Adaptive Dynamic Programming for Decentralized Stabilization of Uncertain Nonlinear Large-Scale Systems With Mismatched Interconnections

Xiong Yang and Haibo He, *Fellow, IEEE*

***Abstract*—This paper presents a novel decentralized control strategy for a class of uncertain nonlinear large-scale systems with mismatched interconnections. First, it is shown that the decentralized controller for the overall system can be represented by an array of optimal control policies of auxiliary subsystems. Then, within the framework of adaptive dynamic programming, a simultaneous policy iteration (SPI) algorithm is developed to solve the Hamilton–Jacobi–Bellman equations associated with auxiliary subsystem optimal control policies. The convergence of the SPI algorithm is guaranteed by an equivalence relationship. To implement the present SPI algorithm, actor and critic neural networks are applied to approximate the optimal control policies and the optimal value functions, respectively. Meanwhile, both the least squares method and the Monte Carlo integration technique are employed to derive the unknown weight parameters. Furthermore, by using Lyapunov's direct method, the overall system with the obtained decentralized controller is proved to be asymptotically stable. Finally, the effectiveness of the proposed decentralized control scheme is illustrated via simulations for nonlinear plants and unstable power systems.**

***Index Terms*—Adaptive dynamic programming (ADP), decentralized control, large-scale systems, mismatched interconnections, reinforcement learning (RL).**

I. Introduction

CHARACTERIZED by high dimensionality, uncertainty, and information structure constraints, large-scale systems have emerged in many real world applications, such as power systems, socioeconomic systems, and transportation systems. The design of stabilizing controllers for large-scale systems often cannot use one-shot methods [1]. Under this circumstance, the decentralized control approach was introduced. The key of the decentralized control method is to break down the control problem of the overall plant into several subproblems which can be handled independently. Thus, the overall system is controlled by a series of independent controllers that all together constitute a decentralized controller. A distinct advantage of the decentralized control approach is that it only requires local subsystem knowledge rather than the whole system information. Owing to this advantage, many studies on decentralized control have been reported [2]–[5].

Among the existing literature, Saberi [2] provided an optimal control method to design the decentralized controller for nonlinear large-scale systems. To be specific, the decentralized control of nonlinear interconnected systems was linked to the optimal control of each isolated subsystem. In this paper, we will follow the line of [2] to solve the decentralized control problem of uncertain nonlinear large-scale systems with mismatched interconnections from an optimal control theory perspective. Different from [2], we present an adaptive dynamic programming (ADP) method to solve the nonlinear decentralized control problem. An advantage of ADP lies in that it can avoid the well-known "curse of dimensionality" while dealing with optimal control problems of complex nonlinear systems [6]. ADP was first introduced to solve optimal control problems in 1970s [7]. A common structure used in ADP is the actor-critic architecture [8]. This architecture can be described as follows: the actor performs an action to the controlled system, and the critic evaluates the value of that action and gives feedback information to the actor. It should be noted here that, when solving optimal control problems, reinforcement learning (RL) [9] is almost in the same spirit as ADP. Thus, RL is often regarded as the synonym for ADP. Since 1970s, many ADP and RL methods have been proposed, such as policy iteration (PI) ADP [10], [11], value iteration ADP [12]–[14], robust ADP [15], [16], goal representation ADP [17], [18], single network ADP [19], [20] integral RL [21], [22], off-policy RL [23]–[25], online RL [26], [27], and Q-learning [28]–[30] (note: Q-learning is generally considered as a kind of ADP [31]).

The past several years have witnessed considerable applications of ADP to decentralized feedback control. Mu *et al.* [32] presented an ADP-based decentralized optimal control scheme for a class of continuous-time (CT) nonlinear large-scale systems with matched interconnections. To implement the control scheme, an initial admissible control was necessary. Qu *et al.* [33] proposed an adaptive-critic architecture to solve the decentralized tracking control problem of CT nonlinear large-scale systems in the framework of ADP.

Manuscript received December 4, 2017; revised March 21, 2018; accepted May 10, 2018. This work was supported in part by the National Natural Science Foundation of China under Grant 61503379, in part by the China Scholarship Council under the State Scholarship Fund, and in part by the National Science Foundation under Grant CMMI 1526835 and Grant ECCS 1731672. This paper was recommended by Associate Editor Q. Wei. *(Corresponding author: Haibo He.)*

X. Yang is with the School of Electrical and Information Engineering, Tianjin University, Tianjin 300072, China (e-mail: xiong.yang@tju.edu.cn).

H. He is with the Department of Electrical, Computer, and Biomedical Engineering, University of Rhode Island, Kingston, RI 02881 USA (e-mail: haibohe@uri.edu).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2018.2837899

The initial admissible control condition was relaxed in [33]. Later, Wang *et al.* [34] applied the same architecture as [33] to solve the decentralized stabilization problem of CT nonlinear interconnected systems. In all the above mentioned literature, the interconnections satisfied the matched condition. Generally, decentralized control approaches for nonlinear large-scale systems with *matched* interconnections do not hold for those systems with *mismatched* interconnections. Recently, Zhao *et al.* [35] designed a decentralized controller for large-scale nonlinear systems subject to unknown mismatched interconnections via ADP. After that, by using ADP, Tong *et al.* [36] presented an observer-based fuzzy decentralized optimal control scheme for strict-feedback nonlinear large-scale systems with unknown internal dynamics and unknown mismatched interconnections. In [35] and [36], the unknown functions (including unknown interconnections) were approximated either using radial basis functions or using fuzzy logic systems. Instead of approximating unknown functions of large-scale systems, Bian *et al.* [37] developed a robust ADP to obtain the decentralized optimal control of linear large-scale systems with unknown mismatched interconnections. The proposed robust ADP only used the available system states. In this sense, it was actually a data-based method. Although there already existed literature applying the data-based method (i.e., the robust ADP) to study decentralized control problems of nonlinear large-scale systems, it required the interconnections to satisfy the matched condition [38]. To the best of our knowledge, there are few studies employing the data-based method to solve the decentralized stabilization problem of CT nonlinear large-scale systems with mismatched interconnections. This motivates this paper. On the other hand, the persistence of excitation (PE) condition or the PE-like condition is necessary when implementing all the aforementioned decentralized control strategies. In general, it is challengeable to find appropriate probe noises or signals to satisfy the PE conditions for nonlinear systems, especially for nonlinear large-scale systems. This difficulty also motivates our research.

In this paper, a novel ADP-based decentralized control strategy is developed for a class of uncertain nonlinear large-scale systems with mismatched interconnections. To begin with, we prove that the decentralized controller for the overall system can be represented by an array of optimal control policies of auxiliary subsystems. Then, within the framework of ADP, we present a simultaneous PI (SPI) algorithm to solve the Hamilton–Jacobi–Bellman (HJB) equations related to auxiliary subsystem optimal control policies. Meanwhile, we establish an equivalence relationship to show the convergence of the SPI algorithm. To implement the SPI algorithm, we use actor neural networks (ANNs) and critic neural networks (CNNs) to estimate the optimal control policies and the optimal value functions, respectively. By using the least squares method and the Monte Carlo integration technique, we obtain the unknown weight parameters without the PE condition. Moreover, we demonstrate that the derived decentralized controller guarantees asymptotic stability of the overall system.

The reminder of this paper is arranged as follows. After briefly presenting the problem description in Section II, we propose the decentralized control strategy in Section III. In Section IV, we develop the SPI algorithm to solve HJB equations related to auxiliary subsystems. In Section V, we implement the SPI algorithm via the actor-critic architecture. To validate the present decentralized control scheme, we provide two examples in Section VI. Finally, Section VII gives concluding remarks and discussions.

Notation: $\mathbb{R}$ denotes the set of all real numbers. $\mathbb{N}$ represents the set of all non-negative integers. $\mathbb{Z}^+$ denotes the set of all positive integers. $\mathbb{R}^{m_i}$ and $\mathbb{R}^{n_i \times m_i}$ represent the Euclidean space of all m_i-vectors and the space of all $n_i \times m_i$ real matrices, respectively. I_{n_i} is the identity matrix with the dimension $n_i \times n_i$. T is the transposition symbol. Ω_i is a compact set of $\mathbb{R}^{n_i}$. For $A \in \mathbb{R}^{n \times m}$, $\|A\| = \sqrt{\operatorname{tr}(A^{\mathsf{T}}A)}$ is the Frobenius-norm, and $\operatorname{tr}(A^{\mathsf{T}}A)$ is the trace of $A^{\mathsf{T}}A$.

II. Problem Description

Consider the CT nonlinear large-scale system described by equations of the form

$$\begin{aligned}\dot{x}_i(t) &= f_i(x_i(t)) + g_i(x_i(t))u_i(t) + \Delta f_i(x(t))\\ x_{i0} &= x_i(0), \quad i = 1, 2, \ldots, N \end{aligned} \tag{1}$$

where $x_i \in \mathbb{R}^{n_i}$ is the measurable state of the ith subsystem, $u_i \in \mathbb{R}^{m_i}$ is the control input of the ith subsystem, $x = [x_1^{\mathsf{T}}, x_2^{\mathsf{T}}, \ldots, x_N^{\mathsf{T}}]^{\mathsf{T}} \in \mathbb{R}^n$ ($n = \sum_{i=1}^{N} n_i$) is the overall state, $f_i(x_i) \in \mathbb{R}^{n_i}$, $g_i(x_i) \in \mathbb{R}^{n_i \times m_i}$, and $\Delta f_i(x) \in \mathbb{R}^{n_i}$ are the *unknown* internal dynamics, the known input matrix, and the *uncertain* interconnection of the ith subsystem, respectively.

To facilitate subsequent analyses, we impose the following basic assumptions. The similar assumptions have been used in [31] and [39].

Assumption 1: For the ith subsystem, $f_i(x_i)$ and $g_i(x_i)$ are Lipschitz continuous in their arguments. Meanwhile, $f_i(0) = 0$, i.e., $x_i = 0$ is an equilibrium point of the ith subsystem given in (1) when $u_i(t) = 0$ and $\Delta f_i(x(t)) = 0$ for all $t \geq 0$.

Assumption 2: For the ith subsystem, the interconnection $\Delta f_i(x)$ satisfies the mismatched condition, that is

$$\Delta f_i(x) = k_i(x_i)\omega_i(x) \quad (k_i(x_i) \neq g_i(x_i))$$

where $k_i(x_i) \in \mathbb{R}^{n_i \times l_i}$ is an unknown smooth function and $\omega_i(x) \in \mathbb{R}^{l_i}$ is an uncertain function bounded as

$$\|\omega_i(x)\| \leq \sum_{s=1}^{N} a_{is}\alpha_{is}(\|x_s\|) \tag{2}$$

where $\alpha_{is}(\cdot)$, $s = 1, 2, \ldots, N$, are class $\mathcal{K}$ functions [40] and a_{is}, $s = 1, 2, \ldots, N$, are non-negative constants. Meanwhile, $\omega_i(0) = 0$, $i = 1, 2, \ldots, N$ and $\alpha_{is}(0) = 0$, $i, s = 1, 2, \ldots, N$. Let

$$\alpha_i(\|x_i\|) = \max\{\alpha_{1i}(\|x_i\|), \alpha_{2i}(\|x_i\|), \ldots, \alpha_{Ni}(\|x_i\|)\}.$$

Then, (2) can be further written as

$$\|\omega_i(x)\| \leq \sum_{s=1}^{N} b_{is}\alpha_s(\|x_s\|) \tag{3}$$

where $b_{is} \geq a_{is}\alpha_{is}(\|x_s\|) \big/ \alpha_s(\|x_s\|)$, $s = 1, 2, \ldots, N$, are non-negative constants.

Similar to (3), we impose another assumption as follows.

Assumption 3: There exist class $\mathcal{K}$ functions $\beta_\iota(\cdot)$, $\iota = 1, 2, \ldots, N$, such that

$$\left\|g_i^+(x_i)\Delta f_i(x)\right\| \le \sum_{\iota=1}^{N} c_{i\iota}\beta_\iota(\|x_\iota\|) \tag{4}$$

where $g_i^+(x_i)$ is the Moore–Penrose pseudo-inverse of $g_i(x_i)$ and $c_{i\iota}$, $\iota = 1, 2, \ldots, N$, are non-negative constants.

Objective of Control: This paper aims at finding a feedback control pair $(u_1(x_1), u_2(x_2), \ldots, u_N(x_N))$ for large-scale system (1), subject to Assumptions 1–3, which guarantees the closed-loop system (1) to be asymptotically stable.

In the above mentioned control pair, the control policies $u_i(x_i)$, $i = 1, 2, \ldots, N$, constitute the decentralized control of system (1). Meanwhile, $u_i(x_i)$ is the control policy for the ith subsystem. Therefore, to achieve the goal (i.e., obtain the decentralized control), we need to derive the control policy for each subsystem. However, the internal dynamics $f_i(x_i)$ and the interconnections $\Delta f_i(x_i)$ are unavailable. Thus, it is hard to design the controller for each subsystem directly. To address this issue, we first transform the decentralized control problem of the overall system into optimal control problems of auxiliary subsystems. Then, we solve these optimal control problems in the framework of ADP, which does not require the information of the internal dynamics $f_i(x_i)$ and the interconnection $\Delta f_i(x_i)$.

III. Decentralized Control Strategy

This section consists of two parts. First, we develop the HJB equation for the ith auxiliary subsystem. Then, we demonstrate that the decentralized controller for system (1) can be obtained via solving the HJB equations related to the auxiliary subsystems.

A. HJB Equation for the ith Auxiliary Subsystem

For the ith subsystem, projecting $\Delta f_i(x)$ onto the range of $g_i(x_i)$, we have

$$\begin{aligned}\Delta f_i(x) = {} & g_i(x_i)g_i^+(x_i)\Delta f_i(x)\\ & + \left(I_{n_i} - g_i(x_i)g_i^+(x_i)\right)\Delta f_i(x)\end{aligned} \tag{5}$$

where the first term is the matched component of $g_i(x_i)$ and the second term is the mismatched component of $g_i(x_i)$.

Based on (1) and (5), the ith auxiliary subsystem can be described as

$$\dot{x}_i = f_i(x_i) + g_i(x_i)u_i + \left(I_{n_i} - g_i(x_i)g_i^+(x_i)\right)k_i(x_i)\upsilon_i \tag{6}$$

where $\upsilon_i \in \mathbb{R}^{l_i}$ is the auxiliary control applied to cope with the mismatched component of $g_i(x_i)$.

Let the augmented control $\mu_i \in \mathbb{R}^{m_i+l_i}$ and the associated augmented input matrix $G_i(x_i) \in \mathbb{R}^{n_i\times(m_i+l_i)}$ be denoted as

$$\mu_i = \left[u_i^{\mathsf{T}}, \upsilon_i^{\mathsf{T}}\right]^{\mathsf{T}} \tag{7}$$

$$G_i(x_i) = \left[g_i(x_i), \left(I_{n_i} - g_i(x_i)g_i^+(x_i)\right)k_i(x_i)\right]. \tag{8}$$

Then, the ith auxiliary subsystem (6) can be rewritten as

$$\dot{x}_i = f_i(x_i) + G_i(x_i)\mu_i. \tag{9}$$

Associated with the ith auxiliary subsystem (9), the value function is given in the form

$$J_i(x_i(t), \mu_i) = \int_t^\infty (\Gamma_i(x_i(\tau)) + r_i(x_i(\tau), \mu_i(\tau)))\mathrm{d}\tau \tag{10}$$

where $\Gamma_i(x_i) = \eta_i P_i^2(x_i)$, $\eta_i > 0$ is a design parameter, $P_i(x_i)$ is a positive-definite function satisfying

$$\max_i\{\alpha_i(\|x_i\|), \beta_i(\|x_i\|)\} \le P_i(x_i) \tag{11}$$

and

$$r_i(x_i, \mu_i) = Q_i(x_i) + \mu_i^{\mathsf{T}}\mathcal{R}_i\mu_i$$

where $Q_i(x_i)$ is a symmetric positive-definite function, $\mathcal{R}_i = \operatorname{diag}\{\underbrace{1, \ldots, 1}_{m_i}, \underbrace{\epsilon_i, \ldots, \epsilon_i}_{l_i}\}$, and $\epsilon_i > 0$ is a constant. Owing to the characteristic of $\mathcal{R}_i$, we have $\mathcal{R}_i = \mathcal{R}_i^{(1/2)}\mathcal{R}_i^{(1/2)}$.

The optimal value function is formulated as [9]

$$V_i^*(x_i) = \min_{\mu_i\in\mathscr{A}(\Omega_i)} J_i(x_i, \mu_i) \tag{12}$$

with $\mathscr{A}(\Omega_i)$ the set of admissible control defined on Ω_i. According to [9], $V_i^*(x_i)$ can be obtained via solving the following HJB equation:

$$\begin{aligned}&\left(\nabla V_i^*(x_i)\right)^{\mathsf{T}}\left(f_i(x_i) + G_i(x_i)\mu_i^*(x_i)\right)\\ &\quad + \Gamma_i(x_i) + r_i\left(x_i, \mu_i^*(x_i)\right) = 0\end{aligned} \tag{13}$$

where $\nabla V_i^*(x_i) = \partial V_i^*(x_i)/\partial x_i$ with $V_i^*(0) = 0$, and $\mu_i^*(x_i)$ is the optimal control. Based on the stationarity condition [41], the closed-form optimal control is formulated as

$$\mu_i^*(x_i) = -\frac{1}{2}\mathcal{R}_i^{-1}G_i^{\mathsf{T}}(x_i)\nabla V_i^*(x_i). \tag{14}$$

Substituting (7) and (8) into (14), we have

$$u_i^*(x_i) = -\frac{1}{2}g_i^{\mathsf{T}}(x_i)\nabla V_i^*(x_i) \tag{15}$$

$$\upsilon_i^*(x_i) = -\frac{1}{2\epsilon_i}h_i^{\mathsf{T}}(x_i)\nabla V_i^*(x_i) \tag{16}$$

where $h_i(x_i)$ is defined as

$$h_i(x_i) = \left(I_{n_i} - g_i(x_i)g_i^+(x_i)\right)k_i(x_i). \tag{17}$$

Combining (13) and (14), the HJB equation for the ith auxiliary subsystem can be developed as

$$\begin{aligned}&\left(\nabla V_i^*(x_i)\right)^{\mathsf{T}}f_i(x_i) + Q_i(x_i) - \left\|\frac{1}{2}g_i^{\mathsf{T}}(x_i)\nabla V_i^*(x_i)\right\|^2\\ &\quad - \left\|\frac{1}{2\sqrt{\epsilon_i}}h_i^{\mathsf{T}}(x_i)\nabla V_i^*(x_i)\right\|^2 + \Gamma_i(x_i) = 0\end{aligned} \tag{18}$$

with $V_i^*(0) = 0$.

B. Decentralized Controller Design Based on Solutions of the HJB Equations

In this section, we establish a theorem to show that the decentralized controller for system (1) is composed of optimal control policies $u_1^*(x_1), u_2^*(x_2), \dots, u_N^*(x_N)$.

Theorem 1: Consider N auxiliary subsystems and the associated value functions sharing the same expressions as (9) and (10), respectively. Let Assumptions 1–3 hold. If $\upsilon_i^*(x_i)$, $i = 1, 2, \dots, N$, given as in (16) satisfy

$$\left\|\upsilon_i^*(x_i(t))\right\|^2 \le Q_i(x_i(t)), \quad t \ge t_0, \quad i = 1, 2, \dots, N \tag{19}$$

where t_0 is a non-negative threshold, and the parameters ϵ_i, $i = 1, 2, \dots, N$, are selected as

$$0 < \epsilon_i < 1/2, \quad i = 1, 2, \dots, N \tag{20}$$

then there exist N constants $\eta_i^* > 0$, $i = 1, 2, \dots, N$, such that, for every $\eta_i \ge \eta_i^*$, $i = 1, 2, \dots, N$, the control policies $u_i^*(x_i)$, $i = 1, 2, \dots, N$, given in (15) can guarantee asymptotic stability of the closed-loop system (1). That is, the feedback control pair $(u_1^*(x_1), u_2^*(x_2), \dots, u_N^*(x_N))$ is the decentralized control of system (1).

Proof: Choose the Lyapunov function candidate as

$$L(x) = \sum_{i=1}^{N} V_i^*(x_i)$$

where $V_i^*(x_i)$, $i = 1, 2, \dots, N$, are the optimal value functions defined as in (12). According to the definition of $V_i^*(x_i)$, we have $V_i^*(x_i) > 0$ $\forall x_i \neq 0$ and $V_i^*(x_i) = 0 \Leftrightarrow x_i = 0$, $i = 1, 2, \dots, N$. Therefore, $L(x)$ is positive definite.

Differentiating $L(x)$ with respect to the time variable t and using the trajectory $\dot{x}_i = f_i(x_i) + g_i(x_i)u_i^*(x_i) + \Delta f_i(x)$, $i = 1, 2, \dots, N$, it follows:

$$\begin{aligned}\dot{L}(x) = &\sum_{i=1}^{N}\Big\{\big(\nabla V_i^*(x_i)\big)^{\mathsf{T}}\big(f_i(x_i) + g_i(x_i)u_i^*(x_i)\big) \\ &+ \big(\nabla V_i^*(x_i)\big)^{\mathsf{T}}\Delta f_i(x)\Big\}.\end{aligned} \tag{21}$$

By using (5) and Assumption 2, we can see that (21) yields

$$\begin{aligned}\dot{L}(x) = &\sum_{i=1}^{N}\Big\{\big(\nabla V_i^*(x_i)\big)^{\mathsf{T}} f_i(x_i) + \big(\nabla V_i^*(x_i)\big)^{\mathsf{T}} g_i(x_i) \\ &\times u_i^*(x_i) + \big(\nabla V_i^*(x_i)\big)^{\mathsf{T}} g_i(x_i) g_i^{+}(x_i)\Delta f_i(x) \\ &+ \big(\nabla V_i^*(x_i)\big)^{\mathsf{T}} h_i(x_i)\omega_i(x)\Big\}\end{aligned} \tag{22}$$

with $h_i(x_i)$ defined as in (17).

From (15), (16), and (18), we find

$$\begin{cases}\big(\nabla V_i^*(x_i)\big)^{\mathsf{T}} f_i(x_i) = -\eta_i P_i^2(x_i) - Q_i(x_i) \\ \qquad\qquad\qquad + \left\|u_i^*(x_i)\right\|^2 + \epsilon_i\left\|\upsilon_i^*(x_i)\right\|^2 \\ \big(\nabla V_i^*(x_i)\big)^{\mathsf{T}} g_i(x_i) = -2\big(u_i^*(x_i)\big)^{\mathsf{T}} \\ \big(\nabla V_i^*(x_i)\big)^{\mathsf{T}} h_i(x_i) = -2\epsilon_i\big(\upsilon_i^*(x_i)\big)^{\mathsf{T}}.\end{cases} \tag{23}$$

Substituting (23) into (22), we obtain

$$\begin{aligned}\dot{L}(x) = &\sum_{i=1}^{N}\bigg\{-\eta_i P_i^2(x_i) - Q_i(x_i) - \left\|u_i^*(x_i)\right\|^2 \\ &+ \epsilon_i\left\|\upsilon_i^*(x_i)\right\|^2 \underbrace{-2\big(u_i^*(x_i)\big)^{\mathsf{T}} g_i^{+}(x_i)\Delta f_i(x)}_{\pounds_1(x)} \\ &\underbrace{-2\epsilon_i\big(\upsilon_i^*(x_i)\big)^{\mathsf{T}}\omega_i(x)}_{\pounds_2(x)}\bigg\}.\end{aligned} \tag{24}$$

Applying the Cauchy–Schwarz inequality to $\pounds_1(x)$ in (24) and using (3) and (11), it follows:

$$\begin{aligned}\pounds_1(x) \le\ & 2\left\|u_i^*(x_i)\right\|\left\|g_i^{+}(x_i)\Delta f_i(x)\right\| \\ \le\ & 2\left\|u_i^*(x_i)\right\|\sum_{s=1}^{N} b_{is}P_s(x_s).\end{aligned} \tag{25}$$

Similarly, using (4) and (11) and noting that $0 < \epsilon_i < 1/2$ in (20), we can conclude that $\pounds_2(x)$ in (24) implies

$$\begin{aligned}\pounds_2(x) \le\ & 2\epsilon_i\left\|\upsilon_i^*(x_i)\right\|\left\|\omega_i(x)\right\| \\ \le\ & 2\left\|\upsilon_i^*(x_i)\right\|\sum_{\iota=1}^{N} c_{i\iota}P_\iota(x_\iota).\end{aligned} \tag{26}$$

From (24)–(26), we can see that

$$\begin{aligned}\dot{L}(x) \le &-\sum_{i=1}^{N} 2\epsilon_i\Big(Q_i(x_i) - \left\|\upsilon_i^*(x_i)\right\|^2\Big) - \sum_{i=1}^{N}(1 - 2\epsilon_i)Q_i(x_i) \\ &-\sum_{i=1}^{N}\bigg\{\eta_i P_i^2(x_i) + \left\|u_i^*(x_i)\right\|^2 + \epsilon_i\left\|\upsilon_i^*(x_i)\right\|^2 \\ &- 2\left\|u_i^*(x_i)\right\|\sum_{s=1}^{N} b_{is}P_s(x_s) - 2\left\|\upsilon_i^*(x_i)\right\|\sum_{\iota=1}^{N} c_{i\iota}P_\iota(x_\iota)\bigg\}.\end{aligned} \tag{27}$$

Denote

$$\begin{aligned}\Lambda_1 &= \mathrm{diag}\{\eta_1, \eta_2, \dots, \eta_N\} \\ \Lambda_2 &= \mathrm{diag}\Bigg\{\underbrace{1, 1, \dots, 1}_{N}\Bigg\} \\ \Lambda_3 &= \mathrm{diag}\{\epsilon_1, \epsilon_2, \dots, \epsilon_N\}\end{aligned}$$

$$B = \begin{bmatrix} b_{11} & \cdots & b_{1N} \\ b_{21} & \cdots & b_{2N} \\ \vdots & \ddots & \vdots \\ b_{N1} & \cdots & b_{NN}\end{bmatrix} \text{ and } C = \begin{bmatrix} c_{11} & \cdots & c_{1N} \\ c_{21} & \cdots & c_{2N} \\ \vdots & \ddots & \vdots \\ c_{N1} & \cdots & c_{NN}\end{bmatrix}.$$

Let

$$\begin{aligned}\zeta = \big[&P_1(x_1), \dots, P_N(x_N), \left\|u_1^*(x_1)\right\|, \dots, \left\|u_N^*(x_N)\right\| \\ &\left\|\upsilon_1^*(x_1)\right\|, \dots, \left\|\upsilon_N^*(x_N)\right\|\big]^{\mathsf{T}}.\end{aligned}$$

Then, by using (19) and (20), we can conclude that (27) yields (for every $t \ge t_0$)

$$\dot{L}(x) \le -\sum_{i=1}^{N}(1 - 2\epsilon_i)Q_i(x_i) - \zeta^{\mathsf{T}}\mathcal{A}\zeta \tag{28}$$

where

$$\mathcal{A} = \begin{bmatrix} \Lambda_1 & B^{\mathsf{T}} & C^{\mathsf{T}} \\ B & \Lambda_2 & 0_{N\times N} \\ C & 0_{N\times N} & \Lambda_3 \end{bmatrix} \tag{29}$$

and $0_{N\times N} = \operatorname{diag}\{\underbrace{0, 0, \ldots, 0}_{N}\}$. Observing the expression $\mathcal{A}$ given in (29), we can find that $\mathcal{A}$ is able to be kept positive definite by choosing sufficiently large η_i, $i = 1, 2, \ldots, N$, in Λ_1. Therefore, there exist η_i^*, $i = 1, 2, \ldots, N$, such that $\eta_i \geq \eta_i^*$, $i = 1, 2, \ldots, N$, imply $-\zeta^{\mathsf{T}}\mathcal{A}\zeta < 0$. Thus, from (28), we obtain

$$\dot{L}(x(t)) \leq -\sum_{i=1}^{N}(1-2\epsilon_i)Q_i(x_i(t)), \quad t \geq t_0. \tag{30}$$

Note that, for each index i, the positive definite matrix $Q_i(x_i)$ implies $\rho_i x_i^{\mathsf{T}} x_i \leq Q_i(x_i)$ with the constant $\rho_i > 0$, Then, from (30), we have

$$\dot{L}(x(t)) \leq -\sum_{i=1}^{N}\rho_i(1-2\epsilon_i)\|x_i(t)\|^2, \quad t \geq t_0. \tag{31}$$

Integrating both sides of (31) over the time interval $[t_0, \infty)$, it follows:

$$\sum_{i=1}^{N}\rho_i(1-2\epsilon_i)\int_{t_0}^{\infty}\|x_i(t)\|^2 \mathrm{d}t \leq L(x(t_0)) - L(x(\infty)).$$

Then, after some computations, we derive

$$\int_{t_0}^{\infty}\|x_i(t)\|^2\mathrm{d}t \leq \frac{L(x(t_0)) - L(x(\infty))}{\rho_i(1-2\epsilon_i)} \tag{32}$$

with $i = 1, 2, \ldots, N$. Because the right side of (32) is finite, using Barbalat's lemma [42], we obtain

$$\lim_{t\to\infty}\|x_i(t)\| = 0, \quad i = 1, 2, \ldots, N.$$

This verifies that system (1) is asymptotically stable with optimal control policies $u_i^*(x_i)$, $i = 1, 2, \ldots, N$. ■

Remark 1: Generally, the condition (19) cannot be verified directly. This is mainly because $\upsilon_i^*(x_i)$ given in (16) has no direct connection with the positive-definite function $Q_i(x_i)$. We only know that both $\upsilon_i^*(x_i)$ and $Q_i(x_i)$ are functions with respect to the state x_i. Moreover, as indicated in (16), $\upsilon_i^*(x_i)$ is linked with $\nabla V_i^*(x_i)$. The explicit expression of $\nabla V_i^*(x_i)$ usually cannot be obtained. Owing to this difficulty, the validity of (19) is often verified via simulation results (see [43], [44]). In this paper, we will illustrate the validity of (19) in Section VI.

Theorem 1 indicates that the decentralized controller for system (1) can be represented by an array of optimal control policies $u_i^*(x_i)$, $i = 1, 2, \ldots, N$. Hence, we need to solve the HJB equation (18) for the ith auxiliary subsystem. However, (18) is a nonlinear partial differential equation with respect to $V_i^*(x_i)$, which often does not exist the closed-form solution. In addition, the knowledge of $f_i(x_i)$ and $G_i(x_i)$ (note: $G_i(x_i)$ contains $k_i(x_i)$) is unavailable, which increases the difficulty in solving (18). To overcome these difficulties, we will present a SPI algorithm to approximately solve (18) in the framework of ADP.

IV. SPI Algorithm to Solve HJB Equations

This section first introduces the traditional PI algorithm. Then, based on the traditional PI algorithm, the SPI algorithm is developed.

For the ith auxiliary subsystem, the HJB equation is described as (18). According to [45], (18) can be solved via the traditional PI algorithm as follows.

1) Find an initial control $\mu_i^{(0)}(x_i) \in \mathscr{A}(\Omega_i)$.

2) For every $j \in \mathbb{N}$, obtain $V_i^{(j)}(x_i)$ by solving

$$\Big(\nabla V_i^{(j)}(x_i)\Big)^{\mathsf{T}}\Big(f_i(x_i) + G_i(x_i)\mu_i^{(j)}(x_i)\Big) + \Gamma_i(x_i) + r_i\Big(x_i, \mu_i^{(j)}(x_i)\Big) = 0 \tag{33}$$

with $V_i^{(j)}(0) = 0$.

3) Update the control policy via

$$\mu_i^{(j+1)}(x_i) = -\frac{1}{2}\mathcal{R}^{-1}G_i^{\mathsf{T}}(x_i)\nabla V_i^{(j)}(x_i). \tag{34}$$

To illustrate the convergence of PI (33) and (34), we establish the following theorem.

Theorem 2: Let $V_i^{(j)}(x_i)$ and $\mu_i{}^{(j)}(x_i)$ be generated from (33) and (34). If $\mu_i^{(0)}(x_i) \in \mathscr{A}(\Omega_i)$, then, for every $x_i \in \Omega_i$

$$\lim_{j\to\infty} V_i^{(j)}(x_i) = V_i^*(x_i) \quad \text{and} \quad \lim_{j\to\infty}\mu_i^{(j)}(x_i) = \mu_i^*(x_i).$$

Proof: Since the proof is almost the same as [45, Th. 4], we omit it here. ■

Remark 2: To implement the PI (33) and (34), the priori knowledge of $f_i(x_i)$ and $G_i(x_i)$ is required to be available. Owing to the unavailability of the knowledge of $f_i(x_i)$ and $G_i(x_i)$, the PI (33) and (34) cannot be employed to solve (18).

For the sake of solving (18), we develop a SPI algorithm based on (33) and (34). By using the SPI algorithm, we no longer need the priori knowledge concerning system dynamics $f_i(x_i)$ and $G_i(x_i)$.

Rewrite the ith auxiliary subsystem (9) as

$$\dot{x}_i = f_i(x_i) + G_i(x_i)\mu_i{}^{(j)}(x_i) + G_i(x_i)\Big(\mu_i - \mu_i^{(j)}(x_i)\Big). \tag{35}$$

Differentiating $V_i^{(j)}(x_i)$ with respect to the time variable t and using the trajectory (35) as well as (33) and (34), it follows:

$$\begin{aligned}\dot{V}_i^{(j)}(x_i) = &\Big(\nabla V_i^{(j)}(x_i)\Big)^{\mathsf{T}}\Big(f_i(x_i) + G_i(x_i)\mu_i^{(j)}(x_i)\Big) \\ &+ \Big(\nabla V_i^{(j)}(x_i)\Big)^{\mathsf{T}}G_i(x_i)\Big(\mu_i - \mu_i^{(j)}(x_i)\Big) \\ = &-\Gamma_i(x_i) - r_i\Big(x_i, \mu_i^{(j)}(x_i)\Big) \\ &- 2\Big(\mu_i^{(j+1)}(x_i)\Big)^{\mathsf{T}}\mathcal{R}_i\Big(\mu_i - \mu_i^{(j)}(x_i)\Big). \end{aligned} \tag{36}$$

Integrating (36) over the time interval $[t, t+\Delta t]$, we have [note: for brevity, we write $\mu_i^{(j)}(x_i(\tau))$ and $\mu_i^{(j+1)}(x_i(\tau))$ as $\mu_i^{(j)}(\tau)$ and $\mu_i^{(j+1)}(\tau)$, respectively]

$$\begin{aligned}V_i^{(j)}&(x_i(t+\Delta t)) - V_i^{(j)}(x_i(t)) \\ &= -\int_t^{t+\Delta t}\Big(\Gamma_i(x_i(\tau)) + r_i\Big(x_i(\tau), \mu_i^{(j)}(\tau)\Big)\Big)\mathrm{d}\tau \\ &\quad - 2\int_t^{t+\Delta t}\Big(\mu_i^{(j+1)}(\tau)\Big)^{\mathsf{T}}\mathcal{R}_i\Big(\mu_i(\tau) - \mu_i^{(j)}(\tau)\Big)\mathrm{d}\tau.\end{aligned}$$

Let

$$n_{e_i} = \mu_i - \mu_i^{(j)}(x_i). \tag{37}$$

Then the SPI algorithm can be described as follows.

1) Find an initial control ${\mu_i}^{(0)}(x_i) \in \mathscr{A}(\Omega_i)$.
2) For every $j \in \mathbb{N}$, derive $V_i^{(j)}(x_i)$ and $\mu_i^{(j+1)}(x_i)$ simultaneously via solving

$$\begin{aligned} V_i^{(j)}(x_i(t)) = &\int_t^{t+\Delta t} (\Gamma_i(x_i(\tau)) + Q_i(x_i(\tau)))\mathrm{d}\tau \\ &+ \int_t^{t+\Delta t} \Big(\mu_i^{(j)}(\tau)\Big)^{\mathsf{T}} \mathcal{R}_i \mu_i^{(j)}(\tau)\mathrm{d}\tau \\ &+ 2\int_t^{t+\Delta t} \Big(\mu_i^{(j+1)}(\tau)\Big)^{\mathsf{T}} \mathcal{R}_i n_{e_i}(\tau)\mathrm{d}\tau \\ &+ V_i^{(j)}(x_i(t+\Delta t)). \end{aligned} \tag{38}$$

Remark 3: Two notes about the SPI algorithm are given as follows [in what follows we call the present SPI algorithm as the SPI (38) for convenience].

1) The key of implementing the SPI (38) is to find an initial admissible control, i.e., ${\mu_i}^{(0)}(x_i) \in \mathscr{A}(\Omega_i)$. However, there is no general method proposed to find such a control. In this paper, the initial admissible control is obtained through the trail-and-error method, which shares the same spirit as [46].
2) From the expression (38), we can see that the knowledge of $f_i(x_i)$ and $G_i(x_i)$ is not necessary while implementing the SPI (38). Actually, only data pairs $\Big(x_i, \mu_i^{(j)}\Big)$ are used. In comparison with the PI (33) and (34), this is an advantage of the SPI (38).

As for the SPI (38), two questions will be asked.

(*I*) Is it possible to obtain $V_i^{(j)}(x_i)$ and $\mu_i^{(j+1)}(x_i)$ simultaneously only by solving (38)?

(*II*) For every $x_i \in \Omega_i$, will the sequences

$$\Big\{V_i^{(j)}(x_i)\Big\} \quad \text{and} \quad \Big\{\mu_i^{(j+1)}(x_i)\Big\}$$

generated from (38) be convergent?

Next, we first answer the question (*II*). Then, we answer the question (*I*) in Section V.

Lemma 1: Assume that the mappings $\gamma_i : \Omega_i \to \mathbb{R}^{m_i+l_i}$, $d_i : \Omega_i \to \mathbb{R}$, and $y_i \in \mathbb{R}^{m_i+l_i}$ is the variable function. If, for every $x_i \in \Omega_i$ and $y_i \neq 0$, the equality $\gamma_i^{\mathsf{T}}(x_i)y_i = d_i(x_i)$ holds, then $\gamma_i(x_i) = 0$ and $d_i(x_i) = 0$.

Proof: Given that there exists a fixed $y_i^0 \neq 0$ such that $\gamma_i^{\mathsf{T}}(x_i)y_i^0 = d_i(x_i)$. Then, we have

$$\gamma_i^{\mathsf{T}}(x_i)\Big(y_i - y_i^0\Big) = 0 \quad \forall x_i, \quad \forall y_i \neq 0.$$

If denoting $F(x_i, y_i) = \gamma_i^{\mathsf{T}}(x_i)(y_i - y_i^0)$, then we obtain

$$F(x_i, y_i) = 0 \qquad \forall x_i,\ \forall y_i \neq 0. \tag{39}$$

Taking the partial derivative of (39) with respect to y_i, we can see that

$$0 = \frac{\partial F(x_i, y_i)}{\partial y_i} = \frac{\partial\big(\gamma_i^{\mathsf{T}}(x_i)\big(y_i - y_i^0\big)\big)}{\partial y_i} = \gamma_i(x_i).$$

Thus, $d_i(x_i) = \gamma_i^{\mathsf{T}}(x_i)y_i = 0$. ■

Theorem 3: The SPI (38) is valid if and only if the PI (33) and (34) holds.

Proof (Necessity): Owing to $V_i^{(j)}(x_i)$ and $\mu_i^{(j+1)}(x_i)$ generated from (33) and (34), we can easily obtain (38) by using (35)–(37).

(Sufficiency): Let $\Delta t \to 0$. Then (38) yields

$$\begin{aligned} -\lim_{\Delta t \to 0} \frac{1}{\Delta t}&\Big(V_i^{(j)}(x_i(t+\Delta t)) - V_i^{(j)}(x_i(t))\Big) \\ &= \lim_{\Delta t \to 0} \frac{1}{\Delta t}\int_t^{t+\Delta t} (\Gamma_i(x_i(\tau)) + Q_i(x_i(\tau)))\mathrm{d}\tau \\ &+ \lim_{\Delta t \to 0} \frac{1}{\Delta t}\int_t^{t+\Delta t} \Big(\mu_i^{(j)}(\tau)\Big)^{\mathsf{T}} \mathcal{R}_i \mu_i^{(j)}(\tau)\mathrm{d}\tau \\ &+ \lim_{\Delta t \to 0} \frac{2}{\Delta t}\int_t^{t+\Delta t} \Big(\mu_i^{(j+1)}(\tau)\Big)^{\mathsf{T}} \mathcal{R}_i n_{e_i}(\tau)\mathrm{d}\tau. \end{aligned} \tag{40}$$

According to the definition of derivative [41], (40) implies

$$\begin{aligned} -\Big(\nabla V_i^{(j)}(x_i)\Big)^{\mathsf{T}} \dot{x}_i = &\ \Gamma_i(x_i) + Q_i(x_i) \\ &+ \Big(\mu_i^{(j)}(x_i)\Big)^{\mathsf{T}} \mathcal{R}_i \mu_i^{(j)}(x_i) \\ &+ 2\Big(\mu^{(j+1)}(x_i)\Big)^{\mathsf{T}} \mathcal{R}_i n_{e_i}. \end{aligned} \tag{41}$$

Note that (35) and (37) yield

$$\dot{x}_i = f_i(x_i) + G_i(x_i)\mu_i^{(j)}(x_i) + G_i(x_i)n_{e_i}.$$

Then (41) can be developed as

$$\begin{aligned} -\bigg[\Big(\nabla V_i^{(j)}(x_i)\Big)^{\mathsf{T}} G_i(x_i) + 2\Big(\mu_i^{(j+1)}(x_i)\Big)^{\mathsf{T}} \mathcal{R}_i\bigg] n_{e_i} \\ = \Big(\nabla V_i^{(j)}(x_i)\Big)^{\mathsf{T}}\Big(f_i(x_i) + G_i(x_i)\mu_i^{(j)}(x_i)\Big) \\ + \Gamma_i(x_i) + Q_i(x_i) + \Big(\mu_i^{(j)}(x_i)\Big)^{\mathsf{T}} \mathcal{R}_i \mu_i^{(j)}(x_i). \end{aligned} \tag{42}$$

Owing to the validity of (42) for arbitrary n_{e_i}, we can conclude that (42) holds for every $n_{e_i} \neq 0$. Then, by Lemma 1, we have

$$\begin{aligned} \Big(\nabla V_i^{(j)}(x_i)\Big)^{\mathsf{T}}\Big(f_i(x_i) + G_i(x_i)\mu_i^{(j)}(x_i)\Big) + \Gamma_i(x_i) \\ + Q_i(x_i) + \Big(\mu_i^{(j)}(x_i)\Big)^{\mathsf{T}} \mathcal{R}_i \mu_i^{(j)}(x_i) = 0 \end{aligned} \tag{43}$$

$$\Big(\nabla V_i^{(j)}(x_i)\Big)^{\mathsf{T}} + 2\Big(\mu_i^{(j+1)}(x_i)\Big)^{\mathsf{T}} \mathcal{R}_i = 0. \tag{44}$$

From (43) and (44), one can easily obtain (33) and (34). ■

Theorem 4: Let $\mu_i^{(0)}(x_i) \in \mathscr{A}(\Omega_i)$. If the sequence pairs $\Big\{V_i^{(j)}(x_i), \mu_i^{(j)}(x_i)\Big\}$ are determined by (38), then, for every $x_i \in \Omega_i$

$$\lim_{j \to \infty} V_i^{(j)}(x_i) = V_i^*(x_i) \quad \text{and} \quad \lim_{j \to \infty} \mu_i^{(j)}(x_i) = \mu_i^*(x_i)$$

where $V_i^*(x_i)$ is the optimal value function given in (12) and $\mu_i^*(x_i)$ is the associated optimal control defined as in (14).

Proof: According to Theorem 3, the sequence pairs $\Big\{V_i^{(j)}(x_i), \mu_i^{(j)}(x_i)\Big\}$ determined by (38) can be viewed as the sequence pairs $\Big\{V_i^{(j)}(x_i), \mu_i^{(j)}(x_i)\Big\}$ generated from (33) and (34). Thus, by using Theorem 2, we can conclude $\lim_{j\to\infty} V_i^{(j)}(x_i) = V_i^*(x_i)$ and $\lim_{j\to\infty} \mu_i^{(j)}(x_i) = \mu_i^*(x_i)$ for every $x_i \in \Omega_i$. ■

Remark 4: Theorem 4 shows that the sequences $\{V_i^{(j)}(x_i)\}$ and $\{\mu_i^{(j+1)}(x_i)\}$ generated from (38) are convergent. Hence, the question *(II)* has been well addressed.

V. Implement the SPI Algorithm via Actor-Critic Architecture

Based on the definition μ_i given in (7), we let

$$\mu_i^{(j+1)}(x_i) = \left[\left(u_i^{(j+1)}(x_i)\right)^{\mathsf{T}}, \left(\upsilon_i^{(j+1)}(x_i)\right)^{\mathsf{T}}\right]^{\mathsf{T}}$$

where

$$u_i^{(j+1)}(x_i) = \left[u_{i1}^{(j+1)}(x_i), \ldots, u_{im_i}^{(j+1)}(x_i)\right]^{\mathsf{T}} \tag{45}$$

$$\upsilon_i^{(j+1)}(x_i) = \left[\upsilon_{i1}^{(j+1)}(x_i), \ldots, \upsilon_{il_i}^{(j+1)}(x_i)\right]^{\mathsf{T}} \tag{46}$$

with $u_{i\kappa}^{(j+1)}(x_i) \in \mathbb{R}$, $\kappa = 1, 2, \ldots, m_i$ and $\upsilon_{i\pi}^{(j+1)}(x_i) \in \mathbb{R}$, $\pi = 1, 2, \ldots, l_i$.

According to the approximation theory proposed in [47], $V_i^{(j)}(x_i)$, $u_{i\kappa}^{(j+1)}(x_i)$, and $\upsilon_{i\pi}^{(j+1)}(x_i)$ can be, respectively, approximated by the CNN and ANNs over Ω_i as

$$\hat{V}_i^{(j)}(x_i) = \sum_{\ell=1}^{\tilde{n}_1} \theta_{i\ell}^{(j)} \sigma_{i\ell}(x_i) = \left(\theta_i^{(j)}\right)^{\mathsf{T}} \sigma_i(x_i) \tag{47}$$

$$\hat{u}_{i\kappa}^{(j+1)}(x_i) = \sum_{p=1}^{\tilde{n}_2} \lambda_{i\kappa,p}^{(j)} \psi_{i\kappa,p}(x_i) = \left(\bar{\lambda}_{i\kappa}^{(j)}\right)^{\mathsf{T}} \bar{\psi}_{i\kappa}(x_i) \tag{48}$$

$$\hat{\upsilon}_{i\pi}^{(j+1)}(x_i) = \sum_{q=1}^{\tilde{n}_3} \nu_{i\pi,q}^{(j)} \phi_{i\pi,q}(x_i) = \left(\bar{\nu}_{i\pi}^{(j)}\right)^{\mathsf{T}} \bar{\phi}_{i\pi}(x_i) \tag{49}$$

where $\theta_i^{(j)} = [\theta_{i1}^{(j)}, \ldots, \theta_{i\tilde{n}_1}^{(j)}]^{\mathsf{T}} \in \mathbb{R}^{\tilde{n}_1}$ is the constant CNN weight vector, $\bar{\lambda}_{i\kappa}^{(j)} = [\lambda_{i\kappa,1}^{(j)}, \ldots, \lambda_{i\kappa,\tilde{n}_2}^{(j)}]^{\mathsf{T}} \in \mathbb{R}^{\tilde{n}_2}$ and $\bar{\nu}_{i\pi}^{(j)} = [\nu_{i\pi,1}^{(j)}, \ldots, \nu_{i\pi,\tilde{n}_3}^{(j)}]^{\mathsf{T}} \in \mathbb{R}^{\tilde{n}_3}$ are the constant ANN weight vectors, $\tilde{n}_1 \in \mathbb{Z}^+$ is the number of neurons in the CNN, $\tilde{n}_2 \in \mathbb{Z}^+$ and $\tilde{n}_3 \in \mathbb{Z}^+$ are the numbers of neurons in ANNs, $\sigma_i(x_i) = [\sigma_{i1}(x_i), \ldots, \sigma_{i\tilde{n}_1}(x_i)]^{\mathsf{T}} \in \mathbb{R}^{\tilde{n}_1}$ is the vector activation function of the CNN with $\sigma_{i\ell}(x_i) \in C^1(\Omega_i)$ and $\sigma_{i\ell}(0) = 0$ $(\ell = 1, 2, \ldots, \tilde{n}_1)$, $\bar{\psi}_{i\kappa}(x_i) = [\psi_{i\kappa,1}(x_i), \ldots, \psi_{i\kappa,\tilde{n}_2}(x_i)]^{\mathsf{T}} \in \mathbb{R}^{\tilde{n}_2}$ and $\bar{\phi}_{i\pi}(x_i) = [\phi_{i\pi,1}(x_i), \ldots, \phi_{i\pi,\tilde{n}_3}(x_i)]^{\mathsf{T}} \in \mathbb{R}^{\tilde{n}_3}$ are the vector activation functions of ANNs with $\psi_{i\kappa,p}(x_i) \in C^1(\Omega_i)$, $\psi_{i\kappa,p}(0) = 0$ $(p = 1, 2, \ldots, \tilde{n}_2)$, and $\phi_{i\pi,q}(x_i) \in C^1(\Omega_i)$, $\phi_{i\pi,g}(0) = 0$ $(q = 1, 2, \ldots, \tilde{n}_3)$. Moreover, the sets $\{\sigma_{i\ell}(x_i)\}_{\ell=1}^{\tilde{n}_1}$, $\{\psi_{i\kappa,p}(x_i)\}_{p=1}^{\tilde{n}_2}$ and $\{\phi_{i\pi,q}(x_i)\}_{q=1}^{\tilde{n}_3}$ are linearly independent, respectively.

By using (48) and (49), we approximate $u_i^{(j+1)}(x_i)$ given in (45) and $\upsilon_i^{(j+1)}(x_i)$ given in (46) via ANNs over Ω_i as

$$\begin{aligned}\hat{u}_i^{(j+1)}(x_i) &= \left[\left(\bar{\lambda}_{i1}^{(j)}\right)^{\mathsf{T}} \bar{\psi}_{i1}(x_i), \ldots, \left(\bar{\lambda}_{im_i}^{(j)}\right)^{\mathsf{T}} \bar{\psi}_{im_i}(x_i)\right]^{\mathsf{T}} \\ \hat{\upsilon}_i^{(j+1)}(x_i) &= \left[\left(\bar{\nu}_{i1}^{(j)}\right)^{\mathsf{T}} \bar{\phi}_{i1}(x_i), \ldots, \left(\bar{\nu}_{il_i}^{(j)}\right)^{\mathsf{T}} \bar{\phi}_{il_i}(x_i)\right]^{\mathsf{T}}.\end{aligned} \tag{50}$$

Then the estimated value of $\mu_i^{(j+1)}(x_i)$ is

$$\hat{\mu}_i^{(j+1)}(x_i) = \left[\left(\hat{u}_i^{(j+1)}(x_i)\right)^{\mathsf{T}}, \left(\hat{\upsilon}_i^{(j+1)}(x_i)\right)^{\mathsf{T}}\right]^{\mathsf{T}}. \tag{51}$$

Rewrite (38) as

$$\begin{aligned}0 = \ & V_i^{(j)}(x_i(t+\Delta t)) - V_i^{(j)}(x_i(t)) \\ & + 2\int_t^{t+\Delta t} \left(\mu_i^{(j+1)}(\tau)\right)^{\mathsf{T}} \mathcal{R}_i n_{e_i}(\tau) \mathrm{d}\tau \\ & + \Sigma\left(x_i, \mu_i^{(j)}(x_i)\right)\end{aligned} \tag{52}$$

where

$$\begin{aligned}\Sigma\left(x_i, \mu_i^{(j)}(x_i)\right) = & \int_t^{t+\Delta t} (\Gamma_i(x_i(\tau)) + Q_i(x_i(\tau))) \mathrm{d}\tau \\ & + \int_t^{t+\Delta t} \left(\mu_i^{(j)}(\tau)\right)^{\mathsf{T}} \mathcal{R}_i \mu_i^{(j)}(\tau) \mathrm{d}\tau.\end{aligned}$$

Due to $n_{e_i} \in \mathbb{R}^{m_i+l_i}$, we denote

$$n_{e_i} = \left[n_{e_{i1}}, \ldots, n_{e_{im_i}}, n_{e_{i(m_i+1)}}, \ldots, n_{e_{i(m_i+l_i)}}\right]^{\mathsf{T}} \in \mathbb{R}^{m_i+l_i}.$$

In (52), let $V_i^{(j)}(x_i)$ and $\mu_i^{(j+1)}(x_i)$ be replaced by $\hat{V}_i^{(j)}(x_i)$ given in (47) and $\hat{\mu}_i^{(j+1)}(x_i)$ given in (51), respectively. Then, the residual error $\delta_i^{(j)}(x_i, n_{e_i}) \in \mathbb{R}$ [48] is formulated as

$$\begin{aligned}\delta_i^{(j)}\left(x_i, n_{e_i}\right) = & \left(\theta_i^{(j)}\right)^{\mathsf{T}} (\sigma_i(x_i(t+\Delta t)) - \sigma_i(x_i(t))) \\ & + 2\sum_{\kappa=1}^{m_i} \int_t^{t+\Delta t} \left(\bar{\lambda}_{i\kappa}^{(j)}\right)^{\mathsf{T}} \bar{\psi}_{i\kappa}(x_i(\tau)) n_{e_{i\kappa}}(\tau) \mathrm{d}\tau \\ & + 2\epsilon_i \sum_{\pi=1}^{l_i} \int_t^{t+\Delta t} \Bigg(\left(\bar{\nu}_{i\pi}^{(j)}\right)^{\mathsf{T}} \bar{\phi}_{i\pi}(x_i(\tau)) \\ & \qquad \times n_{e_{i(m_i+\pi)}}(\tau)\Bigg) \mathrm{d}\tau \\ & + \Sigma\left(x_i, \hat{\mu}_i^{(j)}(x_i)\right).\end{aligned} \tag{53}$$

Note that, for $M^{\mathsf{T}}N \in \mathbb{R}$, the equality $M^{\mathsf{T}}N = N^{\mathsf{T}}M$ holds. Therefore, from (53), we have

$$\begin{aligned}\delta_i^{(j)}\left(x_i, n_{e_i}\right) = & (\sigma_i(x_i(t+\Delta t)) - \sigma_i(x_i(t)))^{\mathsf{T}} \theta_i^{(j)} \\ & + 2\sum_{\kappa=1}^{m_i} \int_t^{t+\Delta t} n_{e_{i\kappa}}(\tau) \bar{\psi}_{i\kappa}^{\mathsf{T}}(x_i(\tau)) \mathrm{d}\tau \bar{\lambda}_{i\kappa}^{(j)} \\ & + 2\epsilon_i \sum_{\pi=1}^{l_i} \int_t^{t+\Delta t} n_{e_{i(m_i+\pi)}}(\tau) \bar{\phi}_{i\pi}^{\mathsf{T}}(x_i(\tau)) \mathrm{d}\tau \\ & \times \bar{\nu}_{i\pi}^{(j)} + \Sigma\left(x_i, \hat{\mu}_i^{(j)}(x_i)\right) \\ = \ & \Psi_i\left(x_i, n_{e_i}\right) \Phi_i^{(j)} + \Sigma\left(x_i, \hat{\mu}_i^{(j)}(x_i)\right)\end{aligned} \tag{54}$$

where

$$\Phi_i^{(j)} = \left[\left(\theta_i^{(j)}\right)^{\mathsf{T}}, \left(\bar{\lambda}_{i1}^{(j)}\right)^{\mathsf{T}}, \ldots, \left(\bar{\lambda}_{im_i}^{(j)}\right)^{\mathsf{T}}, \bar{\nu}_{i1}^{(j)}, \ldots, \bar{\nu}_{il_i}^{(j)}\right]^{\mathsf{T}} \tag{55}$$

and

$$\Psi_i\left(x_i, n_{e_i}\right) = \left[\Delta\sigma_i(x_i(t)), \xi_{i1}, \ldots, \xi_{im_i}, \hbar_{i1}, \ldots, \hbar_{il_i}\right]$$

with $\Delta\sigma_i(x_i(t)) = [\sigma_i(x_i(t+\Delta t)) - \sigma_i(x_i(t))]^{\mathsf{T}}$ and

$$\begin{aligned}\xi_{i\kappa} &= 2\int_t^{t+\Delta t} n_{e_{i\kappa}}(\tau) \bar{\psi}_{i\kappa}^{\mathsf{T}}(x_i(\tau)) \mathrm{d}\tau, \quad \kappa = 1, \ldots, m_i \\ \hbar_{i\pi} &= 2\epsilon_i \int_t^{t+\Delta t} n_{e_{i(m_i+\pi)}}(\tau) \bar{\phi}_{i\pi}^{\mathsf{T}}(x_i(\tau)) \mathrm{d}\tau, \quad \pi = 1, \ldots, l_i.\end{aligned}$$

To calculate $\Phi_i^{(j)}$ in (54), we use the method of weighted residuals [48]. To be specific, we can obtain $\Phi_i^{(j)}$ by projecting the residual error $\delta_i^{(j)}(x_i, n_{e_i})$ onto the term $\partial \delta_i^{(j)}(x_i, n_{e_i})/\partial \Phi_i^{(j)}$ and letting the result be zero. This procedure can be formulated as

$$\left\langle \frac{\partial \delta_i^{(j)}\left(x_i, n_{e_i}\right)}{\partial \Phi_i^{(j)}}, \delta_i^{(j)}\left(x_i, n_{e_i}\right) \right\rangle_{\tilde{\Omega}_i} = 0 \tag{56}$$

where $\langle \cdot, \cdot \rangle_{\tilde{\Omega}_i}$ denotes the $\mathcal{L}_2$ inner product defined on $\tilde{\Omega}_i$ [49], $\tilde{\Omega}_i = \{(x_i, n_{e_i}) | x_i \in \Omega_i, n_{e_i} \in \mathcal{E}_i\}$, and $\mathcal{E}_i$ is the set of n_{e_i}.

Substituting (54) into (56), it follows:

$$\begin{aligned} &\left\langle \Psi_i\left(x_i, n_{e_i}\right), \Psi_i\left(x_i, n_{e_i}\right) \right\rangle_{\tilde{\Omega}_i} \Phi_i^{(j)} \\ &\quad + \left\langle \Psi_i\left(x_i, n_{e_i}\right), \Sigma\left(x_i, \hat{\mu}_i^{(j)}(x_i)\right) \right\rangle_{\tilde{\Omega}_i} = 0. \end{aligned} \tag{57}$$

In order not have to calculate the $\mathcal{L}_2$ inner product in (57), we employ the Monte Carlo integration method [50]. Define the set $\{(x_i^\varsigma, n_{e_i}^\varsigma) | x_i^\varsigma \in \Omega_i, n_{e_i}^\varsigma \in \mathcal{E}_i, \varsigma = 1, 2, \ldots, z_i\}$, and z_i is the number of sample points. Let

$$\begin{aligned} \mathcal{X}_i &= \left[\Psi_i^{\mathsf{T}}\left(x_i^1, n_{e_i}^1\right), \ldots, \Psi_i^{\mathsf{T}}\left(x_i^{z_i}, n_{e_i}^{z_i}\right)\right]^{\mathsf{T}} \\ \mathcal{Y}_i &= \left[\Sigma\left(x_i^1, \hat{\mu}_i^{(j)}\left(x_i^1\right)\right), \ldots, \Sigma\left(x_i^{z_i}, \hat{\mu}_i^{(j)}\left(x_i^{z_i}\right)\right)\right]^{\mathsf{T}}. \end{aligned}$$

Then, letting $z_i \to \infty$, we get

$$\begin{aligned} &\left\langle \Psi_i\left(x_i, n_{e_i}\right), \Psi_i\left(x_i, n_{e_i}\right) \right\rangle_{\tilde{\Omega}_i} \\ &= \lim_{z_i \to \infty} \frac{\mathcal{I}\left(\tilde{\Omega}_i\right)}{z_i} \sum_{\varsigma=1}^{z_i} \Psi_i^{\mathsf{T}}\left(x_i^\varsigma, n_{e_i}^\varsigma\right) \Psi_i\left(x_i^\varsigma, n_{e_i}^\varsigma\right) \\ &= \lim_{z_i \to \infty} \frac{\mathcal{I}\left(\tilde{\Omega}_i\right)}{z_i} \mathcal{X}_i^{\mathsf{T}} \mathcal{X}_i \end{aligned} \tag{58}$$

with $\mathcal{I}(\tilde{\Omega}_i) = \int_{\tilde{\Omega}_i} \mathrm{d}(x_i, n_{e_i})$ the Lebesgue integral [49].

By the same token, we obtain

$$\begin{aligned} &\left\langle \Psi_i\left(x_i, n_{e_i}\right), \Sigma\left(x_i, \hat{\mu}_i^{(j)}(x_i)\right) \right\rangle_{\tilde{\Omega}_i} \\ &\qquad = \lim_{z_i \to \infty} \frac{\mathcal{I}\left(\tilde{\Omega}_i\right)}{z_i} \mathcal{X}_i^{\mathsf{T}} \mathcal{Y}_i. \end{aligned} \tag{59}$$

Using (57)–(59) and selecting sufficiently large z_i, we have

$$\left(\mathcal{X}_i^{\mathsf{T}} \mathcal{X}_i\right) \Phi_i^{(j)} + \mathcal{X}_i^{\mathsf{T}} \mathcal{Y}_i = 0. \tag{60}$$

If there exists the number of sampling points z_0 ($z_0 \geq \tilde{n}_1 + m_i \tilde{n}_2 + l_i \tilde{n}_3$) such that

$$\operatorname{rank}\{\mathcal{X}_i\} = \tilde{n}_1 + m_i \tilde{n}_2 + l_i \tilde{n}_3 \tag{61}$$

then (60) yields

$$\Phi_i^{(j)} = -\left(\mathcal{X}_i^{\mathsf{T}} \mathcal{X}_i\right)^{-1} \mathcal{X}_i^{\mathsf{T}} \mathcal{Y}_i. \tag{62}$$

When the sequence $\{\Phi_i^{(j)}\}$ generated from (62) is convergent, we can obtain the CNN and ANN weights simultaneously through (55). Then, by using (50), we can derive the approximate optimal control for the ith auxiliary subsystem.

Remark 5: To guarantee the validity of (61), one often can select large enough number of sampling points z_i. Moreover, $\Psi_i(x_i, n_{e_i})$ is unnecessary to be persistently exciting, for the sampling points are collected offline.

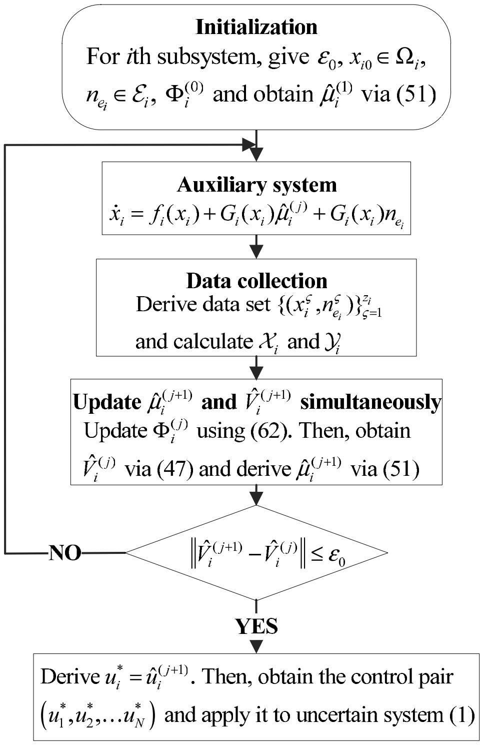

Initialization
For ith subsystem, give ε_0, $x_{i0} \in \Omega_i$, $n_{e_i} \in \mathcal{E}_i$, $\Phi_i^{(0)}$ and obtain $\hat{\mu}_i^{(1)}$ via (51)
Auxiliary system
$\dot{x}_i = f_i(x_i) + G_i(x_i)\hat{\mu}_i^{(j)} + G_i(x_i)n_{e_i}$
Data collection
Derive data set $\{(x_i^\varsigma, n_{e_i}^\varsigma)\}_{\varsigma=1}^{z_i}$ and calculate $\mathcal{X}_i$ and $\mathcal{Y}_i$
Update $\hat{\mu}_i^{(j+1)}$ and $\hat{V}_i^{(j+1)}$ simultaneously
Update $\Phi_i^{(j)}$ using (62). Then, obtain $\hat{V}_i^{(j)}$ via (47) and derive $\hat{\mu}_i^{(j+1)}$ via (51)
NO
$\left\|\hat{V}_i^{(j+1)} - \hat{V}_i^{(j)}\right\| \le \varepsilon_0$
YES
Derive $u_i^* = \hat{u}_i^{(j+1)}$. Then, obtain the control pair $(u_1^*, u_2^*, \ldots u_N^*)$ and apply it to uncertain system (1)

Fig. 1. Block diagram of the present control strategy.

The block diagram of the present control strategy is illustrated in Fig. 1 (note: ε_0 is a small positive computation accuracy, and $x_{i0} \in \Omega_i$ is the initial state of the ith subsystem, where $i = 1, 2, \ldots, N$).

VI. Simulation Results

This section presents two examples to illustrate the effectiveness and applicability of the developed control strategy. First, we consider a nonlinear plant consisting of two interconnected subsystems. Then, we study the power system proposed in [51], which includes three interconnected subsystems.

A. Example 1: Nonlinear Plant

Consider the nonlinear interconnected system given in the form

$$\begin{aligned} \dot{x}_1 &= \begin{bmatrix} -x_{11} + x_{12} \\ -0.5x_{11} - 0.5x_{12}\cos^2(x_{11}) \end{bmatrix} + \begin{bmatrix} 0 \\ \sin(x_{11}) \end{bmatrix} u_1 \\ &\quad + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \Big(\varepsilon_1 (x_{11} + x_{22}) \sin^2(\varepsilon_2 x_{12}) \cos(0.5x_{21})\Big) \\ \dot{x}_2 &= \begin{bmatrix} -x_{21} + 0.5x_{22} \\ -x_{21} - 0.5x_{22} + 0.5x_{21}x_{22}^2 \end{bmatrix} + \begin{bmatrix} 0 \\ x_{21} \end{bmatrix} u_2 \\ &\quad + \begin{bmatrix} 1 \\ 0 \end{bmatrix} \Big(\varepsilon_3 (x_{12} + x_{22}) \cos\left(\varepsilon_4 e^{x_{21}^2}\right)\Big) \end{aligned} \tag{63}$$

where $x_1 = [x_{11}, x_{12}]^{\mathsf{T}} \in \mathbb{R}^2$ and $x_2 = [x_{21}, x_{22}]^{\mathsf{T}} \in \mathbb{R}^2$ are the states of subsystems 1 and 2, respectively, u_1 and u_2 are the control inputs for subsystems 1 and 2, respectively, and

$\varepsilon_s \in \mathbb{R}$ ($s = 1, 2, 3, 4$) are unknown parameters. For simplicity, we choose ε_i ($i = 1, 2$) and $\varepsilon_{i'}$ ($i' = 3, 4$) randomly within the intervals $[-1, 1]$ and $[-0.5, 0.5]$, respectively. Observing the expressions of the interconnections given in (63), we let $\alpha_1(x_1) = \beta_1(x_1) = \|x_1\|$ and $\alpha_2(x_2) = \beta_2(x_2) = \|x_2\|$. To satisfy Assumption 2, the parameters are designed as follows: $b_{11} = 1$, $b_{12} = 1$, $b_{21} = 0.5$, and $b_{22} = 0.5$. Owing to $g_1(x_1) = [0, \sin(x_{11})]^{\mathsf{T}}$, $g_2(x_2) = [0, x_{21}]^{\mathsf{T}}$, and $k_i(x_i) = [1, 0]^{\mathsf{T}}$ ($i = 1, 2$), we derive that $\|g_i^{+}(x_i)k_i(x_i)\omega_i(x)\| = 0$ ($i = 1, 2$). Thus, we can choose parameters $c_{i\iota} = 0$ ($i, \iota = 1, 2$) to satisfy Assumption 3.

By using (9), the auxiliary subsystems 1 and 2 for (63) can be obtained. To derive the decentralized control of system (63), we first solve the optimal control problems of two isolated auxiliary subsystems. According to (11), we let $P_1(x_1) = \|x_1\|$ and $P_2(x_2) = \|x_2\|$. Meanwhile, we choose $\eta_1 = 3$ and $\eta_2 = 3$ to make the matrix $\mathcal{A}$ [see (29)] positive definite. Then, selecting $\epsilon_i = 0.25$ ($i = 1, 2$) (note: according to Theorem 1, we have $0 < \epsilon_i < 1/2$. Therefore, we can let $\epsilon_i = 0.25$), $Q_1(x_1) = \|x_1\|^2$, and $Q_2(x_2) = 2\|x_2\|^2$, we can propose the value functions for auxiliary subsystems 1 and 2 as

$$J_1(x_1, u_1, \upsilon_1) = \int_0^{\infty} \left(4\|x_1\|^2 + u_1^{\mathsf{T}} u_1 + 0.25\upsilon_1^{\mathsf{T}}\upsilon_1\right)\mathrm{d}t$$

$$J_2(x_2, u_2, \upsilon_2) = \int_0^{\infty} \left(5\|x_2\|^2 + u_2^{\mathsf{T}} u_2 + 0.25\upsilon_2^{\mathsf{T}}\upsilon_2\right)\mathrm{d}t.$$

For auxiliary subsystem 1, the vector activation functions for the CNN and ANNs are, respectively, selected as (note: $\tilde{n}_1 = 3$, $\tilde{n}_2 = 3$, and $\tilde{n}_3 = 3$)

$$\sigma_1(x_1) = \left[x_{11}^2, x_{12}^2, x_{11}x_{12}\right]^{\mathsf{T}}$$
$$\psi_1(x_1) = [x_{11}, x_{12}, x_{11}x_{12}]^{\mathsf{T}}$$
$$\phi_1(x_1) = [x_{11}, x_{12}, x_{11}x_{12}]^{\mathsf{T}}.$$

The associated CNN and ANN weight vectors are denoted as $\theta_1^{(j)} = [\theta_{11}^{(j)}, \theta_{12}^{(j)}, \theta_{13}^{(j)}]^{\mathsf{T}}$, $\lambda_1^{(j)} = [\lambda_{11}^{(j)}, \lambda_{12}^{(j)}, \lambda_{13}^{(j)}]^{\mathsf{T}}$, and $\nu_1^{(j)} = [\nu_{11}^{(j)}, \nu_{12}^{(j)}, \nu_{13}^{(j)}]^{\mathsf{T}}$, respectively. The initial weight vectors for CNN and ANNs are set as follows: $\theta_1^{(0)} = [0, 0, 0]^{\mathsf{T}}$, $\lambda_1^{(0)} = [-2, -2, -2]^{\mathsf{T}}$, and $\nu_1^{(0)} = [-1, -1, -1]^{\mathsf{T}}$.

For auxiliary subsystem 2, we choose the vector activation functions for the CNN and ANNs as (note: $\tilde{n}_1$, $\tilde{n}_2$, and $\tilde{n}_3$ are the same as in auxiliary subsystem 1)

$$\sigma_2(x_2) = \left[x_{21}^2, x_{22}^2, x_{21}x_{22}\right]^{\mathsf{T}}$$
$$\psi_2(x_2) = [x_{21}, x_{22}, x_{21}x_{22}]^{\mathsf{T}}$$
$$\phi_2(x_2) = [x_{21}, x_{22}, x_{21}x_{22}]^{\mathsf{T}}.$$

Meanwhile, we denote the associated CNN and ANN weight vectors as $\theta_2^{(j)} = [\theta_{21}^{(j)}, \theta_{22}^{(j)}, \theta_{23}^{(j)}]^{\mathsf{T}}$, $\lambda_2^{(j)} = [\lambda_{21}^{(j)}, \lambda_{22}^{(j)}, \lambda_{23}^{(j)}]^{\mathsf{T}}$, and $\nu_2^{(j)} = [\nu_{21}^{(j)}, \nu_{22}^{(j)}, \nu_{23}^{(j)}]^{\mathsf{T}}$, respectively. The initial weight vectors for CNN and ANNs are given as follows: $\theta_2^{(0)} = [0, 0, 0]^{\mathsf{T}}$, $\lambda_2^{(0)} = [-2, -2, -2]^{\mathsf{T}}$, and $\nu_2^{(0)} = [-2, -2, -2]^{\mathsf{T}}$. Moreover, we set the initial state $x_0 = [1, -1, 1, 0.5]^{\mathsf{T}}$ and the sampling period $\Delta t = 0.01$ s. The compact sets Ω_i, $i = 1, 2$, are both chosen to be the interval $[-1, 1]$, that is, $\Omega_i = [-1, 1]$, $i = 1, 2$.

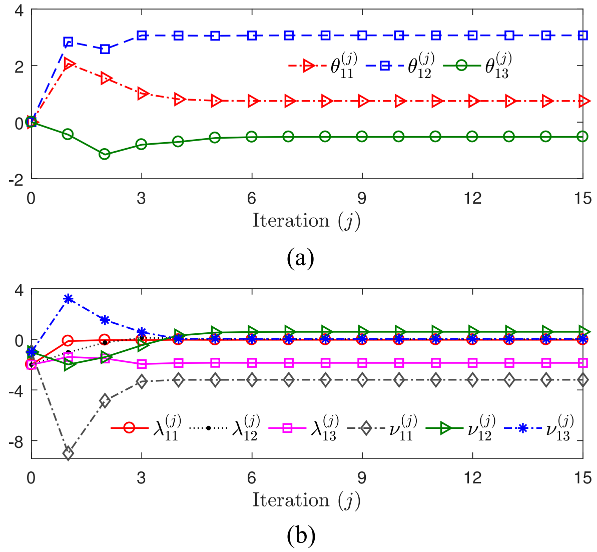


Fig. 2. (a) Performance of the CNN weight vector $\theta_1^{(j)}$. (b) Performance of ANN weight vectors $\lambda_1^{(j)}$ and $\nu_1^{(j)}$.

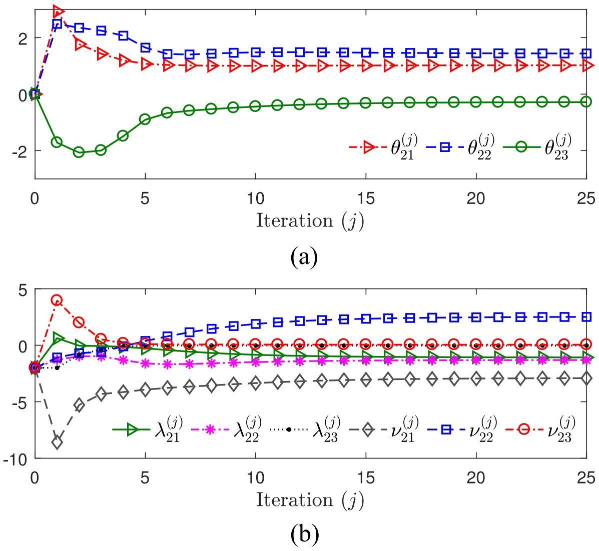


Fig. 3. (a) Performance of the CNN weight vector $\theta_2^{(j)}$. (b) Performance of ANN weight vectors $\lambda_2^{(j)}$ and $\nu_2^{(j)}$.

Remark 6: To our knowledge, the selection of the proper number of neurons for neural networks remains an open question. To address this issue, here we choose the number of neurons through computer simulations. After choosing three neurons for the CNN and ANNs, respectively, we can obtain desirable simulation results. In addition, it should be mentioned that the number of neurons used in the following Example 2 is also determined by computer simulations.

The computer simulation results are depicted in Figs. 2–5. Figs. 2 and 3 describe the performance of CNN and ANN weight vectors used in solving optimal control problems of auxiliary subsystems 1 and 2, respectively. From Fig. 2(a) and (b), we can observe that the weight vectors $\theta_1^{(j)}$, $\lambda_1^{(j)}$, and $\nu_1^{(j)}$ are all convergent after twelve iterations. The converged value of the sequence $\{\lambda_1^{(j)}\}$

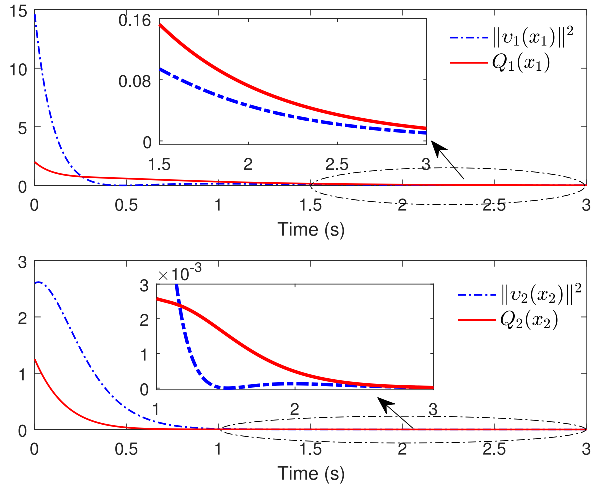


Fig. 4. Verification of condition (19) for isolated subsystems.

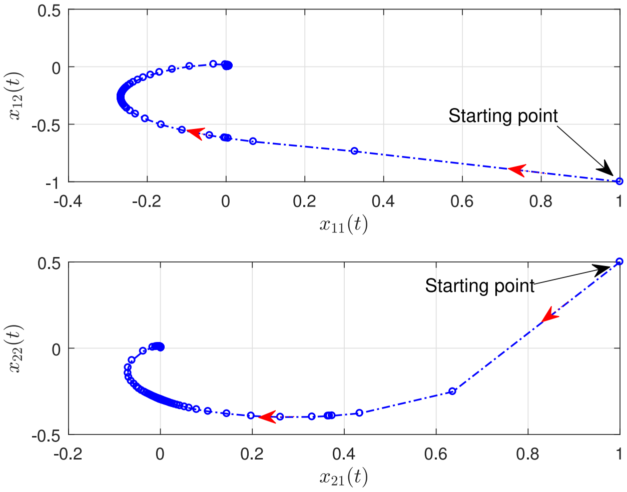


Fig. 5. States of system (63) under the obtained control pair (u_1, u_2).

is $\lambda_1^{(12)} = [-0.0349, 0.0092, -1.8599]^\mathsf{T}$. Fig. 3(a) and (b) indicate that the weight vectors $\theta_2^{(j)}$, $\lambda_2^{(j)}$, and $\nu_2^{(j)}$ are all convergent after twenty iterations. The converged value of the sequence $\{\lambda_2^{(j)}\}$ is $\lambda_2^{(20)} = [-1.0817, -1.3033, -0.0280]^\mathsf{T}$. Thus, substituting $\lambda_1^{(12)}$ and $\lambda_2^{(20)}$ into (50), we can obtain the control pair (u_1, u_2). Fig. 4 is provided to validate the condition (19) for isolated subsystems. As indicated in Fig. 4, the condition (19) holds when $t \geq \max\{0.3, 1.2\} = 1.2$ s (i.e., $t_0 = 1.2$ s). Fig. 5 shows the states of system (63) under the obtained control pair (u_1, u_2). As shown in Fig. 5, system (63) is asymptotically stable.

B. Example 2: Application to Power Systems

Consider the large-scale power systems described by [51]

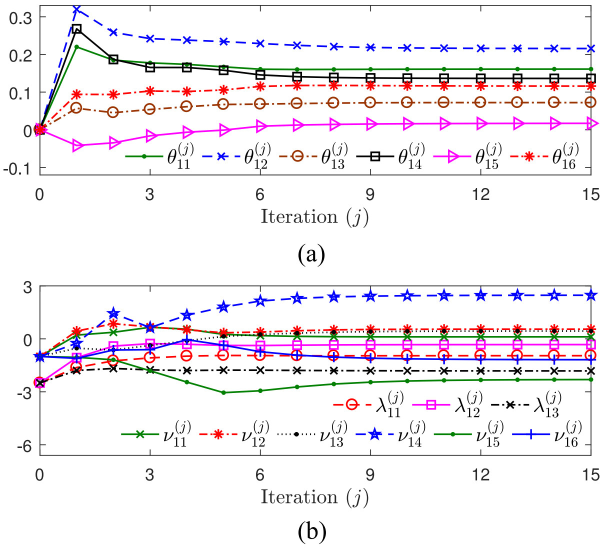
$$\frac{\mathrm{d}(\Delta\vartheta_i(t))}{\mathrm{d}t} = -\frac{1}{T_{g_i}}\Delta\vartheta_i(t) + \frac{1}{R_{g_i}T_{g_i}}\Delta f_{G_i}(t) + \frac{1}{T_{g_i}}u_i(t)$$

$$\frac{\mathrm{d}\big(\Delta P_{\tilde{m}_i}(t)\big)}{\mathrm{d}t} = \frac{K_{t_i}}{T_{t_i}}\Delta\vartheta_i(t) - \frac{1}{T_{t_i}}\Delta P_{\tilde{m}_i}(t)$$

$$\frac{\mathrm{d}\big(\Delta f_{G_i}(t)\big)}{\mathrm{d}t} = \frac{K_{p_i}}{T_{p_i}}\Delta P_{\tilde{m}_i}(t) - \frac{\Delta f_{G_i}(t)}{T_{p_i}} - \frac{K_{p_i}}{T_{p_i}}\Delta P_{G_i}(t) \tag{64}$$

TABLE I
PARAMETERS FOR THE POWER SYSTEM

Parameter	Meaning	Value 1	Value 2	Value 3
K_{p_i} (Hz/MW)	generator model gain	120	120	120
K_{t_i} (s)	turbine model gain	1	1	1
R_{g_i} (Hz/MW)	feedback regulation constant	2.5	2.6	2.7
T_{g_i} (s)	governor time constant	0.08	0.1	0.2
T_{p_i} (s)	generator model time constant	20	20	20
T_{t_i} (s)	turbine time constant	0.1	0.2	0.3



(a)

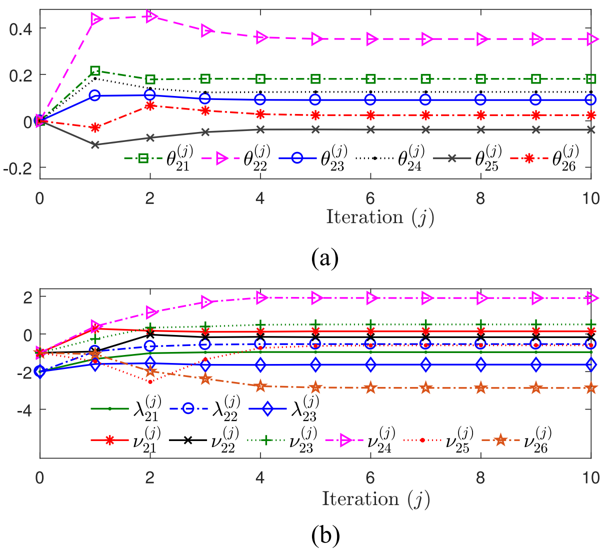
(b)

Fig. 6. (a) Performance of the CNN weight vector $\theta_1^{(j)}$. (b) Performance of ANN weight vectors $\lambda_1^{(j)}$ and $\nu_1^{(j)}$.

where $\mathrm{d}(F_i(t))/\mathrm{d}t$ denotes the time derivative of $F_i(t)$ (note: $F_i(t) = \Delta\vartheta_i(t)$, $\Delta P_{\tilde{m}_i}(t)$, or $\Delta f_{G_i}(t)$), $i = 1, 2, \ldots, N$, $\Delta\vartheta_i(t) \in \mathbb{R}$ is the incremental change in governor value position, $\Delta P_{\tilde{m}_i}(t) \in \mathbb{R}$ is the incremental change in generator output, $\Delta f_{G_i}(t) \in \mathbb{R}$ is the incremental frequency deviation, $u_i \in \mathbb{R}$ is the control input, $\Delta P_{G_i}(t) \in \mathbb{R}$ is the incremental change in the electrical power, and the rest are constant parameters.

A three-machine power system (i.e., $N = 3$) is studied in this simulation. The parameters are displayed in Table I. We assume that $\Delta P_{G_i}(t) = \Upsilon_i(t)\sin(\Delta P_{\tilde{m}_i}(t)\Delta f_{G_i}(t))$, $i = 1, 2, 3$, where $\Upsilon_1(t) = \sum_{i=1}^{3}\varepsilon_{1i}\Delta\vartheta_i(t)$, $\Upsilon_2(t) = \sum_{i=1}^{3}\varepsilon_{2i}\Delta P_{\tilde{m}_i}(t)$, $\Upsilon_3(t) = \sum_{i=1}^{3}\varepsilon_{3i}\Delta f_{G_i}(t)$, and ε_{is}, $i, s = 1, 2, 3$, are unknown parameters. For simplicity, we randomly choose $\varepsilon_{11} \in [-1, 1]$, $\varepsilon_{1\iota} \in [-0.5, 0.5]$, $\varepsilon_{\iota 1} \in [-0.5, 0.5]$ $(\iota = 2, 3)$, and $\varepsilon_{\iota\jmath} \in [-0.25, 0.25]$ $(\iota, \jmath = 2, 3)$.

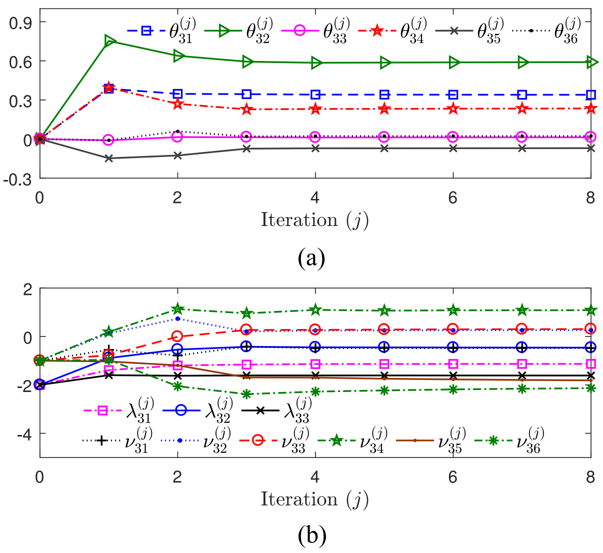
Let $x_i = [\Delta\vartheta_i(t), \Delta P_{\tilde{m}_i}(t), \Delta f_{G_i}(t)]^\mathsf{T} = [x_{i1}, x_{i2}, x_{i3}]^\mathsf{T}$, $i = 1, 2, 3$. Then, based on aforementioned characteristic of $\Delta P_{G_i}(t)$, we choose $\alpha_i(x_i) = \beta_i(x_i) = \|x_i\|$, $i = 1, 2, 3$. Note that $g_i(x_i) = [1/T_{g_i}, 0, 0]^\mathsf{T}$ and $k_i(x_i) = [0, 0, K_{p_i}/T_{p_i}]^\mathsf{T}$ $(i = 1, 2, 3)$. Thus, to satisfy Assumptions 2 and 3, we can design the parameters as follows: $b_{11} = 1$, $b_{1\iota} = 0.5$ $(\iota = 2, 3)$, $b_{21} = 0.5$, $b_{2\iota} = 0.25$ $(\iota = 2, 3)$, $b_{31} = 0.5$, $b_{3\iota} = 0.25$ $(\iota = 2, 3)$, and $c_{is} = 0$ $(i, s = 1, 2, 3)$.



(a)

(b)

Fig. 7. (a) Performance of the CNN weight vector $\theta_2^{(j)}$. (b) Performance of ANN weight vectors $\lambda_2^{(j)}$ and $\nu_2^{(j)}$.



(a)

(b)

Fig. 8. (a) Performance of the CNN weight vector $\theta_3^{(j)}$. (b) Performance of ANN weight vectors $\lambda_3^{(j)}$ and $\nu_3^{(j)}$.

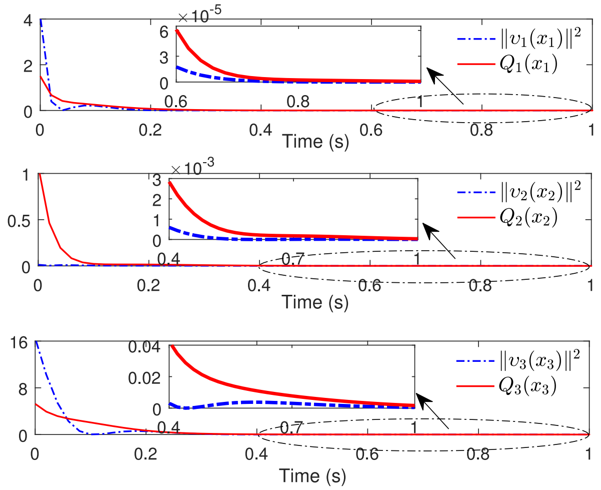


Fig. 9. Verification of condition (19) for isolated subsystems.

By using (9), the auxiliary subsystems 1–3 for (64) can be derived. To obtain the decentralized controller for system (64), we first solve the optimal control problems of three isolated auxiliary subsystems. According to (11), we select $P_i(x_i) = \|x_i\|$, $i = 1, 2, 3$. Meanwhile, we choose $\eta_i = 3$ $(i = 1, 2, 3)$ to keep the matrix $\mathcal{A}$ [see (29)] positive definite. Then, letting $\epsilon_i = 0.25$ and $Q_i(x_i) = \|x_i\|^2$, $i = 1, 2, 3$, we present the value functions for auxiliary subsystems 1–3 as

$$J_i(x_i, u_i, \upsilon_i) = \int_0^\infty \left(4\|x_i\|^2 + u_i^{\mathsf T}u_i + 0.25\upsilon_i^{\mathsf T}\upsilon_i\right)\mathrm{d}t$$

with $i = 1, 2, 3$. For each auxiliary subsystem, the vector activation functions for the CNN and ANNs are, respectively, given as (note: $\tilde{n}_1 = 6$, $\tilde{n}_2 = 3$, and $\tilde{n}_3 = 6$)

$$\begin{aligned}\sigma_i(x_i) &= \left[x_{i1}^2, x_{i2}^2, x_{i3}^2, x_{i1}x_{i2}, x_{i1}x_{i3}, x_{i2}x_{i3}\right]^{\mathsf T}\\ \psi_i(x_i) &= [x_{i1}, x_{i2}, x_{i3}]^{\mathsf T}\\ \phi_i(x_i) &= [x_{i1}, x_{i2}, x_{i3}, x_{i1}x_{i2}, x_{i1}x_{i3}, x_{i2}x_{i3}]^{\mathsf T}.\end{aligned}$$

The associated CNN and ANN weight vectors are written as $\theta_i^{(j)} = [\theta_{i1}^{(j)}, \theta_{i2}^{(j)}, \ldots, \theta_{i6}^{(j)}]^{\mathsf T}$, $\lambda_i^{(j)} = [\lambda_{i1}^{(j)}, \lambda_{i2}^{(j)}, \lambda_{i3}^{(j)}]^{\mathsf T}$, and $\nu_i^{(j)} = [\nu_{i1}^{(j)}, \nu_{i2}^{(j)}, \ldots, \nu_{i6}^{(j)}]^{\mathsf T}$, respectively. The initial weight vectors for CNN and ANNs are given as follows: $\theta_i^{(0)} = [0, 0, \ldots, 0]^{\mathsf T}$ $(i = 1, 2, 3)$, $\lambda_1^{(0)} = [-2.5, -2.5, -2.5]^{\mathsf T}$, $\lambda_2^{(0)} = \lambda_3^{(0)} = [-2, -2, -2]^{\mathsf T}$, and $\nu_i^{(0)} = [-1, -1, \ldots, -1]^{\mathsf T}$ $(i = 1, 2, 3)$. Moreover, the initial state is $x_0 = [1, -0.5, 0.5, 1, -0.2, 0.2, 2, -1, 0.5]^{\mathsf T}$ and the sampling period is $\Delta t = 0.02$ s. The compact sets Ω_i, $i = 1, 2, 3$, are all chosen to be the interval $[-2, 2]$, that is, $\Omega_i = [-2, 2]$, $i = 1, 2, 3$.

The computer simulation results are displayed in Figs. 6–10. Figs. 6–8 depict the performance of CNN and ANN weight vectors used in solving optimal control problems of auxiliary subsystems 1–3, respectively. Fig. 6(a) and (b) show that the weight vectors $\theta_1^{(j)}$, $\lambda_1^{(j)}$, and $\nu_1^{(j)}$ are all convergent after twelve iterations. The converged value of the sequence $\{\lambda_1^{(j)}\}$ is $\lambda_1^{(12)} = [-0.9547, -0.3251, -1.8187]^{\mathsf T}$. Fig. 7(a) and (b) describe that the weight vectors $\theta_2^{(j)}$, $\lambda_2^{(j)}$, and $\nu_2^{(j)}$ are all convergent after eight iterations. The converged value of the sequence $\{\lambda_2^{(j)}\}$ is $\lambda_2^{(8)} = [-0.9681, -0.5386, -1.6280]^{\mathsf T}$. Fig. 8(a) and (b) indicate that the weight vectors $\theta_3^{(j)}$, $\lambda_3^{(j)}$, and $\nu_3^{(j)}$ are all convergent after six iterations. The converged value of the sequence $\{\lambda_3^{(j)}\}$ is $\lambda_3^{(6)} = [-1.1288, -0.4696, -1.6144]^{\mathsf T}$. Then, substituting $\lambda_1^{(12)}$, $\lambda_2^{(8)}$, and $\lambda_3^{(6)}$ into (50), we derive the control pair (u_1, u_2, u_3). Fig. 9 is provided to validate the condition (19) for isolated subsystems. As illustrated in Fig. 9, the condition (19) holds when $t \geq \max\{0.05, 0, 0.1\} = 0.1$ s (i.e., $t_0 = 0.1$ s). Fig. 10 presents the evolution of $\Delta\vartheta_i(t)$, $\Delta P_{\tilde{m}_i}(t)$, and $\Delta f_{G_i}(t)$ $(i = 1, 2, 3)$ under the obtained control pair (u_1, u_2, u_3). As shown in Fig. 10, system (64) is asymptotically stable.

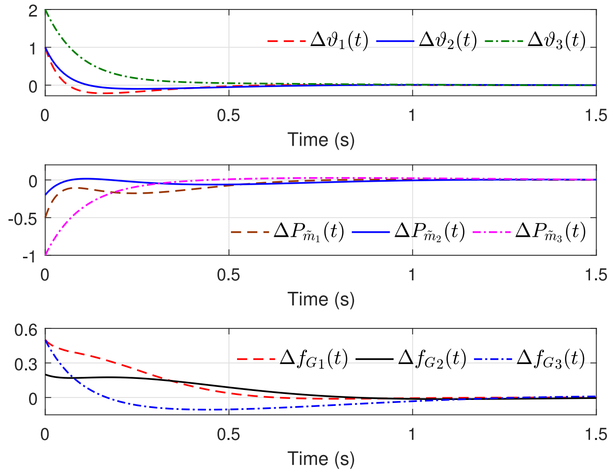


Fig. 10. Evolution of $\Delta\vartheta_i(t)$, $\Delta P_{\tilde{m}_i}(t)$, and $\Delta f_{G_i}(t)$ $(i = 1, 2, 3)$ under the obtained control pair (u_1, u_2, u_3).

VII. Conclusion

We have presented a novel decentralized control scheme for uncertain nonlinear large-scale systems with mismatched interconnections. Specifically speaking, we first partition the given decentralized control problem into optimal control problems of auxiliary subsystems. Then, the SPI algorithm is developed to solve these optimal control problems within the framework of ADP. When developing the decentralized control scheme, we have to calculate the Moore–Penrose pseudo-inverse of the control matrix beforehand. This is mainly because the pseudo-inverse of the control matrix is a part of the value function for each subsystem. This requirement is a limitation of the present method. In our consecutive work, we will focus on removing this condition. On the other hand, it is observed that system (1) is composed of input-affine nonlinear subsystems. In general, the design of controllers for input-nonaffine nonlinear systems is more intractable than for input-affine nonlinear systems [52], [53]. Therefore, how to extend the present decentralized control strategy to input-nonaffine nonlinear interconnected systems is also one direction of our future works.

References

[1] L. Bakule, “Decentralized control: An overview,” *Annu. Rev. Control*, vol. 32, no. 1, pp. 87–98, Apr. 2008.

[2] A. Saberi, “On optimality of decentralized control for a class of nonlinear interconnected systems,” *Automatica*, vol. 24, no. 1, pp. 101–104, Jan. 1988.

[3] Y.-X. Li and G.-H. Yang, “Adaptive fuzzy decentralized control for a class of large-scale nonlinear systems with actuator faults and unknown dead zones,” *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 5, pp. 729–740, May 2017.

[4] Y. Yang and D. Yue, “Observer-based decentralized adaptive NNs fault-tolerant control of a class of large-scale uncertain nonlinear systems with actuator failures,” *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2017.2744676.

[5] Y. Li and S. Tong, “Adaptive neural networks prescribed performance control design for switched interconnected uncertain nonlinear systems,” *IEEE Trans. Neural Netw. Learn. Syst.*, to be published, doi: 10.1109/TNNLS.2017.2712698.

[6] W. B. Powell, *Approximate Dynamic Programming: Solving the Curses of Dimensionality*, 2nd ed. Hoboken, NJ, USA: Wiley, 2007.

[7] P. J. Werbos, “Beyond regression: New tools for prediction and analysis in the behavioral sciences,” Ph.D. dissertation, Dept. Appl. Math., Harvard Univ., Cambridge, MA, USA, 1974.

[8] H. Zhang, L. Cui, X. Zhang, and Y. Luo, “Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method,” *IEEE Trans. Neural Netw.*, vol. 22, no. 12, pp. 2226–2236, Dec. 2011.

[9] D. Vrabie, K. G. Vamvoudakis, and F. L. Lewis, *Optimal Adaptive Control and Differential Games by Reinforcement Learning Principles*. London, U.K.: Inst. Eng. Technol., 2013.

[10] H. Zhang, H. Jiang, C. Luo, and G. Xiao, “Discrete-time nonzero-sum games for multiplayer using policy-iteration-based adaptive dynamic programming algorithms,” *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3331–3340, Oct. 2017.

[11] Q. Wei, D. Liu, Q. Lin, and R. Song, “Discrete-time optimal control via local policy iteration adaptive dynamic programming,” *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3367–3379, Oct. 2017.

[12] Q. Wei, D. Liu, and H. Lin, “Value iteration adaptive dynamic programming for optimal control of discrete-time nonlinear systems,” *IEEE Trans. Cybern.*, vol. 46, no. 3, pp. 840–853, Mar. 2016.

[13] Q. Wei, F. L. Lewis, D. Liu, R. Song, and H. Lin, “Discrete-time local value iteration adaptive dynamic programming: Convergence analysis,” *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 6, pp. 875–891, Jun. 2018, doi: 10.1109/TSMC.2016.2623766.

[14] D. P. Bertsekas, “Value and policy iterations in optimal control and adaptive dynamic programming,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 500–509, Mar. 2017.

[15] Y. Jiang and Z.-P. Jiang, *Robust Adaptive Dynamic Programming*. Hoboken, NJ, USA: Wiley, 2017.

[16] Y. Zhu, D. Zhao, and X. Li, “Iterative adaptive dynamic programming for solving unknown nonlinear zero-sum game based on online data,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 714–725, Mar. 2017.

[17] Z. Ni, H. He, D. Zhao, X. Xu, and D. V. Prokhorov, “GrDHP: A general utility function representation for dual heuristic dynamic programming,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 3, pp. 614–627, Mar. 2015.

[18] X. Zhong, Z. Ni, and H. He, “Gr-GDHP: A new architecture for globalized dual heuristic dynamic programming,” *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3318–3330, Oct. 2017.

[19] H. Zhang, L. Cui, and Y. Luo, “Near-optimal control for nonzero-sum differential games of continuous-time nonlinear systems using single-network ADP,” *IEEE Trans. Cybern.*, vol. 43, no. 1, pp. 206–216, Feb. 2013.

[20] A. Heydari and S. N. Balakrishnan, “Finite-horizon control-constrained nonlinear optimal control using single network adaptive critics,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 1, pp. 145–157, Jan. 2013.

[21] J. Y. Lee, J. B. Park, and Y. H. Choi, “Integral reinforcement learning for continuous-time input-affine nonlinear systems with simultaneous invariant explorations,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 5, pp. 916–932, May 2015.

[22] B. Kiumarsi, K. G. Vamvoudakis, H. Modares, and F. L. Lewis, “Optimal and autonomous control using reinforcement learning: A survey,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 6, pp. 2042–2062, Jun. 2018, doi: 10.1109/TNNLS.2017.2773458.

[23] R. Song, F. L. Lewis, Q. Wei, and H. Zhang, “Off-policy actor-critic structure for optimal control of unknown systems with disturbances,” *IEEE Trans. Cybern.*, vol. 46, no. 5, pp. 1041–1050, May 2016.

[24] R. Song, F. L. Lewis, and Q. Wei, “Off-policy integral reinforcement learning method to solve nonlinear continuous-time multiplayer nonzero-sum games,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 3, pp. 704–713, Mar. 2017.

[25] B. Luo, H.-N. Wu, T. Huang, and D. Liu, “Data-based approximate policy iteration for affine nonlinear continuous-time optimal control design,” *Automatica*, vol. 50, no. 12, pp. 3281–3290, Dec. 2014.

[26] D. Zhao and Y. Zhu, “MEC—A near-optimal online reinforcement learning algorithm for continuous deterministic systems,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 2, pp. 346–356, Feb. 2015.

[27] V. Narayanan and S. Jagannathan, “Event-triggered distributed control of nonlinear interconnected systems using online reinforcement learning with exploration,” *IEEE Trans. Cybern.*, to be published, doi: 10.1109/TCYB.2017.2741342.

[28] Q. Wei, F. L. Lewis, Q. Sun, P. Yan, and R. Song, “Discrete-time deterministic *Q*-learning: A novel convergence analysis,” *IEEE Trans. Cybern.*, vol. 47, no. 5, pp. 1224–1237, May 2017.

[29] V. Narayanan and S. Jagannathan, "Distributed adaptive optimal regulation of uncertain large-scale interconnected systems using hybrid *Q*-learning approach," *IET Control Theory Appl.*, vol. 10, no. 12, pp. 1448–1457, Aug. 2016.

[30] W. Guo *et al.*, "Online supplementary ADP learning controller design and application to power system frequency control with large-scale wind energy integration," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 8, pp. 1748–1761, Aug. 2016.

[31] D. Liu, Q. Wei, D. Wang, X. Yang, and H. Li, *Adaptive Dynamic Programming With Applications in Optimal Control*. Cham, Switzerland: Springer, 2017.

[32] C. Mu, C. Sun, D. Wang, A. Song, and C. Qian, "Decentralized adaptive optimal stabilization of nonlinear systems with matched interconnections," *Soft Comput.*, vol. 22, no. 8, pp. 2705–2715, Apr. 2018.

[33] Q. Qu, H. Zhang, T. Feng, and H. Jiang, "Decentralized adaptive tracking control scheme for nonlinear large-scale interconnected systems via adaptive dynamic programming," *Neurocomputing*, vol. 225, pp. 1–10, Feb. 2017.

[34] D. Wang, H. He, B. Zhao, and D. Liu, "Adaptive near-optimal controllers for non-linear decentralised feedback stabilisation problems," *IET Control Theory Appl.*, vol. 11, no. 6, pp. 799–806, Apr. 2017.

[35] B. Zhao, D. Wang, G. Shi, D. Liu, and Y. Li, "Decentralized control for large-scale nonlinear systems with unknown mismatched interconnections via policy iteration," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2017.2690665.

[36] S. Tong, K. Sun, and S. Sui, "Observer-based adaptive fuzzy decentralized optimal control design for strict-feedback nonlinear large-scale systems," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 2, pp. 569–584, Apr. 2018.

[37] T. Bian, Y. Jiang, and Z.-P. Jiang, "Decentralized adaptive optimal control of large-scale systems with application to power systems," *IEEE Trans. Ind. Electron.*, vol. 62, no. 4, pp. 2439–2447, Apr. 2015.

[38] D. Liu, C. Li, H. Li, D. Wang, and H. Ma, "Neural-network-based decentralized control of continuous-time nonlinear interconnected systems with unknown dynamics," *Neurocomputing*, vol. 165, pp. 90–98, Oct. 2015.

[39] F. Lin, *Robust Control Design: An Optimal Control Approach*. Chichester, U.K.: Wiley, 2007.

[40] H. K. Khalil, *Nonlinear System*. Hoboken, NJ, USA: Prentice-Hall, 2002.

[41] W. Rudin, *Principles of Mathematical Analysis*, 3rd ed. New York, NY, USA: McGraw-Hill, 1976.

[42] P. A. Ioannou and J. Sun, *Robust Adaptive Control*. Upper Saddle River, NJ, USA: Prentice-Hall, 1996.

[43] X. Yang and H. He, "Self-learning robust optimal control for continuous-time nonlinear systems with mismatched disturbances," *Neural Netw.*, vol. 99, pp. 19–30, Mar. 2018.

[44] X. Yang, H. He, and X. Zhong, "Adaptive dynamic programming for robust regulation and its application to power systems," *IEEE Trans. Ind. Electron.*, vol. 65, no. 7, pp. 5722–5732, Jul. 2018.

[45] G. N. Saridis and C.-S. G. Lee, "An approximation theory of optimal control for trainable manipulators," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. SMC-9, no. 3, pp. 152–179, Mar. 1979.

[46] X. Yang, D. Liu, B. Luo, and C. Li, "Data-based robust adaptive control for a class of unknown nonlinear constrained-input systems via integral reinforcement learning," *Inf. Sci.*, vol. 369, pp. 731–747, Nov. 2016.

[47] K. Hornik, M. Stinchcombe, and H. White, "Universal approximation of an unknown mapping and its derivatives using multilayer feedforward networks," *Neural Netw.*, vol. 3, no. 5, pp. 551–560, 1990.

[48] M. Abu-Khalaf and F. L. Lewis, "Nearly optimal control laws for nonlinear systems with saturating actuators using a neural network HJB approach," *Automatica*, vol. 41, no. 5, pp. 779–791, May 2005.

[49] W. Rudin, *Functional Analysis*. New York, NY, USA: McGraw-Hill, 1991.

[50] M. Evans and T. Swartz, *Approximating Integrals via Monte Carlo and Deterministic Methods*. New York, NY, USA: Oxford Univ. Press, 2000.

[51] D. P. Iracleous and A. T. Alexandridis, "A multi-task automatic generation control for power regulation," *Elect. Power Syst. Res.*, vol. 73, no. 3, pp. 275–285, Mar. 2005.

[52] X. Yang, D. Liu, D. Wang, and Q. Wei, "Discrete-time online learning control for a class of unknown nonaffine nonlinear systems using reinforcement learning," *Neural Netw.*, vol. 55, pp. 30–41, Jul. 2014.

[53] K. Esfandiari, F. Abdollahi, and H. A. Talebi, "Adaptive near-optimal neuro controller for continuous-time nonaffine nonlinear systems with constrained input," *Neural Netw.*, vol. 93, pp. 195–204, Sep. 2017.

Xiong Yang received the B.S. degree in mathematics and applied mathematics from Central China Normal University, Wuhan, China, in 2008, the M.S. degree in pure mathematics from Shandong University, Jinan, China, in 2011, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2014.

From 2014 to 2016, he was an Assistant Professor with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences. Since 2016, he has been a Post-Doctoral Fellow with the Department of Electrical, Computer, and Biomedical Engineering, University of Rhode Island, Kingston, RI, USA. He is currently an Associate Professor with the School of Electrical and Information Engineering, Tianjin University, Tianjin, China. His current research interests include adaptive dynamic programming, reinforcement learning, event-triggered control, data-driven control, and their applications.

Dr. Yang was a recipient of the Excellent Award of Presidential Scholarship of the Chinese Academy of Sciences in 2014.

Haibo He (SM'11–F'18) received the B.S. and M.S. degrees in electrical engineering from the Huazhong University of Science and Technology, Wuhan, China, in 1999 and 2002, respectively, and the Ph.D. degree in electrical engineering from Ohio University, Athens, OH, USA, in 2006.

He is currently the Robert Haas Endowed Chair Professor with the Department of Electrical, Computer and Biomedical Engineering, University of Rhode Island, Kingston, RI, USA. He has published one sole-author research book entitled *Self-Adaptive Systems for Machine Intelligence* (Wiley, 2011), edited one book entitled *Imbalanced Learning: Foundations, Algorithms, and Applications* (Wiley-IEEE, 2013), and six conference proceedings (Springer), and authored and co-authored over 300 peer-reviewed journal and conference papers.

Dr. He was a recipient of the IEEE International Conference on Communications Best Paper Award in 2014, the IEEE CIS Outstanding Early Career Award in 2014, the National Science Foundation CAREER Award in 2011, and the Providence Business News "Rising Star Innovator" Award in 2011. He was the Chair of IEEE Computational Intelligence Society Emergent Technologies Technical Committee in 2015 and the Chair of IEEE CIS Neural Networks Technical Committee in 2013 and 2014. He was the General Chair of the IEEE Symposium Series on Computational Intelligence in 2014. He is currently the Editor-in-Chief of the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS.