Data-Based Optimal Control for Weakly Coupled Nonlinear Systems Using Policy Iteration

Chao Li, Derong Liu, *Fellow, IEEE*, and Ding Wang, *Member, IEEE*

***Abstract*—In this paper, a data-based online learning algorithm is established to solve the optimal control problem for weakly coupled continuous-time nonlinear systems with completely unknown dynamics. Using the weak coupling theory, we reformulate the original problem into three reduced-order optimal control problems. We establish an online model-free integral policy iteration algorithm to solve the decoupled optimal control problems without system dynamics. To implement the data-based online learning algorithm, the actor-critic technique based on neural networks and the least squares method are used. Two simulation examples are given to verify the effectiveness of the developed algorithm.**

***Index Terms*—Adaptive dynamic programming (ADP), neural networks (NNs), optimal control, policy iteration (PI), unknown dynamics, weakly coupled systems.**

I. Introduction

IN THE real world, many large-scale systems are naturally weakly coupled, such as electrical networks, transportation systems, chemical reactors, and power systems. For these real physical systems, a traditional challenge is the optimal control problem. A common approach is to split this large-scale optimal control problem into some decoupled subproblems using the decentralized control method [1], [2]. While the coupling effects are usually neglected and the obtained control laws may do not have ideal performance. In 1969, Kokotović *et al.* [3] introduced the weakly coupled linear systems to the control systems community. Since then, many theoretical aspects of the optimal control problem for weakly coupled systems have been studied. Gajić and Shen [4], [5] obtained the optimal control law through a decoupling transformation which leads to solving two independent reduced-order optimal control problems. For weakly coupled bilinear systems, the optimal control problem has also been solved in a similar way [6], [7]. Jiang and Jiang [8] presented a new approach to decouple the weakly coupled large-scale linear systems and accomplished the stability analysis using the small-gain theory. The optimal control law of the nonlinear systems can be obtained by solving the Hamilton–Jacobi–Bellman (HJB) equations. However, due to the intractable form of the HJB equations, obtaining closed-form optimal controllers by directly solving the HJB equations is difficult. By using the reduced-order scheme and the successive Galerkin approximation (SGA), the optimal control law for the weakly coupled nonlinear system has been constructed based on the solutions of two independent reduced-order HJB equations [9]. Carrillo *et al.* [10] proposed a learning algorithm to derive the optimal control law using a three-critics/four-actors approximator structure with system dynamics. For large-scale real physical systems, it is difficult to obtain the exact knowledge of system dynamics. Therefore, a kind of data-based algorithms is needed to solve the optimal control problem with unknown system dynamics.

Dynamic programming provides a principled method for determining optimal control laws for dynamical systems in the case of completely known dynamics. While due to the "curse of dimensionality" [11], it is often computationally untenable to obtain the optimal control laws. Among the methods of solving optimal control problems, adaptive dynamic programming (ADP), and reinforcement learning (RL) relax the need for a complete and exact model of the dynamical systems by using compact parameterized approximators. ADP has received increasing attention due to its learning capabilities [12]–[30]. RL is an effective computational method and it can find the optimal policy interactively [31]–[34]. In the existing literature of ADP-based and RL-based optimal control, either policy iteration (PI) or value iteration is utilized to solve the HJB equation. Vrabie and Lewis [35] established an integral RL algorithm to obtain direct adaptive optimal control for nonlinear continuous-time systems with partial system dynamics. Liu *et al.* [36] developed an online synchronous approximate optimal learning algorithm based on PI to solve a multiplayer nonzero-sum game without the exact knowledge of dynamical systems. Jiang and Jiang [37] presented a novel PI method to solve optimal control problems for linear systems with completely unknown dynamics. Jiang and Jiang [38] presented a novel method of global ADP for the adaptive

Manuscript received January 4, 2016; revised June 29, 2016; accepted August 21, 2016. This work was supported in part by the National Natural Science Foundation of China under Grant 61233001, Grant 61273140, Grant 61304086, Grant 61533017, and Grant U1501251, in part by the Beijing Natural Science Foundation under Grant 4162065, in part by the Early Career Development Award of SKLMCCS, and in part by the Research Fund of Tianjin Key Laboratory of Process Measurement and Control under Grant TKLPMC-201612. This paper was recommended by Associate Editor Z. Liu.

C. Li is with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (e-mail: lichao2012@ia.ac.cn).

D. Liu is with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China (e-mail: derong@ustb.edu.cn).

D. Wang is with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the Tianjin Key Laboratory of Process Measurement and Control, School of Electrical Engineering and Automation, Tianjin University, Tianjin 300072, China (e-mail: ding.wang@ia.ac.cn).

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.2016.2606479

optimal control of nonlinear polynomial systems to achieve global asymptotic stability. Without the exact knowledge of system dynamics, Lee *et al.* [39] derived a model-free integral Q-learning approach for nonlinear system.

Although ADP-based and RL-based algorithms are widely used, there are few related results which can be used to tackle the optimal control problem for the weakly coupled nonlinear systems. The novelty of this paper is that we establish a data-based learning algorithm to solve this problem with completely unknown dynamics. By partitioning the HJB equation, the original optimal control problem of the weakly coupled systems is reformulated into three reduced-order optimal control problems. We establish the model-free integral PI algorithm to solve the decoupled optimal control problems without system dynamics. The actor-critic technique based on neural networks (NNs) and the least squares method are used to implement the derived online learning algorithm.

The rest of this paper is organized as follows. In Section II, the optimal control problem for the weakly coupled nonlinear systems is described. In Section III, the original problem is reformulated into three reduced-order optimal control problems and a model-free integral PI algorithm using online learning manner with unknown system dynamics is established. Two simulation examples are provided to demonstrate the applicability of the established optimal control policy in Section IV. In Section V, we conclude this paper with a few remarks.

II. Problem Formulation

In this paper, we consider the continuous-time nonlinear system with weakly coupled structure

$$\begin{bmatrix}\dot{x}_1(t)\\ \dot{x}_2(t)\end{bmatrix} = \begin{bmatrix} f_{11}(x_1)+\varepsilon f_{12}(x)\\ \varepsilon f_{21}(x)+f_{22}(x_2)\end{bmatrix} + \begin{bmatrix} g_{11}(x_1) & \varepsilon g_{12}(x)\\ \varepsilon g_{21}(x) & g_{22}(x_2)\end{bmatrix}\begin{bmatrix} u_{11}(t)+\varepsilon u_{12}(t)\\ \varepsilon u_{21}(t)+u_{22}(t)\end{bmatrix} \tag{1}$$

where $x_1(t)\in\mathbb{R}^{n_1}$ and $x_2(t)\in\mathbb{R}^{n_2}$ are the system state vectors, $u_{11}(t), u_{12}(t)\in\mathbb{R}^{m_1}$ and $u_{21}(t), u_{22}(t)\in\mathbb{R}^{m_2}$ are the control input vectors, n_1, n_2, m_1, and m_2 are positive integers, and ε is a small positive weak coupling parameter. Using the following expressions:

$$x(t)=\begin{bmatrix}x_1(t)\\ x_2(t)\end{bmatrix}, f(x)=\begin{bmatrix} f_{11}(x_1)+\varepsilon f_{12}(x)\\ \varepsilon f_{21}(x)+f_{22}(x_2)\end{bmatrix}$$
$$g(x)=\begin{bmatrix} g_{11}(x_1) & \varepsilon g_{12}(x)\\ \varepsilon g_{21}(x) & g_{22}(x_2)\end{bmatrix}, u(t)=\begin{bmatrix} u_{11}(t)+\varepsilon u_{12}(t)\\ \varepsilon u_{21}(t)+u_{22}(t)\end{bmatrix}$$

the system dynamics (1) can be rewritten as

$$\dot{x}(t)=f(x)+g(x)u(t). \tag{2}$$

We assume that the system (2) is controllable, $f\colon\mathbb{R}^n\to\mathbb{R}^n$ and $g\colon\mathbb{R}^n\to\mathbb{R}^{n\times m}$ are Lipschitz continuous on the set $\Omega\subseteq\mathbb{R}^n$, where $n=n_1+n_2$, $m=m_1+m_2$, and there must exist a continuous control policy which asymptotically stabilizes the system. Additionally, we let the following assumptions hold through out this paper.

Assumption 1: The state vector $x=0$ is the equilibrium of the system.

Assumption 2: The functions $f(\cdot)$ and $g(\cdot)$ are differentiable in their arguments, and $f(0)=0$.

Assumption 3: The feedback control vector $u(x) = 0$ when $x=0$.

According to the optimal control theory, we known that solving the optimal control problem is equal to find the optimal control policy $u^*(x(t))$ which minimizes the expenditure of control effort. For this, we define the value function as

$$V(x(t))=\int_t^{\infty}\Big[x^{\mathsf{T}}(\tau)Qx(\tau)+u^{\mathsf{T}}(\tau)Ru(\tau)\Big]\mathrm{d}\tau \tag{3}$$

where $Q\in\mathbb{R}^{n\times n}$ and $R\in\mathbb{R}^{m\times m}$ are positive definite symmetric matrices, and $r(x,u)=x^{\mathsf{T}}(t)Qx(t)+u^{\mathsf{T}}(t)Ru(t)$ is the utility function. The matrices Q and R have the following weakly coupled structures:

$$Q=\begin{bmatrix} Q_1 & \varepsilon Q_\varepsilon\\ \varepsilon Q_\varepsilon^{\mathsf{T}} & Q_2\end{bmatrix}, \qquad R=\begin{bmatrix} R_1 & 0\\ 0 & R_2\end{bmatrix}$$

where $Q_1\in\mathbb{R}^{n_1\times n_1}$, $Q_2\in\mathbb{R}^{n_2\times n_2}$, $R_1\in\mathbb{R}^{m_1\times m_1}$, and $R_2\in\mathbb{R}^{m_2\times m_2}$ are positive definite symmetric matrices, and $Q_\varepsilon\in\mathbb{R}^{n_1\times n_2}$. We know that the designed feedback control policy $u(x(t))$ must not only stabilize the system on Ω, but also guarantee that the value function (3) is finite. That is to say, the control policy must be admissible.

Definition 1: A control policy $u(x)$ is said to be admissible with respect to (3) on Ω, denoted by $u(x)\in\Psi(\Omega)$ [$\Psi(\Omega)$ is the set of all admissible control laws], if $u(x)$ is continuous on Ω, $u(0)=0$, $u(x)$ stabilizes the system (2) on Ω, and $V(x(t))$ is finite $\forall x_0\in\Omega$, where x_0 is the initial system state [40].

According to the optimal control theory, the optimal value function is defined as

$$V^*(x(t))=\min_{u\in\Psi(\Omega)}\int_t^{\infty}\Big[x^{\mathsf{T}}(\tau)Qx(\tau)+u^{\mathsf{T}}(\tau)Ru(\tau)\Big]\mathrm{d}\tau.$$

We define the Hamiltonian function of system (2) as

$$H(x,u,V_x)=V_x^{\mathsf{T}}\big[f(x)+g(x)u\big]+r(x,u) \tag{4}$$

with $V(0)=0$, and the term $V_x=\partial V(x)/\partial x$ denotes the partial derivative of the value function with respect to the state. We minimize the Hamiltonian function (4) to obtain the optimal control policy

$$u^*(x)=\arg\min_{u\in\Psi(\Omega)}H(x,u,V_x)=-\frac{1}{2}R^{-1}g^{\mathsf{T}}(x)V_x^*. \tag{5}$$

Using the optimal control policy $u^*(x)$, the optimal value function $V^*(x)$ can be described as the unique positive-definite solution of the following HJB equation:

$$0=V_x^{*\mathsf{T}}\big[f(x)+g(x)u^*(x)\big]+r\big(x,u^*(x)\big). \tag{6}$$

Remark 1: In the class of nonlinear systems, the optimal control scheme is based on the solution of the HJB equation (6). Because the solution of HJB equation for nonlinear systems can hardly be found, the SGA method [41], [42] is developed. However, the SGA method has the weakness that the complexity of computation increases rapidly with the order of the system, where the order indicates the dimension of a system, i.e., n. Kim and Lim [9] established the optimal control from two independent reduced-order HJB equations using the

SGA method. Due to the difficulties when obtaining the exact knowledge of the system dynamics, model-free methods are more practical. Motivated by the weak coupling theory and [9], we establish a novel model-free algorithm to derive the optimal control based on three reduced-order HJB equations.

III. Computational Controller Design Using Data-Based Online Learning Algorithm

By partitioning the HJB equation, the original problem of the weakly coupled systems is reformulated into three reduced-order optimal control problems. We established the model-free integral PI algorithm to solve the decoupled optimal control problems without system dynamics. The actor-critic technique based on NNs and the least squares method are used to implement the derived online learning algorithm.

A. Problem Transformation

The value function (3) can be partitioned as

$$V(x(t)) = V_1(x_1(t)) + V_2(x_2(t)) + \varepsilon V_\varepsilon(x(t))$$

where

$$\begin{aligned} V_1(x_1(t)) &= \int_t^\infty \left[x_1^\mathsf{T} Q_1 x_1 + u_{11}^\mathsf{T} R_1 u_{11}\right] \mathrm{d}\tau \\ V_2(x_2(t)) &= \int_t^\infty \left[x_2^\mathsf{T} Q_2 x_2 + u_{22}^\mathsf{T} R_2 u_{22}\right] \mathrm{d}\tau \\ V_\varepsilon(x(t)) &= 2\int_t^\infty \left[x_1^\mathsf{T} Q_\varepsilon x_2 + u_{11}^\mathsf{T} R_1 u_{12} + u_{22}^\mathsf{T} R_2 u_{21}\right] \mathrm{d}\tau \\ &\quad + \varepsilon \int_t^\infty \left[u_{12}^\mathsf{T} R_1 u_{12} + u_{21}^\mathsf{T} R_2 u_{21}\right] \mathrm{d}\tau. \end{aligned}$$

According to the reduced-order scheme [9], setting $\varepsilon^2 = 0$, $\varepsilon V_\varepsilon(x(t))$ can be represented as

$$\varepsilon V_\varepsilon(x(t)) = 2\varepsilon \int_t^\infty \left[x_1^\mathsf{T} Q_\varepsilon x_2 + u_{11}^\mathsf{T} R_1 u_{12} + u_{22}^\mathsf{T} R_2 u_{21}\right] \mathrm{d}\tau.$$

We give the following definitions to denote the partial derivatives of the value functions $V_1(x_1(t))$, $V_2(x_2(t))$, and $V_\varepsilon(x(t))$ with respect to the states x_1 and x_2, respectively:

$$\begin{aligned} V_{1x_1} &= \frac{\partial V_1}{\partial x_1}, \quad V_{2x_2} = \frac{\partial V_2}{\partial x_2} \\ V_{\varepsilon x_1} &= \frac{\partial V_\varepsilon}{\partial x_1}, \quad V_{\varepsilon x_2} = \frac{\partial V_\varepsilon}{\partial x_2}. \end{aligned}$$

Theorem 1: Partitioning the HJB equation (6), we get an $\mathcal{O}(\varepsilon^2)$ approximation in terms of three reduced-order decoupled HJB equations

$$\begin{aligned} 0 &= V_{1x_1}^{*\mathsf{T}}\left[f_{11}(x_1) + g_{11}(x_1)u_{11}^*(x_1)\right] \\ &\quad + x_1^\mathsf{T} Q_1 x_1 + u_{11}^{*\mathsf{T}}(x_1) R_1 u_{11}^*(x_1) \\ 0 &= V_{2x_2}^{*\mathsf{T}}\left[f_{22}(x_2) + g_{22}(x_2)u_{22}^*(x_2)\right] \\ &\quad + x_2^\mathsf{T} Q_2 x_2 + u_{22}^{*\mathsf{T}}(x_2) R_2 u_{22}^*(x_2) \\ 0 &= V_{1x_1}^{*\mathsf{T}} f_{12}(x) + V_{2x_2}^{*\mathsf{T}} f_{21}(x) + V_{\varepsilon x_1}^{*\mathsf{T}} f_{11}(x_1) \\ &\quad + V_{\varepsilon x_2}^{*\mathsf{T}} f_{22}(x_2) + 2x_1^\mathsf{T} Q_\varepsilon x_2 \\ &\quad - 2u_{11}^{*\mathsf{T}}(x_1) R_1 u_{12}^*(x) - 2u_{22}^{*\mathsf{T}}(x_2) R_2 u_{21}^*(x). \end{aligned}$$

The optimal control law (5) can be partitioned as

$$\begin{aligned} u_{11}^*(x_1) &= -\frac{1}{2} R_1^{-1} g_{11}^\mathsf{T}(x_1) V_{1x_1}^* \\ u_{12}^*(x) &= -\frac{1}{2} R_1^{-1}\left[g_{11}^\mathsf{T}(x_1) V_{\varepsilon x_1}^* + g_{21}^\mathsf{T}(x) V_{2x_2}^*\right] \\ u_{21}^*(x) &= -\frac{1}{2} R_2^{-1}\left[g_{22}^\mathsf{T}(x_2) V_{\varepsilon x_2}^* + g_{12}^\mathsf{T}(x) V_{1x_1}^*\right] \\ u_{22}^*(x_2) &= -\frac{1}{2} R_2^{-1} g_{22}^\mathsf{T}(x_2) V_{2x_2}^*. \end{aligned} \tag{7}$$

Based on the optimal control theory, $u_{11}^*(x_1)$ can be seen as the optimal control law for the subsystem 1

$$\dot{x}_1(t) = f_{11}(x_1) + g_{11}(x_1)u_{11}(t)$$

with respect to the value function $V_1(x_1)$. $u_{22}^*(x_2)$ can be seen as the optimal control law for the subsystem 2

$$\dot{x}_2(t) = f_{22}(x_2) + g_{22}(x_2)u_{22}(t)$$

with respect to the value function $V_2(x_2)$. $u_{12}^*(x)$ and $u_{21}^*(x)$ can be solved from the optimal control problem of the virtual subsystem 3 with respect to the value function $V_3^*(x)$

$$\begin{aligned} V_3^*(x) = 2\int_t^\infty &\Big[u_{11}^{*\mathsf{T}}(x_1) R_1 u_{12}^*(x) + u_{22}^{*\mathsf{T}}(x_2) R_2 u_{21}^*(x) \\ &- x_1^\mathsf{T} Q_\varepsilon x_2\Big] \mathrm{d}\tau \end{aligned}$$

where $V_3^*(x) = V_\varepsilon^*(x) - 4\int_t^\infty x_1^\mathsf{T} Q_\varepsilon x_2 \mathrm{d}\tau$ with $V_3^*(0) = 0$.

Proof: Refer to the Appendix. ■

Remark 2: The original optimal control problem with the HJB equation (6) is transformed into three reduced-order HJB equations which should be solved without system dynamics. In the following section, we will derive the data-based online learning algorithm.

B. Model-Free Integral PI Algorithm

The optimal control formulation developed in (7) displays an array of closed-form expressions, which obviates the need to search for the optimal control law via optimization process. To obtain the optimal control law, the existence of $V^*(x)$ satisfying the HJB equation (6) is the necessary and sufficient condition. Instead of directly solving (6), we can successively solve the nonlinear Lyapunov equation (4) and update the control policy based on (7) to obtain the solution $V^*(x)$. This successive approximation is known as the model-based PI algorithm [42]–[45], and it is fundamental for the model-free integral PI algorithm and we describe it as follows.

1) Model-Based PI Algorithm: Step 1: Give a small positive real number ϵ. Let $i = 0$ and start with an initial admissible control policy $u^0(x)$.

Step 2 (Policy Evaluation): Based on the control policy $u^i(x)$, solve $V^i(x)$ from the following nonlinear Lyapunov equation:

$$r\big(x, u^i(x)\big) + V_x^{i\mathsf{T}}\left[f(x) + g(x)u^i(x)\right] = 0.$$

Step 3 (Policy Improvement): Update the control policy by

$$u^{i+1}(x) = -\frac{1}{2} R^{-1} g^\mathsf{T}(x) V_x^i. \tag{8}$$

Step 4: If $\|u^{i+1}(x) - u^i(x)\| \leq \epsilon$, stop and obtain the approximate optimal control policy $u^{i+1}(x)$; else, set $i = i + 1$ and go to step 2.

In [41], it was shown that on the domain Ω, the cost function $V^i(x)$ uniformly converges to $V^*(x)$ with monotonicity $V^{i+1}(x) \leq V^i(x)$, and the control policy $u^i(x)$ is admissible and converges to $u^*(x)$.

To deal with the optimal control problem without system dynamics, we develop a data-based online learning algorithm called model-free integral PI algorithm. Consider a nonlinear system which is explored by a known bounded piecewise continuous probing signal $e(t)$

$$\dot{x}(t) = f(x) + g(x)[u(t) + e(t)]$$

where

$$u(t) + e(t) = \begin{bmatrix} [u_{11}(t) + e_1(t)] + \varepsilon u_{12}(t) \\ \varepsilon u_{21}(t) + [u_{22}(t) + e_2(t)] \end{bmatrix}.$$

Now, we consider the subsystem 1 with exploration signal

$$\dot{x}_1(t) = f_{11}(x_1) + g_{11}(x_1)[u_{11}(t) + e_1(t)]. \tag{9}$$

The derivative of the value function $V_1(x_1(t))$ with respect to time along the trajectory of the explored system (9) can be calculated as

$$\begin{aligned} \dot{V}_1(x_1(t)) &= V_{1x_1}^{\mathsf{T}}\big[f_{11}(x_1) + g_{11}(x_1)[u_{11}(t) + e_1(t)]\big] \\ &= -r_1(x_1, u_{11}(x_1)) + V_{1x_1}^{\mathsf{T}} g_{11}(x_1) e_1(t) \end{aligned} \tag{10}$$

where $r_1(x_1, u_{11}(x_1)) = x_1^{\mathsf{T}} Q_1 x_1 + u_{11}^{\mathsf{T}}(x_1) R_1 u_{11}(x_1)$ is the utility function for the subsystem 1 given in (9).

We present a lemma which is essential to prove the convergence of the model-free integral PI algorithm.

Lemma 1: Solving for $V_1(x_1)$ in the following equation:

$$\begin{aligned} &V_1(x_1(t+T)) - V_1(x_1(t)) \\ &\quad = \int_t^{t+T} V_{1x_1}^{\mathsf{T}} g_{11}(x_1) e_1(\tau) \mathrm{d}\tau - \int_t^{t+T} r_1(x_1, u_{11}(x_1)) \mathrm{d}\tau \end{aligned} \tag{11}$$

is equivalent to finding the solution of (10).

Proof: Since $u_{11}(x_1) \in \Psi_1(\Omega_1)$ [$\Psi_1(\Omega_1)$ is the set of all admissible control laws for the subsystem 1], the value function $V_1(x_1)$ is a Lyapunov function for the subsystem 1, and it satisfies (10) with $r_1(x_1, u_{11}(x_1)) > 0$, $x_1 \neq 0$. We integrate (10) over the interval $[t, t+T]$ to obtain (11). This means that the unique solution of (10), $V_1(x_1)$, also satisfies (11). To complete the proof, we show that (11) has a unique solution by contradiction.

We assume that there exists another value function $\bar{V}_1(x_1)$ which satisfies (11) with bounding condition $\bar{V}_1(0) = 0$. This value function also satisfies $\dot{\bar{V}}_1(x_1) = -r_1(x_1, u_{11}(x_1)) + \bar{V}_{1x_1}^{\mathsf{T}} g_{11}(x_1) e_1(t)$. Subtracting this from (10), we obtain

$$\begin{aligned} 0 &= \left(\frac{\mathrm{d}\big[\bar{V}_1(x_1) - V_1(x_1)\big]^{\mathsf{T}}}{\mathrm{d}x_1}\right) \times \big[\dot{x}_1(t) - g_{11}(x_1) e_1(t)\big] \\ &= \left(\frac{\mathrm{d}\big[\bar{V}_1(x_1) - V_1(x_1)\big]^{\mathsf{T}}}{\mathrm{d}x_1}\right) \times \big[f_{11}(x_1) + g_{11}(x_1) u_{11}(x_1)\big] \end{aligned}$$

which must hold for any x_1 on the system trajectories generated by the stabilizing policy $u_{11}(x_1)$. According to the above equation, we have $\bar{V}_1(x_1) = V_1(x_1) + c$. As this relation must hold for $x_1(t) = 0$, we know $\bar{V}_1(0) = V_1(0) + c$, $c = 0$. Thus, $\bar{V}_1(x_1) = V_1(x_1)$, i.e., (11) has a unique solution which is equal to the solution of (10). The proof is complete. ■

Algorithm 1 Model-Free Integral PI Algorithm

```
1: Give a small positive real number $\epsilon$. Let $i = 0$ and start
   with an initial admissible control policy $u_{11}^0(x_1)$.
2: **Policy Evaluation and Improvement**: Based on the control
   policy $u_{11}^i(x_1)$, solve $V_1^i(x_1)$ and $u_{11}^{i+1}(x_1)$ from the
   integral equation (13).
3: If $\|u_{11}^{i+1}(x_1) - u_{11}^i(x_1)\| \leq \epsilon$, stop and obtain the approximate
   optimal control policy $u_{11}^{i+1}$ for the subsystem 1; else,
   set $i = i + 1$ and go to Step 2.
```

Based on the model-based PI algorithm and using the representations $V_1^i(x_1(t))$ and $u_{11}^i(x_1)$, the policy improvement (8) for the subsystem 1 can be written as

$$u_{11}^{i+1}(x_1) = -\frac{1}{2} R_1^{-1} g_{11}^{\mathsf{T}}(x_1) V_{1x_1}^i \tag{12}$$

where i is the iteration index. Integrating (10) from t to $t + T$ with a time period $T > 0$, and using the policy improvement (12), we have

$$\begin{aligned} V_1^i(x_1(t)) - V_1^i(x_1(t+T)) = &\int_t^{t+T} r_1\big(x_1, u_{11}^i(x_1)\big) \mathrm{d}\tau \\ &+ 2\int_t^{t+T} \Big(u_{11}^{i+1}(x_1)\Big)^{\mathsf{T}} R_1 e_1(\tau) \mathrm{d}\tau. \end{aligned} \tag{13}$$

Since the dynamics $f_{11}(x_1)$ and $g_{11}(x_1)$ are not in the integral equation (13), the integral PI algorithm can be implemented using the data generated from the system instead of the system dynamics. Thus, we obtain the model-free integral PI algorithm (Algorithm 1).

Theorem 2: Give an initial admissible control policy $u_{11}^0(x_1)$ for the subsystem 1. Using the model-free integral PI algorithm established in Algorithm 1, the value function and the control law converge to the optimal value function and the optimal control law as $i \to \infty$, that is

$$V_1^i(x_1) \to V_1^*(x_1), \quad u_{11}^i(x_1) \to u_{11}^*(x_1).$$

Proof: Based on the results in [35], we known that all the subsequent control policies will be admissible during the algorithm implementation if $u_{11}^0(x_1)$ is admissible. Considering the model-based PI algorithm and the formation process of (13), the value function sequence generated in Algorithm 1 will converge to the solution of the HJB equation. So we can conclude that the value function $V_1^i(x_1)$ and the control policy $u_{11}^i(x_1)$ obtained from the proposed model-free integral PI algorithm will converge to the solution of the optimal control problem for the subsystem 1. The proof is complete. ■

To solve the optimal control policy $u_{22}^*(x_2)$ for the subsystem 2, we can apply Algorithm 1 with some simply replacements. Using the expressions of the optimal control laws $u_{11}^*(x_1)$ and $u_{22}^*(x_2)$, we derive the following equation which will be used to solve $u_{12}^*(x)$ and $u_{21}^*(x)$:

$$\begin{aligned}V_3^i(x(t+T)) - V_3^i(x(t)) &= -2\int_t^{t+T} x_1^{\mathsf{T}} Q_\varepsilon x_2 \mathrm{d}\tau\\ &+ 2\int_t^{t+T}\Big[u_{11}^{*\mathsf{T}}(x_1)R_1u_{12}^i(x) + u_{22}^{*\mathsf{T}}(x_2)R_2u_{21}^i(x)\Big]\mathrm{d}\tau\\ &+ 2\int_t^{t+T}\bigg[\Big(u_{12}^{i+1}(x)\Big)^{\mathsf{T}}R_1e_1(\tau) + \Big(u_{21}^{i+1}(x)\Big)^{\mathsf{T}}R_2e_2(\tau)\bigg]\mathrm{d}\tau.\end{aligned}$$

Using this formulation to replace the integral equation (13) in Algorithm 1, we can calculate $u_{12}^*(x)$ and $u_{21}^*(x)$ iteratively.

C. Algorithm Implementation

For the subsystem 1, we represent $V_1^i(x_1)$ and $u_{11}^{i+1}(x_1)$ by single-layer NNs on a compact set Ω_1 as

$$\begin{aligned}V_1^i(x_1) &= \sum_{j=1}^{N_{1c}} \omega_{1j}^i\phi_{1j}(x_1) + \delta_{1c}^i(x_1)\\ u_{11,p}^{i+1}(x_1) &= \sum_{j=1}^{N_{1a}} \nu_{1j,p}^i\psi_{1j}(x_1) + \delta_{1a,p}^i(x_1)\end{aligned}$$

where $p = 1, 2, \ldots, m_1$, $\omega_{1j}^i \in \mathbb{R}$, and $\nu_{1j,p}^i \in \mathbb{R}$ are bounded ideal weight parameters which will be determined by the developed data-based integral PI algorithm, $\phi_{1j}(x_1) \in \mathbb{R}$ and $\psi_{1j}(x_1) \in \mathbb{R}$, $\{\phi_{1j}\}_{j=1}^{N_{1c}}$ and $\{\psi_{1j}\}_{j=1}^{N_{1a}}$ are the sequences of real-valued activation functions which are linearly complete and independent, and $\delta_{1c}^i(x_1) \in \mathbb{R}$ and $\delta_{1a,p}^i(x_1) \in \mathbb{R}$ are the bounded NN approximation errors. Since the ideal weights are unknown, the outputs of the critic network and the action network are

$$\hat{V}_1^i(x_1) = \sum_{j=1}^{N_{1c}} \hat{\omega}_{1j}^i\phi_{1j}(x_1) = \hat{\omega}_1^{i\mathsf{T}}\phi_1(x_1) \tag{14}$$

$$\hat{u}_{11,p}^{i+1}(x_1) = \sum_{j=1}^{N_{1a}} \hat{\nu}_{1j,p}^i\psi_{1j}(x_1) = \hat{\nu}_{1,p}^{i\mathsf{T}}\psi_1(x_1) \tag{15}$$

where $\hat{\omega}_1^i$ and $\hat{\nu}_{1,p}^i$ are the current estimated weights, and

$$\begin{aligned}\phi_1(x_1) &= \big[\phi_{11}(x_1), \phi_{12}(x_1), \ldots, \phi_{1N_{1c}}(x_1)\big]^{\mathsf{T}} \in \mathbb{R}^{N_{1c}}\\ \psi_1(x_1) &= \big[\psi_{11}(x_1), \psi_{12}(x_1), \ldots, \psi_{1N_{1a}}(x_1)\big]^{\mathsf{T}} \in \mathbb{R}^{N_{1a}}\\ \hat{\omega}_1^i &= \Big[\hat{\omega}_{11}^i, \hat{\omega}_{12}^i, \ldots, \hat{\omega}_{1N_{1c}}^i\Big]^{\mathsf{T}} \in \mathbb{R}^{N_{1c}}\\ \hat{\nu}_{1,p}^i &= \Big[\hat{\nu}_{11,p}^i, \hat{\nu}_{12,p}^i, \ldots, \hat{\nu}_{1N_{1a},p}^i\Big]^{\mathsf{T}} \in \mathbb{R}^{N_{1a}}\\ \hat{\nu}_1^{i\mathsf{T}} &= \Big[\hat{\nu}_{1,1}^i, \hat{\nu}_{1,2}^i, \ldots, \hat{\nu}_{1,m_1}^i\Big]^{\mathsf{T}} \in \mathbb{R}^{m_1\times N_{1a}}.\end{aligned}$$

Define $\mathrm{col}\{\hat{\nu}_1^{i\mathsf{T}}\} = [\hat{\nu}_{1,1}^{i\mathsf{T}}, \hat{\nu}_{1,2}^{i\mathsf{T}}, \ldots, \hat{\nu}_{1,m_1}^{i\mathsf{T}}]^{\mathsf{T}} \in \mathbb{R}^{m_1N_{1a}}$. Then

$$\begin{aligned}\Big(\hat{u}_{11}^{i+1}(x_1)\Big)^{\mathsf{T}}R_1e_1(t) &= \Big(\hat{\nu}_1^{i\mathsf{T}}\psi_1(x_1)\Big)^{\mathsf{T}}R_1e_1(t)\\ &= [\psi_1(x_1)\otimes(R_1e_1(t))]^{\mathsf{T}}\mathrm{col}\Big\{\hat{\nu}_1^{i\mathsf{T}}\Big\}\end{aligned}$$

where $\otimes$ represents the Kronecker product. Using the real outputs of the networks (14) and (15), the integral equation (13) has the following general form:

$$\lambda_{1k}^{\mathsf{T}}\begin{bmatrix}\hat{\omega}_1^i\\ \mathrm{col}\Big\{\hat{\nu}_1^{i\mathsf{T}}\Big\}\end{bmatrix} = \theta_{1k} \tag{16}$$

with

$$\begin{aligned}\theta_{1k} &= \int_{t+(k-1)T}^{t+kT}\Big[x_1^{\mathsf{T}}Q_1x_1 + \hat{u}_{11}^{i\mathsf{T}}(x_1)R_1\hat{u}_{11}^i(x_1)\Big]\mathrm{d}\tau\\ \lambda_{1k} &= \Bigg[(\phi_1(x_1(t+(k-1)T)) - \phi_1(x_1(t+kT)))^{\mathsf{T}}\\ &\qquad - 2\int_{t+(k-1)T}^{t+kT}(\psi_1(x_1)\otimes(R_1e_1(\tau)))^{\mathsf{T}}\mathrm{d}\tau\Bigg]^{\mathsf{T}}\end{aligned}$$

where T is the period of time to measure the data. Since the general form (16) is a 1-D equation, we cannot find the unique weight vector. The least squares method [39] can be used to guarantee the uniqueness of the weights over the compact set Ω_1. For any positive integer K_1, we denote $\Lambda_1 = [\lambda_{11}, \lambda_{12}, \ldots, \lambda_{1K_1}]$ and $\Theta_1 = [\theta_{11}, \theta_{12}, \ldots, \theta_{1K_1}]^{\mathsf{T}}$. Then, we have the following K_1-dimensional equation

$$\Lambda_1^{\mathsf{T}}\begin{bmatrix}\hat{\omega}_1^i\\ \mathrm{col}\Big\{\hat{\nu}_1^{i\mathsf{T}}\Big\}\end{bmatrix} = \Theta_1.$$

The weight vector can be solved by the following equation when Λ_1^{T} has full column rank:

$$\begin{bmatrix}\hat{\omega}_1^i\\ \mathrm{col}\Big\{\hat{\nu}_1^{i\mathsf{T}}\Big\}\end{bmatrix} = \Big(\Lambda_1\Lambda_1^{\mathsf{T}}\Big)^{-1}\Lambda_1\Theta_1. \tag{17}$$

Therefore, we need to make sure $(\Lambda_1\Lambda_1^{\mathsf{T}})^{-1}$ exists; that is to say, the number of collected points K_1 should satisfy $K_1 \geq \mathrm{rank}(\Lambda_1) = N_{1c} + m_1N_{1a}$. By collecting enough data points of the explored system (9), the weight parameters in (17) can be obtained in real time. Using the same implementation procedures for the subsystem 1, we can solve the optimal control problems of the subsystems 2 and 3.

IV. Numerical Simulation

We provide two simulation examples in this section to demonstrate the applicability of the established data-based integral PI algorithm for weakly coupled nonlinear systems.

Example 1: In this example, we consider the system (1) with the following parameters:

$$\begin{aligned}f_{11}(x_1) &= \begin{bmatrix}-1.93x_{11}^2\\ -1.394x_{11}x_{12}\end{bmatrix}\\ f_{12}(x) &= \begin{bmatrix}0\\ -4.26x_{21}x_{22}\end{bmatrix}\\ f_{21}(x) &= \begin{bmatrix}-1.3x_{12}^2\\ 0.95x_{11}x_{21} - 1.03x_{12}x_{22}\end{bmatrix}\\ f_{22}(x_2) &= \begin{bmatrix}-0.63x_{21}^2\\ 0.413x_{21} - 0.426x_{22}\end{bmatrix}\\ g_{11}(x_1) &= \begin{bmatrix}-1.274x_{11}^2\\ 0\end{bmatrix}, \qquad g_{12}(x) = \begin{bmatrix}0\\ -6.5x_{22}\end{bmatrix}\\ g_{21}(x) &= \begin{bmatrix}0.75x_{11}\\ 0\end{bmatrix}, \qquad g_{22}(x_2) = \begin{bmatrix}-0.718x_{21}\\ 0\end{bmatrix}.\end{aligned}$$

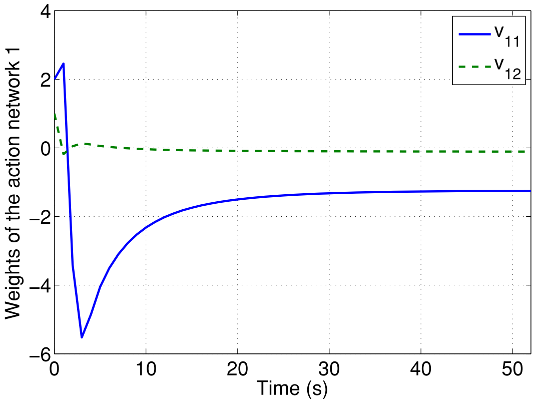


Fig. 1. Evolution of the action network 1's weights.

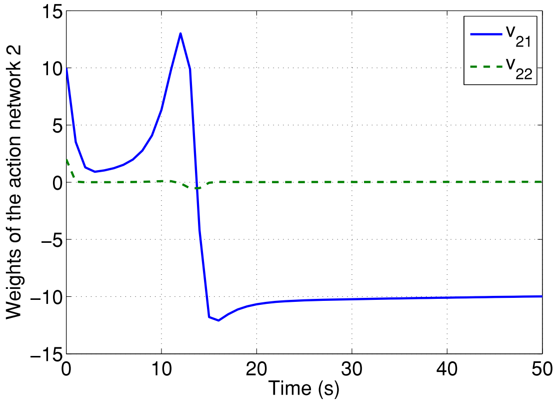


Fig. 2. Evolution of the action network 2's weights.

In the above, $x_1 = [x_{11}, x_{12}]^{\mathsf{T}} \in \mathbb{R}^2$ and $u_{11}(x_1) \in \mathbb{R}$ are the state and control vectors of the subsystem 1, and $x_2 = [x_{21}, x_{22}]^{\mathsf{T}} \in \mathbb{R}^2$ and $u_{22}(x_2) \in \mathbb{R}$ are the state and control vectors of the subsystem 2. The initial system state is $x(0) = [3.4, 2.7, 4.3, 1.2]^{\mathsf{T}}$. The weak coupling parameter is equal to $\varepsilon = 0.05$. The matrices Q and R are chosen as

$$Q_1 = Q_2 = R = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}, \qquad Q_\varepsilon = \begin{bmatrix} 1 & 0 \\ 0 & 0.05 \end{bmatrix}.$$

We assume that the exact knowledge of the system dynamics is completely unknown during the simulation. We adopt the integral PI algorithm to derive the optimal control law.

For the subsystem 1

$$\dot{x}_1 = \begin{bmatrix} -1.93x_{11}^2 \\ -1.394x_{11}x_{12} \end{bmatrix} + \begin{bmatrix} -1.274x_{11}^2 \\ 0 \end{bmatrix} u_{11}(x_1)$$

the weight parameters of the critic network and the action network are

$$\hat{\omega}_1 = \left[\hat{\omega}_{11}, \hat{\omega}_{12}, \hat{\omega}_{13}\right]^{\mathsf{T}}$$
$$\hat{\nu}_1 = \left[\hat{\nu}_{11}, \hat{\nu}_{12}\right]^{\mathsf{T}}.$$

The activation functions are chosen as

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

From the activation functions, we have $N_{1c} = 3$ and $N_{1a} = 2$ and we select $K_1 = 10$ to conduct the simulation. The initial weights are chosen as $\hat{\omega}_1 = [0, 0, 0]^{\mathsf{T}}$ and $\hat{\nu}_1 = [2, 1]^{\mathsf{T}}$. During the online learning process, the time period $T = 0.1$[s] and the exploration signal $e_1(t) = 3\sin(2\pi t) + 3\cos(2\pi t)$ are used. The least squares problem is solved after K_1 samples are acquired, thus the weights of the NNs are updated every 1[s]. The evolution of the action network 1's weights is illustrated in Fig. 1. After 52 iterations, the precision $\epsilon = 10^{-4}$ is achieved. At time $t = 52$[s], $\hat{\nu}_1^* = [-1.2557, -0.1067]^{\mathsf{T}}$.

For the subsystem 2, the activation functions are chosen as

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

As $N_{2c} = 3$ and $N_{2a} = 2$, we conduct the simulation with $K_2 = 10$. The initial weights are chosen as $\hat{\omega}_2 = [0, 0, 0]^{\mathsf{T}}$ and $\hat{\nu}_2 = [10, 2]^{\mathsf{T}}$. During the online learning process, the time period $T = 0.1$[s] and the exploration signal $e_2(t) = 5\sin(2\pi t) + 5\cos(2\pi t)$ are used. The evolution of the action network 2's weights is illustrated in Fig. 2. After 50 iterations, the precision ϵ is achieved. At time $t = 50$[s], $\hat{\nu}_2^* = [-9.9814, 0.0367]^{\mathsf{T}}$.

For the virtual subsystem 3, the weight parameters of the critic network and the action network are

$$\hat{\omega}_3 = \left[\hat{\omega}_{31}, \hat{\omega}_{32}, \hat{\omega}_{33}, \hat{\omega}_{34}, \hat{\omega}_{35}, \hat{\omega}_{36}\right]^{\mathsf{T}}$$
$$\hat{\nu}_3 = \left[\hat{\nu}_{31}, \hat{\nu}_{32}, \hat{\nu}_{33}, \hat{\nu}_{34}\right]^{\mathsf{T}}.$$

The activation functions are chosen as

$$\phi_3(x) = \left[x_{11}^2, x_{11}x_{12}, x_{12}^2, x_{21}^2, x_{21}x_{22}, x_{22}^2\right]^{\mathsf{T}}$$
$$\psi_3(x) = \left[x_{11}x_{12}, x_{12}^2, x_{21}x_{22}, x_{22}^2\right]^{\mathsf{T}}.$$

From the activation functions, we have $N_{3c} = 6$ and $N_{3a} = 4$ and we select $K_3 = 10$ to conduct the simulation. The initial weights are chosen as $\hat{\omega}_3 = [0, 0, 0, 0, 0, 0]^{\mathsf{T}}$ and $\hat{\nu}_3 = [0, -2, 2, 3]^{\mathsf{T}}$. During the online learning process, the time period $T = 0.1$[s] and the exploration signals $e_1(t)$ and $e_2(t)$ are used. The least squares problem is solved after K_3 samples are acquired, and the weights are updated every 1[s]. After 20 iterations, the precision $\epsilon = 10^{-4}$ is achieved. At time $t = 20$[s], $\hat{\nu}_3^* = [0.3830, 0.0533, -0.0899, -0.9548]^{\mathsf{T}}$.

According to the results in Section III, the optimal control law of the weakly coupled system can be derived as

$$u^*(x) = \begin{bmatrix} u_{11}^*(x_1) + \varepsilon u_{12}^*(x) \\ \varepsilon u_{21}^*(x) + u_{22}^*(x_2) \end{bmatrix}.$$

Using the optimal control $u^*(x)$ to control the weakly coupled system for 20[s], we obtain the evolution process of the state trajectory and control trajectory shown in Figs. 3 and 4. Obviously, these simulation results have verified the effectiveness of the developed model-free integral PI algorithm.

Example 2: In this example, we use the established model-free integral PI algorithm to balance a bicycle riding at a constant speed on a horizontal surface. The steering column of the bicycle is vertical, which means that the bicycle is not

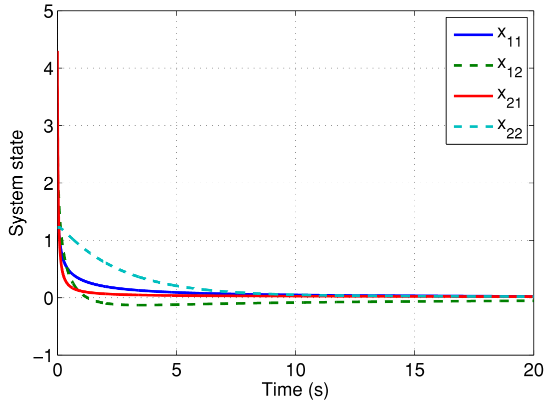


Fig. 3. State trajectory of the weakly coupled system under the derived optimal control.

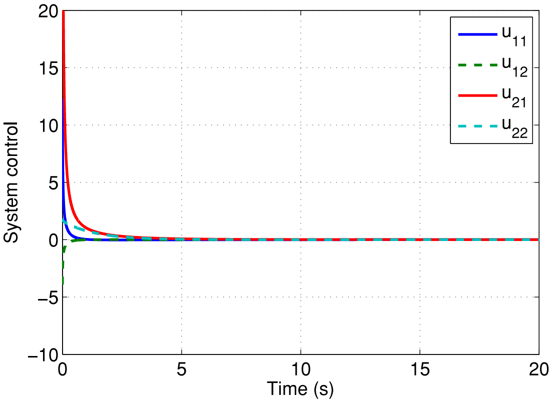


Fig. 4. Control trajectory of the weakly coupled system under the derived optimal control.

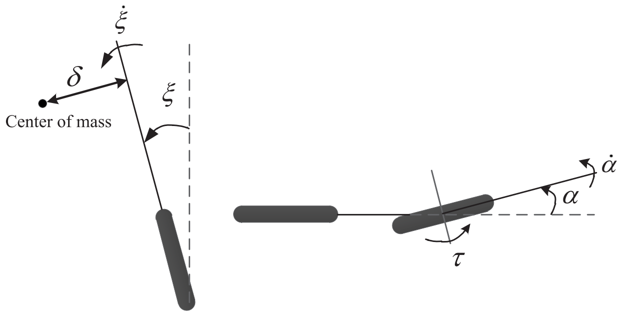


Fig. 5. Schematic representation of the bicycle, as seen from behind (left) and from the top (right).

self-stabilizing, but must be actively stabilized to prevent it from falling [47]. This is a variant of a bicycle balancing and riding problem which is widely used as a benchmark for RL algorithms [46].

The schematic representation of the bicycle is provided in Fig. 5, which includes the system state and control variables. The system state variables are the roll angle ξ[rad] of the bicycle measured from the vertical axis, the angle α[rad] of the handlebar, and the respective angular velocities $\dot{\xi}, \dot{\alpha}$[rad/s]. The control variables are the displacement δ[m] of the bicycle-rider common center of mass perpendicular to the plane of the bicycle, and the torque τ[Nm] applied to the handlebar.

TABLE I
PARAMETERS OF THE BICYCLE

Symbol	Value	Units	Symbol	Value	Units
M_c	15	kg	h	0.94	m
M_d	1.7	kg	l	1.11	m
M_r	60	kg	r	0.34	m
g	9.81	m/s^2	d_{CM}	0.3	m
v	10/3.6	m/s	c	0.66	m

Therefore, the state vector is $x = [\xi, \dot{\xi}, \alpha, \dot{\alpha}]^{\mathsf{T}}$, and the control vector is $u = [\delta, \tau]^{\mathsf{T}}$.

The dynamics of the bicycle can be represented as [46]

$$\begin{aligned}\ddot{\xi} &= \frac{1}{J_{bc}}\Bigg[\sin\beta(M_c + M_r)gh - \cos\beta\Bigg(\frac{J_{dc}v}{r}\dot{\alpha} \\ &\qquad + \operatorname{sign}(\alpha)\frac{M_d r v^2}{l}(|\sin\alpha| + |\tan\alpha|)\Bigg)\Bigg] \\ \ddot{\alpha} &= \frac{1}{J_{dl}}\left(\tau - \frac{J_{dv}v}{r}\dot{\xi}\right)\end{aligned}$$

where

$$\begin{aligned}J_{bc} &= \frac{13}{3}M_c h^2 + M_r(h + d_{CM})^2, \quad J_{dc} = M_d r^2 \\ J_{dv} &= \frac{3}{2}M_d r^2, \quad J_{dl} = \frac{1}{2}M_d r^2, \quad \beta = \xi + \arctan\frac{\delta}{h}.\end{aligned}$$

Table I shows the values of the parameters in the bicycle model. The meanings of these parameters are the same as those in [46]. Using the notations $x_1 = [\xi, \dot{\xi}]^{\mathsf{T}}$, $x_2 = [\alpha, \dot{\alpha}]^{\mathsf{T}}$, $u_1 = \delta$, and $u_2 = \tau$, we rewrite the bicycle dynamics as

$$\begin{bmatrix}\dot{x}_{11} \\ \dot{x}_{12} \\ \dot{x}_{21} \\ \dot{x}_{22}\end{bmatrix} = \begin{bmatrix} x_{12} \\ 4.62x_{11} + 0.054x_{21} + 0.011x_{22} + 4.62u_1 \\ x_{22} \\ 24.51x_{12} + 10.18u_2\end{bmatrix}.$$

Compared with the system (1) with weakly coupled structure, we have the following system dynamics:

$$\begin{aligned}f_{11}(x_1) &= \begin{bmatrix} x_{12} \\ 4.62x_{11}\end{bmatrix}, & f_{12}(x) &= \begin{bmatrix} 0 \\ 0.54x_{21} + 0.11x_{22}\end{bmatrix} \\ f_{21}(x) &= \begin{bmatrix} 0 \\ 245.1x_{12}\end{bmatrix}, & f_{22}(x_2) &= \begin{bmatrix} x_{22} \\ 0\end{bmatrix} \\ g_{11}(x_1) &= \begin{bmatrix} 0 \\ 4.62\end{bmatrix}, & g_{12}(x) &= \begin{bmatrix} 0 \\ 0\end{bmatrix} \\ g_{21}(x) &= \begin{bmatrix} 0 \\ 0\end{bmatrix}, & g_{22}(x_2) &= \begin{bmatrix} 0 \\ 10.18\end{bmatrix}.\end{aligned}$$

The initial system state is $x(0) = [0.1, -0.1, 0.1, -0.1]^{\mathsf{T}}$. The weak coupling parameter is $\varepsilon = 0.1$. The matrices Q and R are chosen as

$$Q_1 = Q_2 = R = \begin{bmatrix} 1 & 0 \\ 0 & 1\end{bmatrix}, \qquad Q_\varepsilon = \begin{bmatrix} 1 & 0 \\ 0 & 0.1\end{bmatrix}.$$

Assume that the exact knowledge of the bicycle is completely unknown during the simulation. We adopt the model-free integral PI algorithm to solve the bicycle balancing and riding problem.

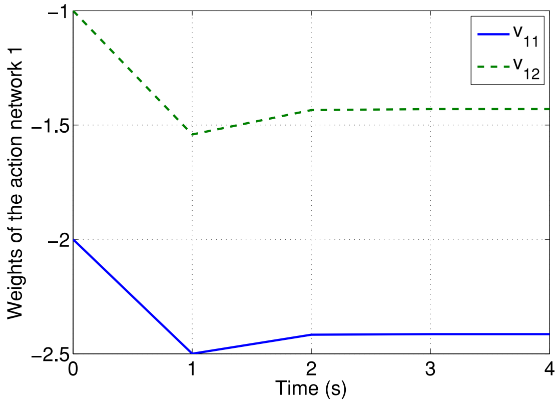


Fig. 6. Evolution of the action network 1's weights.

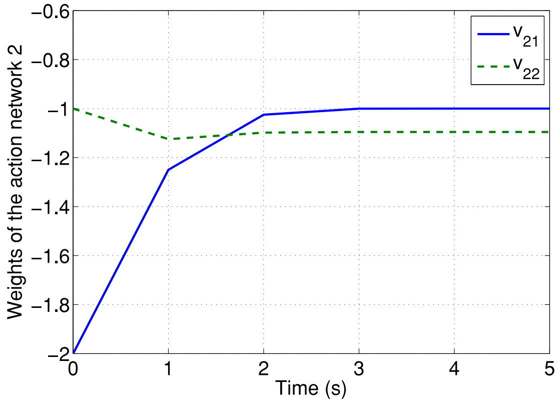


Fig. 7. Evolution of the action network 2's weights.

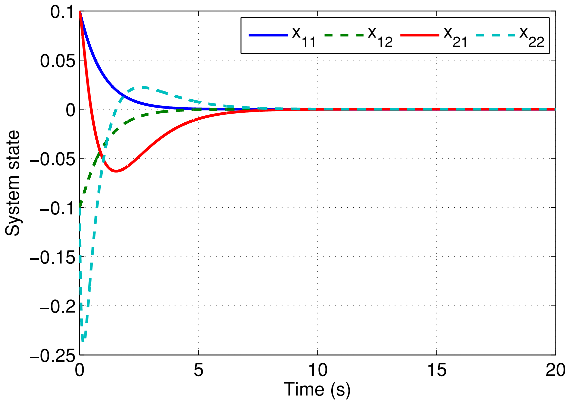


Fig. 8. State trajectory of the bicycle under the derived optimal control.

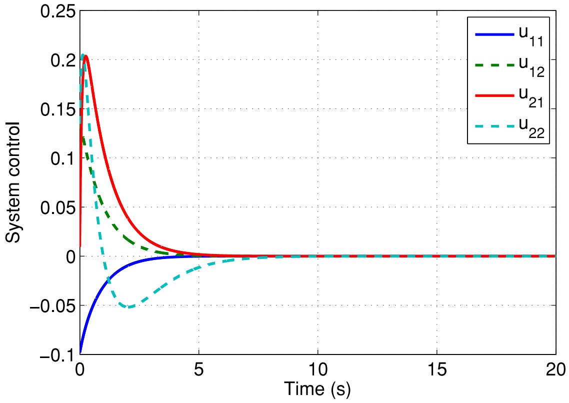


Fig. 9. Control trajectory of the bicycle under the derived optimal control.

As in Example 1, for the subsystem 1 the weight parameters of the critic network and the action network are

$$\hat{\omega}_1 = \left[\hat{\omega}_{11}, \hat{\omega}_{12}, \hat{\omega}_{13}\right]^{\mathsf{T}}$$
$$\hat{\nu}_1 = \left[\hat{\nu}_{11}, \hat{\nu}_{12}\right]^{\mathsf{T}}.$$

The activation functions are chosen as

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

From the activation functions, we have $N_{1c} = 3$ and $N_{1a} = 2$ and we select $K_1 = 10$ to conduct the simulation. We set the initial weights as $\hat{\omega}_1 = [0, 0, 0]^{\mathsf{T}}$ and $\hat{\nu}_1 = [-2, -1]^{\mathsf{T}}$. During the online learning process, the time period $T = 0.1$[s] and the exploration signal $e_1(t) = 0.05\sin(2\pi t) + 0.05\cos(2\pi t)$ are used. The least squares problem is solved after K_1 samples are acquired, and thus the weights of the NNs are updated every 1[s]. The evolution of the action network 1's weights is illustrated in Fig. 6. After 4 iterations, the precision $\epsilon = 10^{-4}$ is achieved. At time $t = 4$[s], $\hat{\nu}_1^* = [-2.4142, -1.4301]^{\mathsf{T}}$.

For the subsystem 2, the activation functions are chosen as

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

As $N_{2c} = 3$ and $N_{2a} = 2$, we conduct the simulation with $K_2 = 10$. The initial weights are chosen as $\hat{\omega}_2 = [0, 0, 0]^{\mathsf{T}}$ and $\hat{\nu}_2 = [-2, -1]^{\mathsf{T}}$. During the online learning process, the time period $T = 0.1$[s] and the exploration signal $e_2(t) = 0.05\sin(2\pi t) + 0.05\cos(2\pi t)$ are used. The evolution of the action network 2's weights is illustrated in Fig. 7. After 5 iterations, the precision $\epsilon = 10^{-4}$ is achieved. At time $t = 5$[s], $\hat{\nu}_2^* = [-1.0000, -1.0955]^{\mathsf{T}}$.

For the virtual subsystem 3, the weight parameters of the critic network and the action network are

$$\hat{\omega}_3 = \left[\hat{\omega}_{31}, \hat{\omega}_{32}, \hat{\omega}_{33}, \hat{\omega}_{34}, \hat{\omega}_{35}, \hat{\omega}_{36}\right]^{\mathsf{T}}$$
$$\hat{\nu}_3 = \left[\hat{\nu}_{31}, \hat{\nu}_{32}, \hat{\nu}_{33}, \hat{\nu}_{34}\right]^{\mathsf{T}}.$$

The activation functions are chosen as

$$\phi_3(x) = \left[x_{11}^2, x_{11}x_{12}, x_{12}^2, x_{21}^2, x_{21}x_{22}, x_{22}^2\right]^{\mathsf{T}}$$
$$\psi_3(x) = [x_{11}, x_{12}, x_{21}, x_{22}]^{\mathsf{T}}.$$

From the activation functions, we have $N_{3c} = 6$ and $N_{3a} = 4$ and we select $K_3 = 10$ to conduct the simulation. The initial weights are chosen as $\hat{\omega}_3 = [0, 0, 0, 0, 0, 0]^{\mathsf{T}}$ and $\hat{\nu}_3 = [0, -2, 2, 3]^{\mathsf{T}}$. During the online learning process, the time period $T = 0.1$[s] and the exploration signals $e_1(t)$ and $e_2(t)$ are used. The least squares problem is solved after K_3 samples are acquired, and the weights of the NNs are updated every 1[s]. After 23 iterations, the precision $\epsilon = 10^{-4}$ is achieved. At time $t = 23$[s], $\hat{\nu}_3^* = [0.5420, -0.8267, 0.6952, -0.6516]^{\mathsf{T}}$.

$$\begin{aligned}
0 &= \begin{bmatrix} V^*_{1x_1} + \varepsilon V^*_{\varepsilon x_1} \\ \varepsilon V^*_{\varepsilon x_2} + V^*_{2x_2} \end{bmatrix}^{\mathsf{T}} \left(\begin{bmatrix} f_{11}(x_1) + \varepsilon f_{12}(x) \\ \varepsilon f_{21}(x) + f_{22}(x_2) \end{bmatrix} + \begin{bmatrix} g_{11}(x_1) & \varepsilon g_{12}(x) \\ \varepsilon g_{21}(x) & g_{22}(x_2) \end{bmatrix} \begin{bmatrix} u^*_{11}(x_1) + \varepsilon u^*_{12}(x) \\ \varepsilon u^*_{21}(x) + u^*_{22}(x_2) \end{bmatrix} \right) \\
&\quad + \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} Q_1 & \varepsilon Q_\varepsilon \\ \varepsilon Q_\varepsilon & Q_2 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} u^*_{11}(x_1) + \varepsilon u^*_{12}(x) \\ \varepsilon u^*_{21}(x) + u^*_{22}(x_2) \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} R_1 & 0 \\ 0 & R_2 \end{bmatrix} \begin{bmatrix} u^*_{11}(x_1) + \varepsilon u^*_{12}(x) \\ \varepsilon u^*_{21}(x) + u^*_{22}(x_2) \end{bmatrix} \\
&= \begin{bmatrix} V^*_{1x_1} + \varepsilon V^*_{\varepsilon x_1} \\ \varepsilon V^*_{\varepsilon x_2} + V^*_{2x_2} \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} f_{11}(x_1) + g_{11}(x_1)u^*_{11}(x_1) + \varepsilon\big[f_{12}(x) + g_{11}(x_1)u^*_{12}(x) + g_{12}(x)u^*_{22}(x_2)\big] \\ f_{22}(x_2) + g_{22}(x_2)u^*_{22}(x_2) + \varepsilon\big[f_{21}(x) + g_{22}(x_2)u^*_{21}(x) + g_{21}(x)u^*_{11}(x_1)\big] \end{bmatrix} \\
&\quad + x_1^{\mathsf{T}} Q_1 x_1 + x_2^{\mathsf{T}} Q_2 x_2 + 2\varepsilon x_1^{\mathsf{T}} Q_\varepsilon x_2 + u^{*\mathsf{T}}_{11}(x_1) R_1 u^*_{11}(x_1) + u^{*\mathsf{T}}_{22}(x_2) R_2 u^*_{22}(x_2) \\
&\quad + 2\varepsilon u^{*\mathsf{T}}_{11}(x_1) R_1 u^*_{12}(x) + 2\varepsilon u^{*\mathsf{T}}_{22}(x_2) R_2 u^*_{21}(x) \\
&= \underbrace{V^{*\mathsf{T}}_{1x_1}\big[f_{11}(x_1) + g_{11}(x_1)u^*_{11}(x_1)\big] + x_1^{\mathsf{T}} Q_1 x_1 + u^{*\mathsf{T}}_{11}(x_1) R_1 u^*_{11}(x_1)}_{\text{HJB1}} \\
&\quad + \underbrace{V^{*\mathsf{T}}_{2x_2}\big[f_{22}(x_2) + g_{22}(x_2)u^*_{22}(x_2)\big] + x_2^{\mathsf{T}} Q_2 x_2 + u^{*\mathsf{T}}_{22}(x_2) R_2 u^*_{22}(x_2)}_{\text{HJB2}} \\
&\quad + \varepsilon V^{*\mathsf{T}}_{\varepsilon x_1}\big[f_{11}(x_1) + g_{11}(x_1)u^*_{11}(x_1)\big] + \varepsilon V^{*\mathsf{T}}_{1x_1}\big[f_{12}(x) + g_{11}(x_1)u^*_{12}(x) + g_{12}(x)u^*_{22}(x_2)\big] \\
&\quad + \varepsilon V^{*\mathsf{T}}_{\varepsilon x_2}\big[f_{22}(x_2) + g_{22}(x_2)u^*_{22}(x_2)\big] + \varepsilon V^{*\mathsf{T}}_{2x_2}\big[f_{21}(x) + g_{22}(x_2)u^*_{21}(x) + g_{21}(x)u^*_{11}(x_1)\big] \\
&\quad + 2\varepsilon x_1^{\mathsf{T}} Q_\varepsilon x_2 + 2\varepsilon u^{*\mathsf{T}}_{11}(x_1) R_1 u^*_{12}(x) + 2\varepsilon u^{*\mathsf{T}}_{22}(x_2) R_2 u^*_{21}(x)
\end{aligned} \tag{A1}$$

$$0 = \underbrace{V^{*\mathsf{T}}_{1x_1} f_{12}(x) + V^{*\mathsf{T}}_{2x_2} f_{21}(x) + V^{*\mathsf{T}}_{\varepsilon x_1} f_{11}(x_1) + V^{*\mathsf{T}}_{\varepsilon x_2} f_{22}(x_2) + 2x_1^{\mathsf{T}} Q_\varepsilon x_2 - 2u^{*\mathsf{T}}_{11}(x_1) R_1 u^*_{12}(x) - 2u^{*\mathsf{T}}_{22}(x_2) R_2 u^*_{21}(x)}_{\text{HJB3}} \tag{A2}$$

$$\begin{aligned}
u^*(x) &= -\frac{1}{2} R^{-1} g^{\mathsf{T}}(x) V^*_x = -\frac{1}{2} \begin{bmatrix} R_1 & 0 \\ 0 & R_2 \end{bmatrix}^{-1} \begin{bmatrix} g_{11}(x_1) & \varepsilon g_{12}(x) \\ \varepsilon g_{21}(x) & g_{22}(x_2) \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} V^*_{1x_1} + \varepsilon V^*_{\varepsilon x_1} \\ \varepsilon V^*_{\varepsilon x_2} + V^*_{2x_2} \end{bmatrix} \\
&= \begin{bmatrix} u^*_{11}(x_1) + \varepsilon u^*_{12}(x) \\ \varepsilon u^*_{21}(x) + u^*_{22}(x_2) \end{bmatrix} = -\frac{1}{2} \begin{bmatrix} R_1^{-1} g^{\mathsf{T}}_{11}(x_1) V^*_{1x_1} + \varepsilon R_1^{-1}\Big[g^{\mathsf{T}}_{11}(x_1) V^*_{\varepsilon x_1} + g^{\mathsf{T}}_{21}(x) V^*_{2x_2}\Big] \\ \varepsilon R_2^{-1}\Big[g^{\mathsf{T}}_{22}(x_2) V^*_{\varepsilon x_2} + g^{\mathsf{T}}_{12}(x) V^*_{1x_1}\Big] + R_2^{-1} g^{\mathsf{T}}_{22}(x_2) V^*_{2x_2} \end{bmatrix}
\end{aligned} \tag{A3}$$

According to the results in Section III, the optimal control law of the weakly coupled system can be derived as

$$u^*(x) = \begin{bmatrix} u^*_{11}(x_1) + \varepsilon u^*_{12}(x) \\ \varepsilon u^*_{21}(x) + u^*_{22}(x_2) \end{bmatrix}.$$

Using the optimal control $u^*(x)$ to control the weakly coupled system for 20[s], we obtain the evolution process of the state trajectory and control trajectory shown in Figs. 8 and 9. Obviously, these simulation results have verified the effectiveness of the developed model-free integral PI algorithm.

Remark 3: In Figs. 1 and 2, one weight parameter is largely dominated by the other one. While in Figs. 6 and 7, we can find that the weight parameters have the same order of magnitude. Selecting different activation functions may result in different converged weight vector.

V. Conclusion

In this paper, a data-based online learning algorithm for weakly coupled nonlinear systems is established. The optimal control law is derived by the optimal controllers of the reduced-order subsystems. We use the model-free integral PI algorithm with an exploration to solve the HJB equations related to the subsystems. We use the actor-critic technique and the least squares method to implement the constructed algorithm. The effectiveness of the developed optimal control law is demonstrated by two simulation examples.

Appendix
Proof of Theorem 1

Using the notation

$$V^*_x = \begin{bmatrix} V^*_{1x_1} + \varepsilon V^*_{\varepsilon x_1} \\ \varepsilon V^*_{\varepsilon x_2} + V^*_{2x_2} \end{bmatrix}$$

and setting $\varepsilon^2 = 0$, the HJB equation (6) can be rewritten as (A1), shown at the top of the page, which consists of three parts, i.e., HJB1, HJB2, and the last term which will be simplified as HJB3 in (A2), shown at the top of the page. The optimal control law $u^*(x)$ can be calculated as (A3), shown at the top of the page. Then we have the expressions of $u^*_{11}(x_1)$, $u^*_{12}(x)$, $u^*_{21}(x)$, and $u^*_{22}(x_2)$ as in (7). According to the optimal control theory, HJB1 $= 0$ is the HJB equation for the subsystem 1

$$\dot{x}_1(t) = f_{11}(x_1) + g_{11}(x_1)u_{11}(t)$$

and the optimal control law is $u^*_{11}(x_1) = -(1/2)R_1^{-1} g^{\mathsf{T}}_{11}(x_1) V^*_{1x_1}$. HJB2 $= 0$ is the HJB equation for the subsystem 2

$$\dot{x}_2(t) = f_{22}(x_2) + g_{22}(x_2)u_{22}(t)$$

and the optimal control law is $u^*_{22}(x_2) = -(1/2)R_2^{-1} g^{\mathsf{T}}_{22}(x_2) V^*_{2x_2}$.

To simplify the last term in (A1) besides HJB1 and HJB2, we give the following equations according to (A3):

$$\begin{aligned} V_{\varepsilon x_1}^{*\mathsf{T}} g_{11}(x_1) + V_{2x_2}^{*\mathsf{T}} g_{21}(x) &= -2u_{12}^{*\mathsf{T}}(x)R_1 \\ V_{\varepsilon x_2}^{*\mathsf{T}} g_{22}(x_2) + V_{1x_1}^{*\mathsf{T}} g_{12}(x) &= -2u_{21}^{*\mathsf{T}}(x)R_2 \\ V_{1x_1}^{*\mathsf{T}} g_{11}(x_1) &= -2u_{11}^{*\mathsf{T}}(x)R_1 \\ V_{2x_2}^{*\mathsf{T}} g_{22}(x_1) &= -2u_{22}^{*\mathsf{T}}(x)R_2. \end{aligned} \tag{A4}$$

Based on (A4), we obtain HJB3 $= 0$ as (A2). To solve $u_{12}^*(x)$ and $u_{21}^*(x)$ from HJB3, we integrate both sides of (A2) from t to ∞, and obtain

$$\begin{aligned} &\int_t^\infty \Big[V_{1x_1}^{*\mathsf{T}} f_{12}(x) + V_{2x_2}^{*\mathsf{T}} f_{21}(x) + V_{\varepsilon x_1}^{*\mathsf{T}} f_{11}(x_1) + V_{\varepsilon x_2}^{*\mathsf{T}} f_{22}(x_2) \Big] \mathrm{d}\tau \\ &\quad = 2\int_t^\infty \Big[u_{11}^{*\mathsf{T}}(x_1)R_1 u_{12}^*(x) + u_{22}^{*\mathsf{T}}(x_2)R_2 u_{21}^*(x) - x_1^{\mathsf{T}} Q_\varepsilon x_2 \Big] \mathrm{d}\tau. \end{aligned}$$

Using $V_\varepsilon^*(x)$, we have

$$\begin{aligned} V_3^*(x) = \int_t^\infty \Big[& V_{1x_1}^{*\mathsf{T}} f_{12}(x) + V_{2x_2}^{*\mathsf{T}} f_{21}(x) \\ & + V_{\varepsilon x_1}^{*\mathsf{T}} f_{11}(x_1) + V_{\varepsilon x_2}^{*\mathsf{T}} f_{22}(x_2) \Big] \mathrm{d}\tau \end{aligned}$$

where $V_3^*(x) = V_\varepsilon^*(x) - 4\int_t^\infty x_1^{\mathsf{T}} Q_\varepsilon x_2 \mathrm{d}\tau$ with $V_3^*(0) = 0$. The proof is complete.

References

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

[2] D. Liu, D. Wang, and H. Li, "Decentralized stabilization for a class of continuous-time nonlinear interconnected systems using online learning optimal control approach," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 2, pp. 418–428, Feb. 2014.

[3] P. Kokotović, W. R. Perkins, J. B. Cruz, and G. D'Ans, "ϵ-coupling method for near-optimum design of large-scale linear systems," *Proc. Inst. Elect. Eng.*, vol. 116, no. 5, pp. 889–892, May 1969.

[4] Z. Gajić and X. Shen, "Decoupling transformation for weakly coupled linear systems," *Int. J. Control*, vol. 50, no. 4, pp. 1517–1523, 1989.

[5] Z. Gajić and X. Shen, *Parallel Algorithms for Optimal Control of Large Scale Linear Systems*. London, U.K.: Springer, 1992.

[6] Z. Aganovic and Z. Gajic, "Optimal control of weakly coupled bilinear systems," *Automatica*, vol. 29, no. 6, pp. 1591–1593, 1993.

[7] Z. Aganović and Z. Gajic, *Linear Optimal Control of Bilinear Systems With Applications to Singular Perturbations and Weak Coupling*. London, U.K.: Springer, 1995.

[8] Y. Jiang and Z.-P. Jiang, "Robust adaptive dynamic programming for large-scale systems with an application to multimachine power systems," *IEEE Trans. Circuits Syst. II, Exp. Briefs*, vol. 59, no. 10, pp. 693–697, Oct. 2012.

[9] Y.-J. Kim and M.-T. Lim, "Parallel optimal control for weakly coupled nonlinear systems using successive Galerkin approximation," *IEEE Trans. Autom. Control*, vol. 53, no. 6, pp. 1542–1547, Jul. 2008.

[10] L. R. G. Carrillo, K. G. Vamvoudakis, and J. P. Hespanha, "Approximate optimal adaptive control for weakly coupled nonlinear systems: A neuro-inspired approach," *Int. J. Adapt. Control Signal Process.*, to be published. doi: 10.1002/acs.2631.

[11] R. Bellman, *Dynamic Programming*. Princeton, NJ, USA: Princeton Univ. Press, 1957.

[12] F.-Y. Wang, H. Zhang, and D. Liu, "Adaptive dynamic programming: An introduction," *IEEE Comput. Intell. Mag.*, vol. 4, no. 2, pp. 39–47, May 2009.

[13] Q. Wei and D. Liu, "Numerical adaptive learning control scheme for discrete-time non-linear systems," *IET Control Theory Appl.*, vol. 7, no. 11, pp. 1472–1486, Jul. 2013.

[14] Q. Wei and D. Liu, "A novel iterative θ-adaptive dynamic programming for discrete-time nonlinear systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 4, pp. 1176–1190, Oct. 2014.

[15] Q. Wei, F.-Y. Wang, D. Liu, and X. Yang, "Finite-approximation-error-based discrete-time iterative adaptive dynamic programming," *IEEE Trans. Cybern.*, vol. 44, no. 12, pp. 2820–2833, Dec. 2014.

[16] 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.

[17] Z. Ni, H. He, X. Zhong, and D. V. Prokhorov, "Model-free dual heuristic dynamic programming," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 8, pp. 1834–1839, Aug. 2015.

[18] D. Wang, D. Liu, and H. Li, "Policy iteration algorithm for online design of robust control for a class of continuous-time nonlinear systems," *IEEE Trans. Autom. Sci. Eng.*, vol. 11, no. 2, pp. 627–632, Apr. 2014.

[19] D. Wang, D. Liu, H. Li, and H. Ma, "Neural-network-based robust optimal control design for a class of uncertain nonlinear systems via adaptive dynamic programming," *Inf. Sci.*, vol. 282, pp. 167–179, Oct. 2014.

[20] Y. Jiang and Z.-P. Jiang, "Robust adaptive dynamic programming with an application to power systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 24, no. 7, pp. 1150–1156, Jul. 2013.

[21] H. Li and D. Liu, "Optimal control for discrete-time affine non-linear systems using general value iteration," *IET Control Theory Appl.*, vol. 6, no. 18, pp. 2725–2736, Dec. 2012.

[22] D. Liu, H. Li, and D. Wang, "Neural-network-based zero-sum game for discrete-time nonlinear systems via iterative adaptive dynamic programming algorithm," *Neurocomputing*, vol. 110, pp. 92–100, Jun. 2013.

[23] 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.

[24] D. Wang, C. Li, D. Liu, and C. Mu, "Data-based robust optimal control of continuous-time affine nonlinear systems with matched uncertainties," *Inf. Sci.*, vol. 366, pp. 121–133, Oct. 2016.

[25] D. Wang, D. Liu, Q. Zhang, and D. Zhao, "Data-based adaptive critic designs for nonlinear robust optimal control with uncertain dynamics," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published, doi: 10.1109/TSMC.2015.2492941.

[26] K. G. Vamvoudakis, "Event-triggered optimal adaptive control algorithm for continuous-time nonlinear systems," *IEEE/CAA J. Autom. Sinica*, vol. 1, no. 3, pp. 282–293, Jul. 2014.

[27] H. Zhang, D. Liu, Y. Luo, and D. Wang, *Adaptive Dynamic Programming for Control: Algorithms and Stability*. London, U.K.: Springer-Verlag, 2013.

[28] 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.

[29] H. Zhang, J. Zhang, G.-H. Yang, and Y. Luo, "Leader-based optimal coordination control for the consensus problem of multiagent differential games via fuzzy adaptive dynamic programming," *IEEE Trans. Fuzzy Syst.*, vol. 23, no. 1, pp. 152–163, Feb. 2015.

[30] 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.

[31] F. L. Lewis and D. Vrabie, "Reinforcement learning and adaptive dynamic programming for feedback control," *IEEE Circuits Syst. Mag.*, vol. 9, no. 3, pp. 32–50, Jul. 2009.

[32] R. S. Sutton and A. G. Barto, *Reinforcement Learning: An introduction*, vol. 1. Cambridge, U.K.: Cambridge Univ. Press, 1998.

[33] J. Si, A. G. Barto, W. B. Powell, and D. C. Wunsch, *Handbook of Learning and Approximate Dynamic Programming*. Los Alamitos, CA, USA: IEEE Press, 2004.

[34] S. J. Bradtke, B. E. Ydstie, and A. G. Barto, "Adaptive linear quadratic control using policy iteration," in *Proc. Amer. Control Conf.*, vol. 3. Baltimore, MD, USA, Jul. 1994, pp. 3475–3479.

[35] D. Vrabie and F. Lewis, "Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear systems," *Neural Netw.*, vol. 22, no. 3, pp. 237–246, 2009.

[36] D. Liu, H. Li, and D. Wang, "Online synchronous approximate optimal learning algorithm for multi-player non-zero-sum games with unknown dynamics," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 8, pp. 1015–1027, Aug. 2014.

[37] Y. Jiang and Z.-P. Jiang, "Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics," *Automatica*, vol. 48, no. 10, pp. 2699–2704, Oct. 2012.

[38] Y. Jiang and Z.-P. Jiang, "Global adaptive dynamic programming for continuous-time nonlinear systems," *IEEE Trans. Autom. Control*, vol. 60, no. 11, pp. 2917–2929, Nov. 2015.

[39] J. Y. Lee, J. B. Park, and Y. H. Choi, "Integral *Q*-learning and explorized policy iteration for adaptive optimal control of continuous-time linear systems," *Automatica*, vol. 48, no. 11, pp. 2850–2859, Nov. 2012.

[40] F. L. Lewis, D. L. Vrabie, and V. L. Syrmos, *Optimal Control*. Hoboken, NJ, USA: Wiley, 2012.

[41] G. N. Saridis and C.-S. G. Lee, "An approximation theory of optimal control for trainable manipulators," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 9, no. 3, pp. 152–159, Mar. 1979.

[42] R. W. Beard, G. N. Saridis, and J. T. Wen, "Galerkin approximations of the generalized Hamilton-Jacobi-Bellman equation," *Automatica*, vol. 33, no. 12, pp. 2159–2177, Dec. 1997.

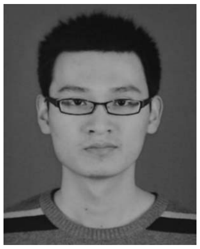
[43] H. Zhang, Y. Luo, and D. Liu, "Neural-network-based near-optimal control for a class of discrete-time affine nonlinear systems with control constraints," *IEEE Trans. Neural Netw.*, vol. 20, no. 9, pp. 1490–1503, Sep. 2009.

[44] H. Zhang, T. Feng, G.-H. Yang, and H. Liang, "Distributed cooperative optimal control for multiagent systems on directed graphs: An inverse optimal approach," *IEEE Trans. Cybern.*, vol. 45, no. 7, pp. 1315–1326, Jul. 2015.

[45] D. Wang, D. Liu, H. Li, B. Luo, and H. Ma, "An approximate optimal control approach for robust stabilization of a class of discrete-time nonlinear systems with uncertainties," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 5, pp. 713–717, May 2016.

[46] D. Ernst, P. Geurts, and L. Wehenkel, "Tree-based batch mode reinforcement learning," *J. Mach. Learn. Res.*, vol. 6, pp. 503–556, Apr. 2005.

[47] L. Busoniu, R. Babuška, B. De Schutter, and D. Ernst, *Reinforcement Learning and Dynamic Programming Using Function Approximators*, vol. 39. Boca Raton, FL, USA: CRC press, 2010.



Chao Li received the B.S. degree in mechatronics from the Nanjing University of Science and Technology, Nanjing, China, in 2012. He is currently pursuing the Ph.D. degree with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing, China.

He is with the University of Chinese Academy of Sciences, Beijing. His current research interests include intelligent control, neural networks, reinforcement learning, and adaptive dynamic programming.

Derong Liu (S'91–M'94–SM'96–F'05) received the Ph.D. degree in electrical engineering from the University of Notre Dame, Notre Dame, IN, USA, in 1994.

He was a Staff Fellow with the General Motors Research and Development Center, from 1993 to 1995. He was an Assistant Professor with the Department of Electrical and Computer Engineering, Stevens Institute of Technology, Hoboken, NJ, USA, from 1995 to 1999. He joined the University of Illinois at Chicago, Chicago, IL, USA, in 1999, where he became a Full Professor of Electrical and Computer Engineering and of Computer Science in 2006. He was selected for the "100 Talents Program" by the Chinese Academy of Sciences, Beijing, China, in 2008, where he served as the Associate Director of the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, from 2010 to 2015. He is currently a Full Professor with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing. He has published 15 books (six research monographs and nine edited volumes).

Prof. Liu was a recipient of the Faculty Early Career Development Award from the National Science Foundation in 1999, the University Scholar Award from the University of Illinois from 2006 to 2009, the Overseas Outstanding Young Scholar Award from the National Natural Science Foundation of China in 2008, and the Outstanding Achievement Award from Asia Pacific Neural Network Assembly in 2014. He is an elected AdCom Member of the IEEE Computational Intelligence Society and he is an Editor-in-Chief of *Artificial Intelligence Review*. He was the General Chair of 2014 IEEE World Congress on Computational Intelligence and is the General Chair of 2016 World Congress on Intelligent Control and Automation. He is a fellow of the International Neural Network Society.

Ding Wang (M'15) received the B.S. degree in mathematics from the Zhengzhou University of Light Industry, Zhengzhou, China, in 2007, the M.S. degree in operations research and cybernetics from Northeastern University, Shenyang, China, in 2009, and the Ph.D. degree in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2012.

He has been a Visiting Scholar with the Department of Electrical, Computer, and Biomedical Engineering, University of Rhode Island, Kingston, RI, USA, since 2015. He is currently an Associate Professor with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences. He has published over 70 journal and conference papers, and coauthored two monographs. His current research interests include adaptive and learning systems, intelligent control, and neural networks.

Dr. Wang was a recipient of the Excellent Doctoral Dissertation Award of the Chinese Academy of Sciences in 2013, and the Nominee of the Excellent Doctoral Dissertation Award of Chinese Association of Automation (CAA) in 2014. He was the Secretariat of the 2014 IEEE World Congress on Computational Intelligence, and the Registration Chair of the 5th International Conference on Information Science and Technology and the 4th International Conference on Intelligent Control and Information Processing, and served as the Program Committee Member for several international conferences. He is the Finance Chair of the 12th World Congress on Intelligent Control and Automation. He serves as an Associate Editor for the IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS and *Neurocomputing*. He is a member of Asia–Pacific Neural Network Society and CAA.