi_{cs}^{\mathrm{T}} \otimes I) \hat{W}_{cs}. \tag{88}$$

Substituting (84) and (85) into the Bellman equation (68) generates the slow residual errors as follows:

$$\begin{aligned}
e_{ris} =& 2 \sum_{i=1}^{N} \left(\int_t^{t+T} ((u_{is} - \hat{u}_{is}^{(j_s)})^{\mathrm{T}} R_{is}) \otimes \phi_{ais}^{\mathrm{T}} \mathrm{d}\tau \mathrm{vec}(\hat{W}_{ais}) \right)\\
&- \int_t^{t+T} r_s^{'(j_s)} \mathrm{d}\tau - (\Delta\varphi_{cs}^{\mathrm{T}} \otimes I) \hat{W}_{cs}.
\end{aligned} \tag{89}$$

Note that

$$V_s(\hat{Y}(tf), tf) = \bar{\Psi}_s(\hat{Y}(tf)). \tag{90}$$

One has the terminal term's error as follows:

$$e_{cs} = \hat{W}_{cs}^{\mathrm{T}} \varphi_{cs}(\hat{Y}(tf), tf) - \bar{\Psi}_s(\hat{Y}(tf)). \tag{91}$$

Then, the slow residual errors e_{ris} are rewritten as follows:

$$e_{ris} = -M_{1s}\hat{W}_{cs} + \sum_{i=1}^{N} M_{2is}\mathrm{vec}(\hat{W}_{ais}) - N_s, \tag{92}$$

where

$$\begin{aligned} M_{1s} &= \Delta\varphi_{cs}^{\mathrm{T}} \otimes I, \\ M_{2is} &= 2\int_t^{t+T} ((u_{is}(\tau) - \hat{u}_{is}^{(j_s)}(\tau))^{\mathrm{T}} R_{is}) \otimes \phi_{ais}^{\mathrm{T}} \mathrm{d}\tau, \\ N_s &= \int_t^{t+T} r_s^{\prime(j_s)} \mathrm{d}\tau. \end{aligned} \tag{93}$$

According to the gradient descent method, the NN weights are tuned below:

$$\begin{aligned} \dot{\hat{W}}_{cs} &= -\alpha_{1s}(e_{ris} \cdot -M_{1s}^{\mathrm{T}} + e_{cs} \cdot \varphi_{cs}(\hat{Y}(tf), tf)), \\ \mathrm{vec}(\dot{\hat{W}}_{ais}) &= -\alpha_{2s}(e_{ris} \cdot M_{2is}^{\mathrm{T}}), \end{aligned} \tag{94}$$

where α_{1s} and α_{2s} are the learning rate.

Algorithm 4 will be implemented using the actor-critic NN framework, and Eqs. (81) and (94) will be used in order to find the solutions of (65) and (68) in Steps 2 and 4, such that the composite control inputs u_{kc} can be learned. Figure 2 depicts the detailed SP-based RL scheme for designing the composite control inputs.

Similar to [29, 33], the following general assumption is presented.

Assumption 2. Assume that the initial NN weights, learning rate, and number of hidden-layer neurons are chosen properly, which implies that $\epsilon_{ckf}, \epsilon_{akf}, \epsilon_{aif}, \epsilon_{cs}$, and ϵ_{ais} can be bounded.

Theorem 2. The NN weights estimation errors $\tilde{W}_{ckf}^{(j_f)} = \hat{W}_{ckf}^{(j_f)} - W_{ckf}$, $\tilde{W}_{akf}^{(j_f)} = \hat{W}_{akf}^{(j_f)} - W_{akf}$, $\tilde{W}_{aif}^{(j_f)} = \hat{W}_{aif}^{(j_f)} - W_{aif}$, $\tilde{W}_{cs}^{(j_s)} = \hat{W}_{cs}^{(j_s)} - W_{cs}$, and $\tilde{W}_{ais}^{(j_s)} = \hat{W}_{ais}^{(j_s)} - W_{ais}$ can be bounded; i.e., the NN weight estimation errors are uniformly ultimately bounded (UUB).

Proof. (a) For the estimation error of NN weight update in (81), consider the Lyapunov candidate function, which is defined as follows:

$$\begin{aligned} L_f(t) \equiv& \frac{1}{2}\mathrm{tr}(\tilde{W}_{ckf}^{\mathrm{T}}\alpha_{1f}^{-1}\tilde{W}_{ckf}) + \frac{1}{2}\sum_{i=1}^{k-1}\mathrm{tr}(\mathrm{vec}(\tilde{W}_{aif})^{\mathrm{T}}\alpha_{2f}^{-1}\mathrm{vec}(\tilde{W}_{aif})) \\ &+ \frac{1}{2}\mathrm{tr}(\mathrm{vec}(\tilde{W}_{akf})^{\mathrm{T}}\alpha_{3f}^{-1}\mathrm{vec}(\tilde{W}_{akf})). \end{aligned} \tag{95}$$

Let $L_{1f}(t) = \frac{1}{2}\mathrm{tr}(\tilde{W}_{ckf}^{\mathrm{T}}\alpha_{1f}^{-1}\tilde{W}_{ckf}), L_{2f}(t) = \frac{1}{2}\sum_{i=1}^{k-1}\mathrm{tr}(\mathrm{vec}(\tilde{W}_{aif})^{\mathrm{T}}\alpha_{2f}^{-1}\mathrm{vec}(\tilde{W}_{aif})), L_{3f}(t) = \frac{1}{2}\mathrm{tr}(\mathrm{vec}(\tilde{W}_{akf})^{\mathrm{T}} \cdot \alpha_{3f}^{-1}\mathrm{vec}(\tilde{W}_{akf}))$. It follows

$$\dot{L}_f \equiv \dot{L}_{1f}(t) + \dot{L}_{2f}(t) + \dot{L}_{3f}(t). \tag{96}$$

The first term of (96) is

$$\begin{aligned} \dot{L}_{1f} =& \tilde{W}_{ckf}^{\mathrm{T}}\alpha_{1f}^{-1}\dot{\tilde{W}}_{ckf} \\ =& -\tilde{W}_{ckf}^{\mathrm{T}}M_{1kf}^{\mathrm{T}}\Delta\varphi_{ckf}^{\mathrm{T}}\tilde{W}_{ckf} + \tilde{W}_{ckf}^{\mathrm{T}}M_{1kf}M_{2kf}\mathrm{vec}(\tilde{W}_{akf}) \end{aligned}$$

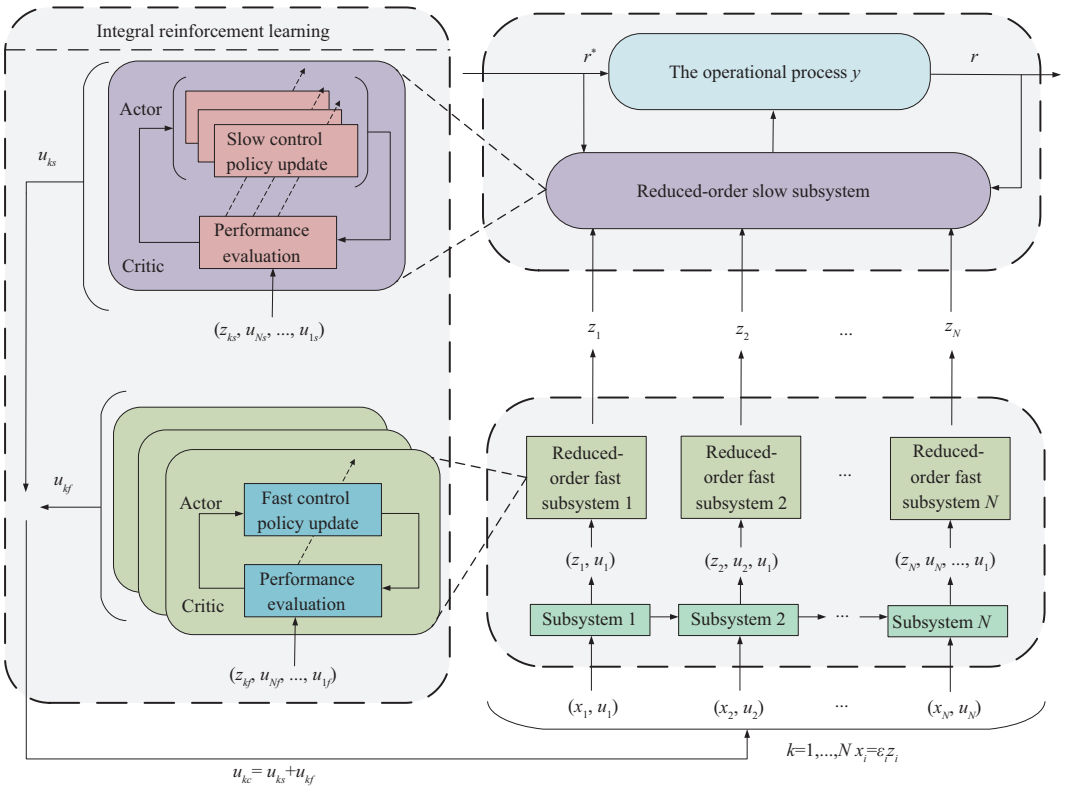

Figure 2 (Color online) Singular perturbation-based RL scheme for designing the composite control inputs.

$$+\tilde{W}_{ckf}^{\mathrm{T}}\sum_{i=1}^{k-1}M_{1kf}M_{2if}\mathrm{vec}(\tilde{W}_{aif})+\tilde{W}_{ckf}^{\mathrm{T}}\xi_{kf}M_{1kf}^{\mathrm{T}}$$
$$-\tilde{W}_{ckf}^{\mathrm{T}}\varphi_{ckf}(\eta_{kf},tf)\varphi_{ckf}(\eta_{kf},tf)^{\mathrm{T}}\tilde{W}_{ckf}-\tilde{W}_{ckf}^{\mathrm{T}}\epsilon_{ckf}\varphi_{ckf}(\eta_{kf},tf), \tag{97}$$

where $\xi_{kf}=\epsilon_{ckf}(\eta_{kf},t)-\epsilon_{ckf}(\eta_{kf},t+T)+2\int_t^{t+T}\epsilon_{akf}^{\mathrm{T}}R_{kf}(u_{kf}-\hat{u}_{kf}^{(j_f)})\mathrm{d}\tau-2\gamma_k^2\sum_{i=1}^{k-1}\int_t^{t+T}\epsilon_{aif}^{\mathrm{T}}R_{if}(\upsilon_{if}-\hat{\upsilon}_{if}^{(j_f)})\mathrm{d}\tau$.

The second term of (96) is

$$\begin{aligned}\dot{L}_{2f}=&\sum_{i=1}^{k-1}\mathrm{vec}(\tilde{W}_{aif})^{\mathrm{T}}\alpha_{2f}^{-1}\mathrm{vec}(\dot{\tilde{W}}_{aif})\\=&\sum_{i=1}^{k-1}\mathrm{vec}(\tilde{W}_{aif})^{\mathrm{T}}M_{2if}^{\mathrm{T}}\Delta\varphi_{ckf}^{\mathrm{T}}\tilde{W}_{ckf}-\sum_{i=1}^{k-1}\mathrm{vec}(\tilde{W}_{aif})^{\mathrm{T}}M_{2if}M_{2kf}\mathrm{vec}(\tilde{W}_{akf})\\&-\sum_{i=1}^{k-1}\mathrm{vec}(\tilde{W}_{aif})^{\mathrm{T}}M_{2if}^{\mathrm{T}}\sum_{i=1}^{k-1}M_{2if}\mathrm{vec}(\tilde{W}_{aif})-\sum_{i=1}^{k-1}\mathrm{vec}(\tilde{W}_{aif})^{\mathrm{T}}\xi_{kf}M_{2if}^{\mathrm{T}}.\end{aligned} \tag{98}$$

The third term of (96) is

$$\begin{aligned}\dot{L}_{3f}=&\mathrm{vec}(\tilde{W}_{akf})^{\mathrm{T}}\alpha_{3f}^{-1}\mathrm{vec}(\dot{\tilde{W}}_{akf})\\=&\mathrm{vec}(\tilde{W}_{akf})^{\mathrm{T}}M_{2kf}^{\mathrm{T}}\Delta\varphi_{ckf}^{\mathrm{T}}\tilde{W}_{ckf}-\mathrm{vec}(\tilde{W}_{akf})^{\mathrm{T}}M_{2kf}^{\mathrm{T}}M_{2kf}\mathrm{vec}(\tilde{W}_{akf})\\&-\mathrm{vec}(\tilde{W}_{akf})^{\mathrm{T}}\sum_{i=1}^{k-1}M_{2kf}M_{2if}\mathrm{vec}(\tilde{W}_{aif})-\mathrm{vec}(\tilde{W}_{akf})^{\mathrm{T}}\xi_{kf}M_{2kf}^{\mathrm{T}}.\end{aligned} \tag{99}$$

Letting $\bar{Z}_f=[\tilde{W}_{ckf},\mathrm{vec}(\tilde{W}_{a1f}),\ldots,\mathrm{vec}(\tilde{W}_{akf})]^{\mathrm{T}}$, Eq. (96) becomes

$$\dot{L}_f=-\bar{Z}_f^{\mathrm{T}}M_f\bar{Z}_f+\bar{Z}_f^{\mathrm{T}}N_f. \tag{100}$$

Define

$$M_f = \begin{bmatrix} \Delta\varphi_{ckf}\Delta\varphi_{ckf}^{\mathrm{T}} + \varphi_{ckf}\varphi_{ckf}^{\mathrm{T}} & -\Delta\varphi_{ckf}^{\mathrm{T}}M_{21f} & \cdots & -\Delta\varphi_{ckf}^{\mathrm{T}}M_{2k-1f} & -\Delta\varphi_{ckf}^{\mathrm{T}}M_{2kf} \\ -M_{21f}^{\mathrm{T}}\Delta\varphi_{ckf} & M_{21f}^{\mathrm{T}}M_{21f} & \cdots & M_{21f}^{\mathrm{T}}M_{2,k-1,f} & M_{21f}^{\mathrm{T}}M_{2kf} \\ \vdots & \vdots & \cdots & \vdots & \vdots \\ -M_{2k-1f}^{\mathrm{T}}\Delta\varphi_{ckf} & M_{2k-1f}^{\mathrm{T}}M_{21f} & \cdots & M_{2k-1f}^{\mathrm{T}}M_{2k-1f} & M_{2k-1f}^{\mathrm{T}}M_{2kf} \\ -M_{2kf}^{\mathrm{T}}\Delta\varphi_{ckf} & M_{2kf}^{\mathrm{T}}M_{21f} & \cdots & M_{2kf}^{\mathrm{T}}M_{2,k-1,f} & M_{2kf}^{\mathrm{T}}M_{2kf} \end{bmatrix} \tag{101}$$

and

$$N_f = \begin{bmatrix} \xi_{kf}\Delta\varphi_{ckf}^{\mathrm{T}} - \epsilon_{ckf}(\eta_{kf}, tf)\varphi_{ckf}^{\mathrm{T}}(\eta_{kf}, tf) \\ -\xi_{kf}M_{21f}^{\mathrm{T}} \\ \vdots \\ -\xi_{kf}M_{2k-1f}^{\mathrm{T}} \\ -\xi_{kf}M_{2kf}^{\mathrm{T}} \end{bmatrix}.$$

One has

$$\dot{L}_f < -\parallel \bar{Z}_f \parallel^2 \lambda_{\min}(M_f) + \parallel N_f \parallel \parallel \bar{Z}_f \parallel, \tag{102}$$

where M_f is a positive definite matrix by calculating all leading principal minors of it. Completing the squares yields that the Lyapunov derivative is negative if

$$\parallel \bar{Z}_f \parallel > \frac{\parallel N_f \parallel}{\lambda_{\min}(M_f)}. \tag{103}$$

As a result, we concluded that $\dot{L}_f$ is negative outside the compact residual set,

$$\Omega_f = \left\{ \parallel \bar{Z}_f \parallel : \bar{Z}_f \leqslant \frac{\parallel N_f \parallel}{\lambda_{\min}(M_f)} \right\}. \tag{104}$$

(b) For the estimation error of NN weight update in (94), consider the Lyapunov candidate function defined as follows:

$$L_s(t) \equiv \frac{1}{2}\mathrm{tr}(\tilde{W}_{cs}^{\mathrm{T}}\alpha_{1s}^{-1}\tilde{W}_{cs}) + \frac{1}{2}\sum_{i=1}^{N}\mathrm{tr}(\mathrm{vec}(\tilde{W}_{ais})^{\mathrm{T}}\alpha_{2s}^{-1}\mathrm{vec}(\tilde{W}_{ais})). \tag{105}$$

Letting $L_{1s}(t) = \frac{1}{2}\mathrm{tr}(\tilde{W}_{cs}^{\mathrm{T}}\alpha_{1s}^{-1}\tilde{W}_{cs}), L_{2s}(t) = \frac{1}{2}\sum_{i=1}^{N}\mathrm{tr}(\mathrm{vec}(\tilde{W}_{ais})^{\mathrm{T}}\alpha_{2s}^{-1}\mathrm{vec}(\tilde{W}_{ais}))$, the derivative of L_s is derived as follows:

$$\dot{L}_s \equiv \dot{L}_{1s}(t) + \dot{L}_{2s}(t). \tag{106}$$

Then, it follows

$$\begin{aligned} \dot{L}_{1s} =& \tilde{W}_{cs}^{\mathrm{T}}\alpha_{1s}^{-1}\dot{\tilde{W}}_{cs} \\ =& -\tilde{W}_{cs}^{\mathrm{T}}M_{1s}^{\mathrm{T}}\Delta\varphi_{cs}^{\mathrm{T}}\tilde{W}_{cs} + \tilde{W}_{cs}^{\mathrm{T}}\sum_{i=1}^{N}M_{1s}M_{2is}\mathrm{vec}(\tilde{W}_{ais}) + \tilde{W}_{cs}^{\mathrm{T}}\xi_{ks}M_{1s}^{\mathrm{T}} \\ & -\tilde{W}_{cs}^{\mathrm{T}}\varphi_{cs}(\hat{Y}(tf), tf)\varphi_{cs}(\hat{Y}(tf), tf)^{\mathrm{T}}\tilde{W}_{cs} - \tilde{W}_{cs}^{\mathrm{T}}\epsilon_{cs}(\hat{Y}(tf), tf)\varphi_{cs}(\hat{Y}(tf), tf), \end{aligned} \tag{107}$$

where $\xi_{ks} = \epsilon_{cs}(\hat{Y}(t), t) - \epsilon_{cs}(\hat{Y}(t+T), t+T) + 2\sum_{i=1}^{N}\int_{t}^{t+T}\epsilon_{ais}^{\mathrm{T}}R_{is}(u_{is} - \hat{u}_{is}^{(j_s)})\mathrm{d}\tau$.

The second term of (106) is

$$\begin{aligned} \dot{L}_{2s} =& \sum_{i=1}^{N}\mathrm{vec}(\tilde{W}_{ais})^{\mathrm{T}}\alpha_{2s}^{-1}\mathrm{vec}(\dot{\tilde{W}}_{ais}) \\ =& \sum_{i=1}^{N}\mathrm{vec}(\tilde{W}_{ais})^{\mathrm{T}}M_{2is}^{\mathrm{T}}\Delta\varphi_{cs}^{\mathrm{T}}\tilde{W}_{cs} - \sum_{i=1}^{N}\mathrm{vec}(\tilde{W}_{ais})^{\mathrm{T}}M_{2is}^{\mathrm{T}}\sum_{i=1}^{N}M_{2is}\mathrm{vec}(\tilde{W}_{ais}) \end{aligned}$$

$$-\sum_{i=1}^{N}\mathrm{vec}(\tilde{W}_{ais})^{\mathrm{T}}\xi_{ks}M_{2is}^{\mathrm{T}}. \tag{108}$$

Letting $\bar{Z}_s=[\tilde{W}_{cs},\mathrm{vec}(\tilde{W}_{a1s}),\ldots,\mathrm{vec}(\tilde{W}_{aNs})]^{\mathrm{T}}$, Eq. (106) becomes

$$\dot{L}_s=-\bar{Z}_s^{\mathrm{T}}M_s\bar{Z}_s+\bar{Z}_s^{\mathrm{T}}N_s. \tag{109}$$

Define

$$M_s=\begin{bmatrix}\Delta\varphi_{cs}\Delta\varphi_{cs}^{\mathrm{T}}+\varphi_{cs}(\hat{Y}(tf),tf)\varphi_{cs}^{\mathrm{T}}(\hat{Y}(tf),tf) & -\Delta\varphi_{cs}^{\mathrm{T}}M_{21s} & \cdots & -\Delta\varphi_{cs}^{\mathrm{T}}M_{2Ns}\\ -M_{21s}^{\mathrm{T}}\Delta\varphi_{cs} & M_{21s}^{\mathrm{T}}M_{21s} & \cdots & M_{21s}^{\mathrm{T}}M_{2Ns}\\ \vdots & \vdots & \cdots & \vdots\\ -M_{2Ns}^{\mathrm{T}}\Delta\varphi_{cs} & M_{2Ns}^{\mathrm{T}}M_{21s} & \cdots & M_{2Ns}^{\mathrm{T}}M_{2Ns}\end{bmatrix} \tag{110}$$

and

$$N_s=\begin{bmatrix}\xi_{ks}\Delta\varphi_{cs}^{\mathrm{T}}-\epsilon_{cs}(\hat{Y}(tf),tf)\varphi_{cs}^{\mathrm{T}}(\hat{Y}(tf),tf)\\ -\xi_{ks}M_{21s}^{\mathrm{T}}\\ \vdots\\ -\xi_{ks}M_{2Ns}^{\mathrm{T}}\end{bmatrix}.$$

Then, one has

$$\dot{L}_s<-\parallel\bar{Z}_s\parallel^2\lambda_{\min}(M_s)+\parallel N_s\parallel\parallel\bar{Z}_s\parallel, \tag{111}$$

where M_s is a positive definite matrix obtained by calculating all of its leading principal minors. Completing the squares yields that the Lyapunov derivative is negative if

$$\parallel\bar{Z}_s\parallel>\frac{\parallel N_s\parallel}{\lambda_{\min}(M_s)}. \tag{112}$$

As a result, we concluded that $\dot{L}_s$ is negative outside the compact residual set,

$$\Omega_s=\left\{\parallel\bar{Z}_s\parallel:\bar{Z}_s\leqslant\frac{\parallel N_s\parallel}{\lambda_{\min}(M_s)}\right\}. \tag{113}$$

Therefore, the estimation errors of NN weights are UUB. The proof is finished.

Remark 7. In [21,22], the developed RL method for finding the composite controller for tracking problems of large-scale systems is not a fully model-free approach, since the model parameters of fast unit processes must be known. Furthermore, only one unit device process is in [18–20], and the coupling relationship between fast unit processes is ignored when designing OTCs in [21,22]. However, the developed SP-based RL algorithm (Algorithm 4) can find the composite controller that drives the output $r(t)$ of the slow operational process to the reference trajectory r^* via an approximately optimal approach. Using Algorithm 4 combined and the NN-based approximation, only the measured data, including the state of the system (1) and the iterative control inputs $u_{kf}^{(j_f+1)}$ and $u_{ks}^{(j_s+1)}$ as well as the behavior control inputs $u(t)$, are required to learn the approximate optimal tracking composite control policies. Moreover, another advantage of this paper is that the time-scale parameters ε_k do not need to be known. In this sense, the developed method is indeed a completely data-driven approach that does not require the information of model parameters of the entire system even though it runs with more than multitime scales.

5 Simulation results

5.1 Numerical example of two subsystems

A numerical example comprising multiple subsystems is presented to show the efficiency of the proposed method. The fast dynamics of the two subsystems at the bottom are given by

$$\dot{x}_1(t)=\begin{bmatrix}-500 & 400\\ -300 & -200\end{bmatrix}x_1(t)+\begin{bmatrix}2\\ -2\end{bmatrix}u_1(t), \tag{114}$$

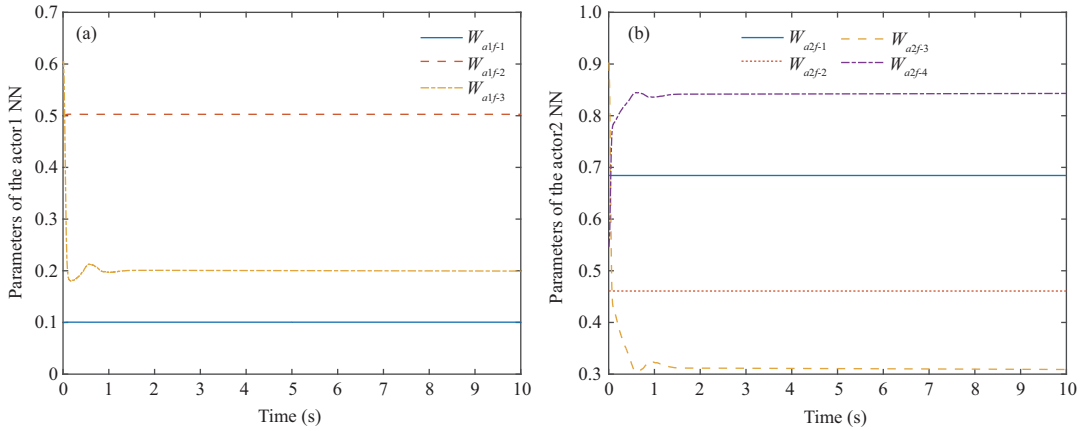

Figure 3 (Color online) Fast actor NN weights for (a) subsystem 1 and (b) subsystem 2.

$$\dot{x}_2(t) = \begin{bmatrix} -300 & -100 \\ 0 & -200 \end{bmatrix} x_2(t) + \begin{bmatrix} -100 & 100 \\ 0 & -100 \end{bmatrix} x_1(t) + \begin{bmatrix} -100 \\ 100 \end{bmatrix} u_2(t). \tag{115}$$

The slow dynamics of the top operational process is as follows:

$$\dot{y}(t) = \begin{bmatrix} -2 & 0 \\ 0 & -3 \end{bmatrix} y(t) + \begin{bmatrix} 100 & -100 \\ -100 & -300 \end{bmatrix} x_1(t) + \begin{bmatrix} -300 & 200 \\ -400 & -200 \end{bmatrix} x_2(t), \tag{116}$$

$$r(t) = \begin{bmatrix} 1 & 0 \\ 0 & 0.5 \end{bmatrix} y(t). \tag{117}$$

Let $r^* = [6, 2]^{\mathrm{T}}$, $Q_{1f} = Q_{2f} = 0.01I$, $Q_0 = 10I$, $R_{1f} = R_{2f} = 0.1$, and $R_{1s} = R_{2s} = 0.1, tf = 10$. The time-varying activation functions for the fast critic NN and the slow critic NN are respectively presented as follows:

$$\varphi_{c1f}(\eta_{1f}, t) = [\eta_{1f_1}^2(1 - 0.1\mathrm{e}^{-\tau})/(1 + 0.1\mathrm{e}^{-\tau}), \eta_{1f_1}\eta_{1f_2}, \eta_{1f_1}\eta_{1f_3}, \eta_{1f_2}^2, \eta_{1f_2}\eta_{1f_3}, \eta_{1f_3}^2], \tag{118}$$

$$\begin{aligned}\varphi_{c2f}(\eta_{2f}, \eta_{1f}, t) =& [\eta_{2f_1}^2(1 - 0.1\mathrm{e}^{-\tau})/(1 + 0.1\mathrm{e}^{-\tau}), \eta_{2f_1}\eta_{2f_2}, \eta_{2f_1}\eta_{2f_3}, \eta_{2f_1}\eta_{2f_4}, \eta_{2f_2}^2, \\ & \eta_{2f_2}\eta_{2f_3}, \eta_{2f_2}\eta_{2f_4}, \eta_{2f_3}^2, \eta_{2f_3}\eta_{2f_4}, \eta_{2f_4}^2],\end{aligned} \tag{119}$$

$$\begin{aligned}\varphi_{cs}(\hat{Y}(t), t) =& [\hat{Y}_1^2(1 - 0.1\mathrm{e}^{-\tau})/(1 + 0.1\mathrm{e}^{-\tau}), \hat{Y}_1\hat{Y}_2, \hat{Y}_1\hat{Y}_3, \hat{Y}_1\hat{Y}_4, \hat{Y}_2^2, \\ & \hat{Y}_2\hat{Y}_3, \hat{Y}_2\hat{Y}_4, \hat{Y}_3^2, \hat{Y}_3\hat{Y}_4, \hat{Y}_4^2],\end{aligned} \tag{120}$$

where $\tau = tf - t$. The time-varying activation functions for the fast actor NN and the slow actor NN are respectively expressed as follows:

$$\varphi_{a1f}(\eta_{1f}, t) = [\eta_{1f_1}(1 - 0.1\mathrm{e}^{-\tau})/(1 + 0.1\mathrm{e}^{-\tau}), \eta_{1f_2}, \eta_{1f_3}], \tag{121a}$$

$$\varphi_{a2f}(\eta_{2f}, \eta_{1f}, t) = [\eta_{2f_1}(1 - 0.1\mathrm{e}^{-\tau})/(1 + 0.1\mathrm{e}^{-\tau}), \eta_{2f_2}, \eta_{2f_3}, \eta_{2f_4}], \tag{121b}$$

$$\varphi_{a1s}(\hat{Y}(t), t) = \varphi_{a2s}(\hat{Y}(t), t) = [\hat{Y}_1(1 - 0.1\mathrm{e}^{-\tau})/(1 + 0.1\mathrm{e}^{-\tau}), \hat{Y}_2, \hat{Y}_3, \hat{Y}_4]. \tag{121c}$$

Set the initial values $x_1(0) = [-1, -3]^{\mathrm{T}}$, $x_2(0) = [-4, 3]^{\mathrm{T}}$, $y(0) = [50, -50]^{\mathrm{T}}$. The initial values of the NN weights of the slow and fast subsystems are presented below:

$$\begin{aligned} W_{a1f} =& [0.3968, 0.4754, 0.7447]^{\mathrm{T}}, & W_{a2f} =& [0.9072, 0.1979, 0.6001, 0.2465]^{\mathrm{T}}, \\ W_{a1s} =& [0.4873, 0.5344, 0.4144, 0.3438]^{\mathrm{T}}, & W_{a2s} =& [0.0858, 0.2648, 0.3992, 0.3628]^{\mathrm{T}}. \end{aligned} \tag{122}$$

Figures 3–6 show the simulation results after implementing Algorithm 4 and the NN approximation. Figures 3(a) and (b) depict the evolution of the actor NN weights W_{a1f} and W_{a2f} of the fast processes.

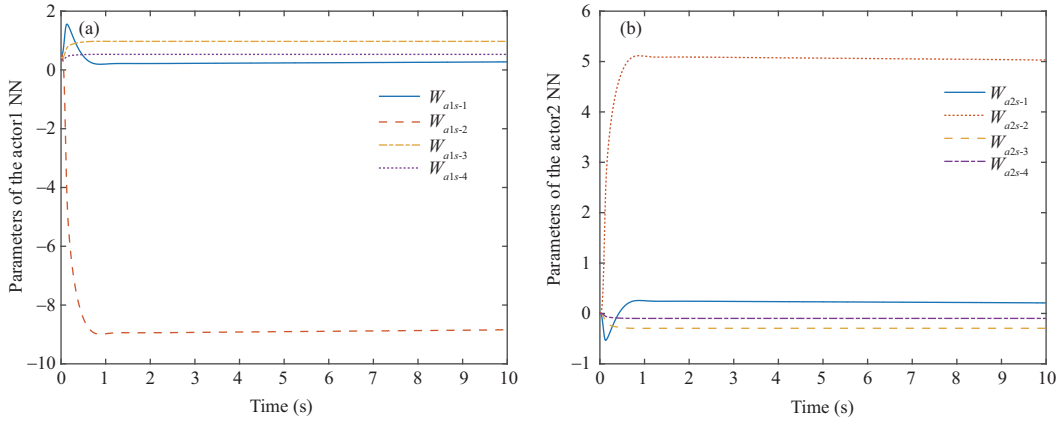

Figure 4 (Color online) Slow actor NN weights for (a) subsystem 1 and (b) subsystem 2.

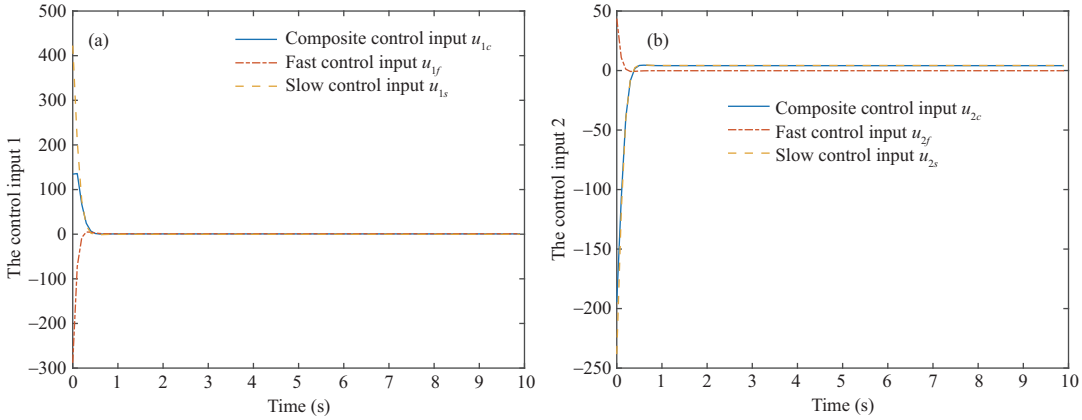

Figure 5 (Color online) Trajectories of (a) the composite control policy u_{1c} and its slow u_{1s} and fast u_{1f} components and (b) the composite control policy u_{2c} and its slow u_{2s} and fast u_{2f} components.

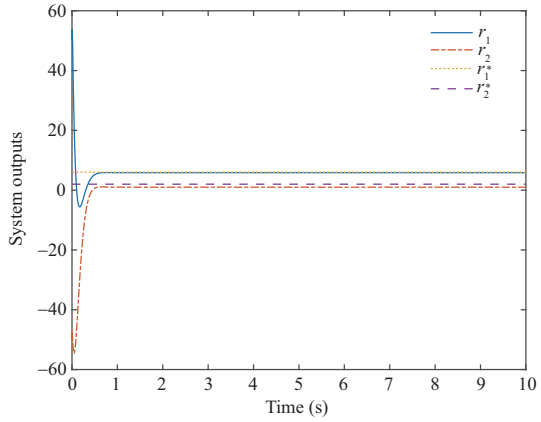

Figure 6 (Color online) System outputs and reference trajectories.

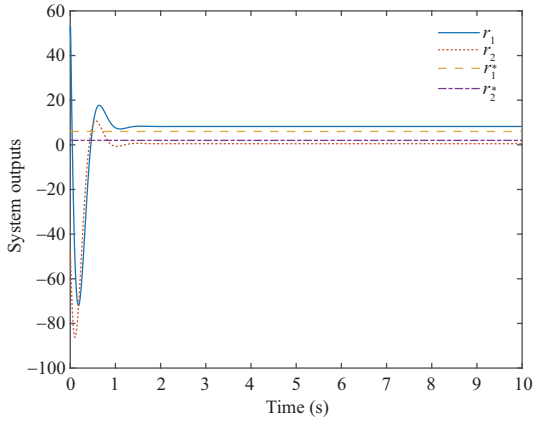

Figure 7 (Color online) System outputs and reference trajectories without coupling consideration.

Figures 4(a) and (b) depict the evolution of the actor NN weights W_{a1s} and W_{a2s} of dynamics of the operational indicators. Figures 5(a) and (b) show the trajectories of composite controllers $u_{1c}(t) = u_{1f}(t) + u_{1s}(t)$, $u_{2c}(t) = u_{2f}(t) + u_{2s}(t)$, and Figure 6 shows the tracking results of operation indices. Figure 6 shows that the operational indices r_1, r_2 successfully track the desired operational indices r_1^*, r_2^*. The simulation results show that the proposed approach in Algorithm 4 is capable of achieving satisfactory tracking performance of the system with multitime scales and bringing the system to a steady state in a completely model-free manner.

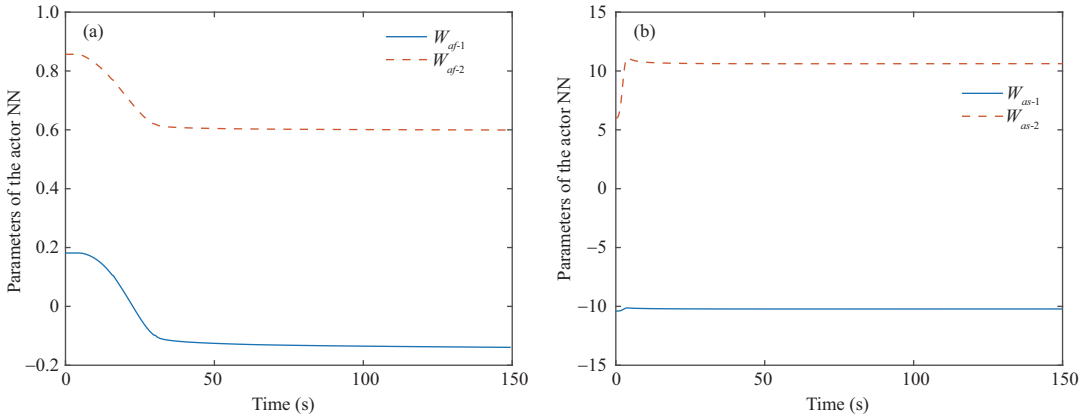

Figure 8 (Color online) (a) Fast and (b) slow actor NN weights for the underflow slurry pump frequency.

Assume that this numerical example does account for global system coupling. We suppose that

$$\dot{x}_1(t) = \begin{bmatrix} -500 & 400 \\ -300 & -200 \end{bmatrix} x_1(t) + \begin{bmatrix} 2 \\ -2 \end{bmatrix} u_1(t), \tag{123}$$

$$\dot{x}_2(t) = \begin{bmatrix} -300 & -100 \\ 0 & -200 \end{bmatrix} x_2(t) + \begin{bmatrix} -1 \\ 1 \end{bmatrix} u_2(t). \tag{124}$$

For comparison, the method in [22] is used to find the composite controllers for the system composed by (123)–(124). Then, these controllers actually act the system (114)–(117) with a coupling relationship between $x_1(t)$ and $x_2(t)$. As a result, the tracking results are shown in Figure 7. When Figures 6 and 7 are compared, it is clear that the tracking results obtained using the developed Algorithm 4 in this paper outperform those obtained without taking into account subsystem coupling.

5.2 A practical system example

5.2.1 *Application to the mixed separation thickening process*

The mixed separation thickening process (MSTP) is a fast and slow scale industrial system that operates near its operating point. Consider the linear dynamics of MSTP

$$\begin{cases} \dot{y}(t) = -0.68y(t) + 2.6u(t), \\ \dot{r}(t) = -0.057r(t) + 0.055y(t), \end{cases} \tag{125}$$

where $y(t)$ represents the underflow slurry flow rate, $r(t)$ represents the underflow concentration, and $u(t)$ represents the underflow slurry pump frequency. Furthermore, $y(t)$ operates on a fast time scale, whereas $r(t)$ operates on a slow time scale.

To achieve the tracking control goal, we choose the desired underflow concentration value as $r^* = 33$. Set the values of the parameters $Q_f = 120$, $Q_0 = 105$, $R_f = R_s = 1$, and $\tau = tf - t$. The initial NN weights of the fast and slow subsystems are chosen as $W_{af}(0) = [0.753, 0.134]^{\mathrm{T}}$ and $W_{as}(0) = [-10.4, 6]^{\mathrm{T}}$, respectively. Next, Algorithm 4 with the NN approximation is utilized to learn the NN weights in the fast and slow subsystems. Figures 8(a) and (b) depict the evolution of the NN weights in the fast and slow subsystems, respectively. Figure 9 depicts the learned composite control strategy $u_c(t) = u_f(t) + u_s(t)$. Figure 10 demonstrates that the desired operational index r^* can be obtained using the learned $u_c(t) = u_f(t) + u_s(t)$ in Figure 9.

5.2.2 *Comparison*

Using the method [34] for Problem 1, in which system (125) is not decomposed by the SP technique, Figure 11 plots the tracking results. When Figure 11 is compared with Figure 10, it is clear that the composite controller learned by the developed method in this paper produces a faster tracking result.

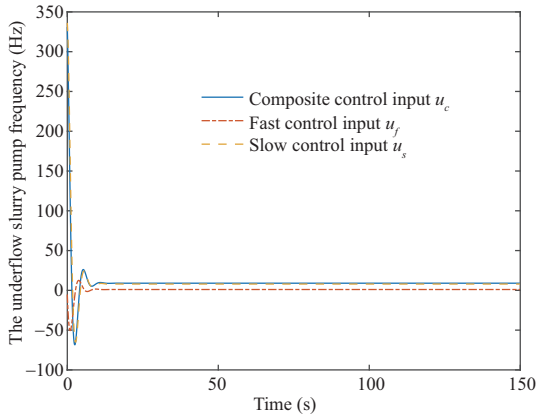

Figure 9 (Color online) Trajectories of the underflow slurry pump frequency u_c and its slow u_s and fast u_f components.

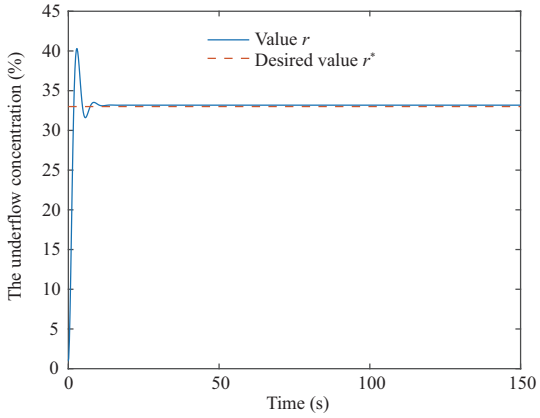

Figure 10 (Color online) Underflow concentration r tracks the desired value r^*.

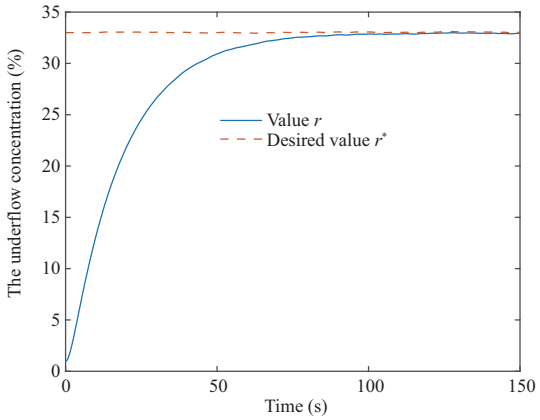

Figure 11 (Color online) Concentration value r tracks the desired value r^* using the method [34].

Table 1 Data comparison of simulation results where $k^* = 600$, $n = 200$

	IAE	MSE	Convergence time (s)
Algorithm 4	12.14	0.004	13.865
The method [34]	19.86	0.009	108.2

To specifically evaluate the control performance, the integral absolute error (IAE) and the mean square error (MSE) [35] are used below:

$$\text{IAE} = \sum_{i=k^*}^{k^*+n} |r(i) - r^*(i)|, \qquad \text{MSE} = \sqrt{\frac{1}{n}\sum_{i=k^*}^{k^*+n} |r(i) - r^*(i)|^2}. \tag{126}$$

Table 1 shows the comparative data. According to Table 1, the developed method in this paper produces satisfactory IAE and MSE, implying that the developed method in this paper outperforms the method without fast and slow mode decomposition, since ill-condition could be caused by the different time scales without using the SP decomposition.

6 Conclusion

In this paper, we present a novel data-driven RL-based method combined with SP theory for achieving the OTC of large-scale systems with multitime scales using only measured data. A global optimization problem is decomposed into the reduced-order fast game and reduced-order slow LQT subproblems, such that the sum of separate subproblem performances is roughly equal to the global performance, which is

rigorously proven. The actor-critic NN structure is established, resulting in the development of a distributed off-policy IRL algorithm to learn the optimal control protocols for reaching global optimization. Finally, simulation results are used to demonstrate the efficacy of the proposed algorithm using an MSTP and a numerical example.

Acknowledgements This work was supported by National Natural Science Foundation of China (Grant Nos. 62073158, 61991404, 61991400, 61673280), Science and Technology Major Project 2020 of Liaoning Province (Grant No. 2020JH1/10100008), Open Project of Key Field Alliance of Liaoning Province (Grant No. 2019KF0306), and Basic Research Project of Education Department of Liaoning Province (Grant No. LJKZ0401).

References

1 Xie S, Huang J, Zhao C, et al. Application of neural network to hierarchical optimal control of the class of continuous time-varying large-scale systems. In: Proceedings of the IEEE International Conference on Intelligent Processing Systems, Beijing, 1997. 477–481
2 Bakule L. Decentralized control: an overview. Annu Rev Control, 2008, 32: 87–98
3 Chai T, Qin S J, Wang H. Optimal operational control for complex industrial processes. Annu Rev Control, 2014, 38: 81–92
4 Chai T, Ding J, Wu F. Hybrid intelligent control for optimal operation of shaft furnace roasting process. Control Eng Pract, 2011, 19: 264–275
5 Yuan Y, Wang Z, Guo L. Distributed quantized multi-modal H_∞ fusion filtering for two-time-scale systems. Inf Sci, 2018, 432: 572–583
6 Chen W H, Liu Y, Zheng W X. Synchronization analysis of two-time-scale nonlinear complex networks with time-scale-dependent coupling. IEEE Trans Cybern, 2018, 49: 3255–3267
7 Jiang Y, Fan J, Chai T, et al. Dual-rate operational optimal control for flotation industrial process with unknown operational model. IEEE Trans Ind Electron, 2018, 66: 4587–4599
8 Chow J, Kokotovic P. A decomposition of near-optimum regulators for systems with slow and fast modes. IEEE Trans Automat Contr, 1976, 21: 701–705
9 Khalil H. Output feedback control of linear two-time-scale systems. IEEE Trans Automat Contr, 1987, 32: 784–792
10 Kokotović P, Khalil H, O'Reilly J. Singular Perturbation Methods in Control: Analysis and Design. Philadelphia: Society for Industrial and Applied Mathematics, 1999
11 Cavallo A, de Maria G, Nistri P. Robust control design with integral action and limited rate control. IEEE Trans Automat Contr, 1999, 44: 1569–1572
12 Bouyekhf R, Hami A E, Moudni A E. Optimal control of a particular class of singularly perturbed nonlinear discrete-time systems. IEEE Trans Automat Contr, 2001, 46: 1097–1101
13 Litkouhi B, Khalil H. Multirate and composite control of two-time-scale discrete-time systems. IEEE Trans Automat Contr, 1985, 30: 645–651
14 Kodra K, Gajic Z. Optimal control for a new class of singularly perturbed linear systems. Automatica, 2017, 81: 203–208
15 Ding J, Modares H, Chai T, et al. Data-based multiobjective plant-wide performance optimization of industrial processes under dynamic environments. IEEE Trans Ind Inf, 2016, 12: 454–465
16 Vrabie D, Pastravanu O, Abu-Khalaf M, et al. Adaptive optimal control for continuous-time linear systems based on policy iteration. Automatica, 2009, 45: 477–484
17 Jiang Y, Jiang Z P. Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics. Automatica, 2012, 48: 2699–2704
18 Xue W, Fan J, Lopez V G, et al. New methods for optimal operational control of industrial processes using reinforcement learning on two time scales. IEEE Trans Ind Inf, 2019, 16: 3085–3099
19 Xue W, Fan J, Lopez V G, et al. Off-policy reinforcement learning for tracking in continuous-time systems on two time scales. IEEE Trans Neural Netw Learn Syst, 2020, 32: 4334–4346
20 Li J, Kiumarsi B, Chai T, et al. Off-policy reinforcement learning: optimal operational control for two-time-scale industrial processes. IEEE Trans Cybern, 2017, 47: 4547–4558
21 Zhou L, Zhao J, Ma L, et al. Decentralized composite suboptimal control for a class of two-time-scale interconnected networks with unknown slow dynamics. Neurocomputing, 2020, 382: 71–79
22 Zhao J, Yang C, Dai W, et al. Reinforcement learning-based composite optimal operational control of industrial systems with multiple unit devices. IEEE Trans Ind Inf, 2021, 18: 1091–1101
23 Zhang L, Wang S, Wu Q, et al. Were mercury emission factors for Chinese non-ferrous metal smelters overestimated? Evidence from onsite measurements in six smelters. Environ Pollution, 2012, 171: 109–117
24 Li J, Ding J, Chai T, et al. Nonzero-sum game reinforcement learning for performance optimization in large-scale industrial processes. IEEE Trans Cybern, 2019, 50: 4132–4145
25 Saksena V, Cruz J J. Nash strategies in decentralized control of multiparameter singularly perturbed large scale systems. Large Scale Syst, 1981, 2: 219–234
26 Modares H, Lewis F L. Linear quadratic tracking control of partially-unknown continuous-time systems using reinforcement learning. IEEE Trans Automat Contr, 2014, 59: 3051–3056
27 Chen C, Xie L, Jiang Y, et al. Robust output regulation and reinforcement learning-based output tracking design for unknown linear discrete-time systems. IEEE Trans Automat Contr, 2023, 68: 2391–2398
28 Kiumarsi B, Lewis F L, Modares H, et al. Reinforcement Q-learning for optimal tracking control of linear discrete-time systems with unknown dynamics. Automatica, 2014, 50: 1167–1175
29 Zhang H, Cui X, Luo Y, et al. Finite-horizon H_∞ tracking control for unknown nonlinear systems with saturating actuators. IEEE Trans Neural Netw Learn Syst, 2017, 29: 1200–1212
30 Lopez V G, Lewis F L, Wan Y, et al. Stability and robustness analysis of minmax solutions for differential graphical games. Automatica, 2020, 121: 109177
31 Liu M, Wan Y, Lopez V G, et al. Differential graphical game with distributed global Nash solution. IEEE Trans Control Netw Syst, 2021, 8: 1371–1382
32 Wang D, Hu L, Zhao M, et al. Dual event-triggered constrained control through adaptive critic for discrete-time zero-sum games. IEEE Trans Syst Man Cybern Syst, 2023, 53: 1584–1595
33 Li J, Chai T, Lewis F L, et al. Off-policy interleaved Q-learning: optimal control for affine nonlinear discrete-time systems. IEEE Trans Neural Netw Learn Syst, 2018, 30: 1308–1320
34 Wang Y Y, Shi S J, Zhang Z J. A descriptor-system approach to singular perturbation of linear regulators. IEEE Trans Automat Contr, 1988, 33: 370–373
35 Jiang Y, Fan J, Jia Y, et al. Data-driven flotation process operational feedback decoupling control. Acta Autom Sin, 2019, 45: 759–770