2014 International Joint Conference on Neural Networks (IJCNN)
July 6-11, 2014, Beijing, China

Near-Optimal Online Control of Uncertain Nonlinear Continuous-Time Systems Based on Concurrent Learning

Xiong Yang, Derong Liu, and Qinglai Wei

Abstract— **This paper presents a novel observer–critic architecture for solving the near-optimal control problem of uncertain nonlinear continuous-time systems. Two neural networks (NNs) are employed in the architecture: an observer NN is constructed to get the knowledge of uncertain system dynamics and a critic NN is utilized to derive the optimal control. The observer NN and the critic NN are tuned simultaneously. By using the recorded and instantaneous data together, the optimal control can be derived without the persistence of excitation condition. Meanwhile, the closed-loop system is guaranteed to be stable in the sense of uniform ultimate boundedness. No initial stabilizing control is required in the developed algorithm. An illustrated example is provided to demonstrate the effectiveness of the present approach.**

I. Introduction

THE core challenge of obtaining the solution for the nonlinear optimal control problem is that it is often necessary to solve the Hamilton-Jacobi-Bellman (HJB) equation [1], [2], [3], [4], [5]. The HJB equation is actually a partial differential/difference equation (PDE), which is intractable to solve by analytical methods. For the sake of coping with the problem, Bellman introduced the dynamic programming (DP) method [6]. Though DP is successfully utilized to derive the optimal control, a shortcoming of the approach is that the computation grows exponentially with increase in the dimensionality of nonlinear systems.

In order to overcome the difficulty of applying DP, Werbos proposed adaptive/approximate dynamic programming (ADP) algorithms, which derive approximate solutions of HJB equations forward-in-time [7], [8]. Unfortunately, most of ADP approaches are either implemented offline by using iterative schemes or they require a priori knowledge of system dynamics. Hence, many ADP approaches cannot be applied to real-time process control. After that, reinforcement learning (RL) methods are developed. RL is a class of approaches used in machine learning to revise the actions of an agent based on responses from its environment [9]. The actor–critic architecture has been typically used to implement the RL algorithm, where the actor performs actions by interacting with its environment, and the critic evaluates actions and offers feedback information to the actor, leading to the improvement in performance of the subsequent actor [10]. In contrast to ADP methods, a distinct advantage of RL approaches is that no prescribed behavior or training model is required.

Up to now, while RL has been widely employed to derive the optimal control for nonlinear systems [11], [12], [13], [14], [15], [16], [17], [18], most of these applications depend on an initial stabilizing control. From a mathematical perspective, the initial stabilizing control is a suboptimal control. The suboptimal control of the nonlinear system is intractable to be obtained since it is generally impossible to get analytical solutions of PDEs. On the other hand, persistence of excitation (PE) is an indispensable condition for obtaining the optimal control in aforementioned literature. It should be mentioned that the PE condition is often difficult to verify. In addition, all above literature assumed that the system states were known. In practice, however, system states are often unavailable. Estimations of states from the system output for obtaining the adaptive optimal control is necessary.

In this paper, we develop a novel observer–critic architecture for solving the near-optimal control problem of uncertain nonlinear continuous-time (CT) systems. We employ two neural networks (NNs) in the architecture: an observer NN is utilized to get the knowledge of uncertain systems dynamics and a critic NN is used to derive the optimal control. We tune the observer NN and the critic NN simultaneously. By using the recorded and instantaneous data together (i.e., concurrent learning method), we obtain the optimal control without the PE condition. Meanwhile, we keep the closed-loop system stable in the sense of uniform ultimate boundedness. In addition, we do not need the initial stabilizing control based on the developed algorithm.

The paper is organized as follows. Section II provides preliminaries of optimal control problems of nonlinear CT systems. Section III presents the design of an online optimal control. Section IV develops the stability analysis. Section V provides an example to demonstrate the effectiveness of theorem developed in Section IV . Finally, Section VI gives several conclusions.

II. Problem Statement and Preliminaries

Consider the nonlinear CT system given by the form

$$\begin{aligned} \dot{x}(t) &= f(x(t)) + g(x(t))u(x(t)) \\ y(t) &= Cx(t) \end{aligned} \tag{1}$$

with the state $x(t) \in \mathbb{R}^n$, the control $u(t) \in \mathbb{R}^m$, the output $y(t) \in \mathbb{R}^p$, the unknown nonlinear function $f(x) \in \mathbb{R}^n$, and the functional matrix $g(x) \in \mathbb{R}^{n\times m}$. It is assumed that $f(x)+$

The authors are with The State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China (phone: +86-10-82544761; fax: +86-10-82544799; email: xiong.yang@ia.ac.cn; derong.liu@ia.ac.cn; qinglai.wei@ia.ac.cn).

This work was supported in part by the National Natural Science Foundation of China under Grants 61034002, 61233001, 61273140, and 61374105, and in part by Beijing Natural Science Foundation under Grant 4132078.

978-1-4799-1484-5/14/$31.00 ©2014 IEEE

$g(x)u$ is Lipschitz continuous on a compact set $\Omega \subset \mathbb{R}^n$ containing the origin, such that the solution $x(t)$ of system (1) is unique on Ω, and $f(0) = 0$. The state of system (1) is unavailable, only the system output $y(t)$ can be measured.

Assumption 1: The control matrix $g(x)$ is known and bounded over Ω, i.e., there exist g_m and g_M $(0 < g_m < g_M)$ such that $g_m \le \|g(x)\| \le g_M$, for $\forall x \in \Omega$.

Assumption 2: System (1) is observable and the state $x(t)$ is bounded in L_∞ [19]. In addition, $C \in \mathbb{R}^{p\times n}$ $(p \le n)$ is a full row rank matrix, i.e., $\mathrm{rank}(C) = p$.

The value function for system (1) is given by

$$V(x(t)) = \int_t^\infty r\big(y(s), u(s)\big)\mathrm{d}s \quad (s \ge t) \tag{2}$$

where $r(y,u) = y^\mathsf{T}Qy + u^\mathsf{T}Ru$, Q and R are constant symmetric positive definite matrices.

Objective of Control: The control goal of the paper is to get an online adaptive control not only stabilizes system (1) but also minimizes the value function (2), while guaranteeing that all the signals involved in the closed-loop system are uniformly ultimately bounded (UUB).

III. Online Optimal Control Design

A. NN State Observer

Due to the unavailability of system states, a two-layer feedforward NN is employed to construct the state observer. According to [20], $\mathcal{F}(x) \in \mathrm{C}^n(\Omega)$ can be represented by feedforward NNs as

$$\mathcal{F}(x) = W_1^\mathsf{T}\sigma(V_1^\mathsf{T}x) + \varepsilon_1(x) \tag{3}$$

where $\sigma(\cdot) \in \mathbb{R}^{N_1}$ is the activation function, $V_1 \in \mathbb{R}^{n\times N_1}$ and $W_1 \in \mathbb{R}^{N_1\times n}$ are the weights for the input layer to the hidden layer and the hidden layer to the output layer, respectively, N_1 is the number of nodes in the hidden layer, and $\varepsilon_1(x) \in \mathbb{R}^n$ is the NN function reconstruction error.

From system (1), we have

$$\begin{aligned}\dot{x}(t) &= Ax + \mathcal{F}(x) + g(x)u\\ y(t) &= Cx(t)\end{aligned} \tag{4}$$

where $\mathcal{F}(x) = f(x) - Ax$, $A \in \mathbb{R}^{n\times n}$ is a known constant matrix, and (C, A) is observable.

By using (3), we can rewrite (4) as

$$\begin{aligned}\dot{x}(t) &= Ax + W_1^\mathsf{T}\sigma(V_1^\mathsf{T}x) + g(x)u + \varepsilon_1(x)\\ y(t) &= Cx(t).\end{aligned} \tag{5}$$

The NN state observer for system (1) is developed as

$$\begin{aligned}\dot{\hat{x}}(t) &= A\hat{x} + \hat{W}_1^\mathsf{T}\sigma(\hat{V}_1^\mathsf{T}\hat{x}) + g(\hat{x})u + B(y - \hat{y})\\ \hat{y}(t) &= C\hat{x}(t)\end{aligned} \tag{6}$$

where $\hat{x}(t) \in \mathbb{R}^n$ and $\hat{y}(t) \in \mathbb{R}^p$ are the state and the output of the observer respectively, $\hat{W}_1 \in \mathbb{R}^{N_1\times n}$ and $\hat{V}_1 \in \mathbb{R}^{n\times N_1}$ are estimated weights, and the observer gain $B \in \mathbb{R}^{n\times p}$ is chosen such that the matrix $A - BC$ is Hurwitz.

Define $\tilde{x}(t) = x(t) - \hat{x}(t)$ and $\tilde{y}(t) = y(t) - \hat{y}(t)$. From (5) and (6), the observer error dynamics is derived as

$$\begin{aligned}\dot{\tilde{x}}(t) &= A_c\tilde{x}(t) + \tilde{W}_1^\mathsf{T}\sigma(\hat{V}_1^\mathsf{T}\hat{x}) + \delta(x)\\ \tilde{y}(t) &= C\tilde{x}(t)\end{aligned} \tag{7}$$

where $A_c = A - BC$, $\tilde{W}_1 = W_1 - \hat{W}_1$, and $\delta(x) = W_1^\mathsf{T}\big[\sigma(V_1{}^\mathsf{T}x) - \sigma(\hat{V}_1^\mathsf{T}\hat{x})\big] + \big[g(x) - g(\hat{x})\big]u + \varepsilon_1(x)$.

Before showing the stability of the observer error $\tilde{x}(t)$, we provide the following assumptions and facts.

Assumption 3: The ideal observer NN weights W_1 and V_1 are bounded over Ω by known positive constants W_M and V_M, respectively. That is, $\|W_1\| \le W_M,\ \|V_1\| \le V_M$.

Assumption 4: The NN function reconstruction error $\varepsilon_1(x)$ is bounded over Ω as $\|\varepsilon_1(x)\| \le \varepsilon_M$, where $\varepsilon_M > 0$.

Fact 1: The NN activation function is bounded over Ω, that is, there exists $\sigma_M > 0$ such that $\|\sigma(x)\| \le \sigma_M$, for $\forall x \in \Omega$.

Fact 2: Since the matrix A_c is Hurwitz, there exists a positive-definite symmetric matrix $P \in \mathbb{R}^{n\times n}$ satisfying the Lyapunov equation

$$A_c^\mathsf{T}P + PA_c = -\theta I_n$$

where $\theta > 0$ is a design parameter.

Theorem 1: Let Assumptions 1–4 hold. If NN estimated weights $\hat{W}_1$ and $\hat{V}_1$ are updated as

$$\dot{\hat{W}}_1 = -\,l_1\sigma\big(\hat{V}_1^\mathsf{T}\hat{x}\big)\tilde{y}^\mathsf{T}CA_c^{-1} - \kappa_1\|\tilde{y}\|\hat{W}_1 \tag{8}$$

$$\dot{\hat{V}}_1 = -\,l_2\mathrm{sgn}(\hat{x})\tilde{y}^\mathsf{T}CA_c^{-1}\hat{W}_1^\mathsf{T}\Big(I_{N_1} - \Phi(\hat{V}_1^\mathsf{T}\hat{x})\Big) - \kappa_2\|\tilde{y}\|\hat{V}_1 \tag{9}$$

with design parameters $l_i > 0$ and $\kappa_i > 0$ $(i = 1, 2)$, $\Phi\big(\hat{V}_1^\mathsf{T}\hat{x}\big) = \mathrm{diag}\big\{\sigma_k^2\big(\hat{V}_{1k}^\mathsf{T}\hat{x}\big)\big\}$ $(k = 1, \ldots, N_1)$, and $\mathrm{sgn}(\hat{x}) = [\mathrm{sgn}(\hat{x}_1), \ldots, \mathrm{sgn}(\hat{x}_n)]^\mathsf{T}$, where $\mathrm{sgn}(\hat{x}_\iota)$ $(\iota = 1, \ldots, n)$ is the sign function with respect to $\hat{x}_\iota$. Then, the NN observer developed in (6) can ensure that $\tilde{x}(t)$ converges to the compact set

$$\Omega_{\tilde{x}} = \left\{\tilde{x}\colon \|\tilde{x}\| \le \frac{2\mathcal{B}}{\theta\|C\|\lambda_{\min}\big[(C^+)^\mathsf{T}C^+\big]}\right\} \tag{10}$$

where $\mathcal{B}$ is to be detailed (see (15)), C^+ is the Moore-Penrose pseudoinverse of C, and $\lambda_{\min}\big[(C^+)^\mathsf{T}C^+\big]$ is the minimum eigenvalue of $(C^+)^\mathsf{T}C^+$. In addition, NN weight estimation errors $\tilde{W}_1$ and $\tilde{V}_1 = V_1 - \hat{V}_1$ are guaranteed to be UUB.

Proof: Consider the Lyapunov function candidate

$$J(t) = J_1(t) + J_2(t)$$

where

$$\begin{aligned}J_1(t) &= \frac{1}{2}\tilde{x}^\mathsf{T}P\tilde{x}\\ J_2(t) &= \frac{1}{2}\mathrm{tr}\big(\tilde{W}_1^\mathsf{T}l_1^{-1}\tilde{W}_1\big) + \frac{1}{2}\mathrm{tr}\big(\tilde{V}_1^\mathsf{T}l_2^{-1}\tilde{V}_1\big).\end{aligned}$$

Taking the time derivative of $J_1(t)$ and using Facts 1–2, we derive

$$\begin{aligned}\dot{J}_1(t) \le &-\frac{\theta}{2}\lambda_{\min}\big[(C^+)^\mathsf{T}C^+\big]\|\tilde{y}\|^2\\ &+ \|\tilde{y}\|\|(C^+)^\mathsf{T}P\|\big(\|\tilde{W}_1\|\sigma_M + \delta_M\big).\end{aligned} \tag{11}$$

Taking the time derivative of $J_2(t)$ and using (8) and (9), we obtain

$$\begin{aligned}\dot{J}_2(t) \leq\ & \alpha\sigma_M\|\tilde{y}\|\|\tilde{W}_1\| + \frac{\kappa_1}{l_1}\|\tilde{y}\|\Big(W_M\|\tilde{W}_1\| - \|\tilde{W}_1\|^2\Big) \\ & + \alpha\big\|I_{N_1} - \Phi(\hat{V}_1^{\mathsf{T}}\hat{x})\big\|\|\tilde{y}\|\big(W_M + \|\tilde{W}_1\|\big)\|\tilde{V}_1\| \\ & + \frac{\kappa_2}{l_2}\|\tilde{y}\|\Big(V_M\|\tilde{V}_1\| - \|\tilde{V}_1\|^2\Big) \end{aligned} \tag{12}$$

where $\alpha = \|CA_c^{-1}\|$.

Combining (11) with (12) and noting $\|I_{N_1} - \Phi(\hat{V}_1^{\mathsf{T}}\hat{x})\| \leq 1$, we get

$$\begin{aligned}\dot{J}(t) \leq & -\frac{\theta}{2}\lambda_{\min}\big[(C^+)^{\mathsf{T}}C^+\big]\|\tilde{y}\|^2 + \bigg\{\delta_M\|(C^+)^{\mathsf{T}}P\| \\ & + \Big(\big(\|(C^+)^{\mathsf{T}}P\| + \alpha\big)\sigma_M + \frac{\kappa_1}{l_1}W_M\Big)\|\tilde{W}_1\| \\ & + \Big(\alpha W_M + \frac{\kappa_2}{l_2}V_M\Big)\|\tilde{V}_1\| - \Big(\frac{\kappa_1}{l_1} - \frac{\alpha^2}{4}\Big)\|\tilde{W}_1\|^2 \\ & - \Big(\frac{\kappa_2}{l_2} - 1\Big)\|\tilde{V}_1\|^2 - \Big(\frac{\alpha}{2}\|\tilde{W}_1\| - \|\tilde{V}_1\|\Big)^2\bigg\}\|\tilde{y}\| \\ = & -\frac{\theta}{2}\lambda_{\min}\big[(C^+)^{\mathsf{T}}C^+\big]\|\tilde{y}\|^2 + \bigg\{\delta_M\|(C^+)^{\mathsf{T}}P\| \\ & + \Big(\frac{\kappa_1}{l_1} - \frac{\alpha^2}{4}\Big)\beta_1^2 + \Big(\frac{\kappa_2}{l_2} - 1\Big)\beta_2^2 \\ & - \Big(\frac{\kappa_1}{l_1} - \frac{\alpha^2}{4}\Big)\|\tilde{W}_1 + \beta_1\|^2 - \Big(\frac{\kappa_2}{l_2} - 1\Big)\|\tilde{V}_1 + \beta_2\|^2 \\ & - \Big(\frac{\alpha}{2}\|\tilde{W}_1\| - \|\tilde{V}_1\|\Big)^2\bigg\}\|\tilde{y}\| \end{aligned} \tag{13}$$

where

$$\begin{aligned}\beta_1 &= \frac{2l_1\big(\alpha + \|(C^+)^{\mathsf{T}}P\|\big)\sigma_M + 2\kappa_1 W_M}{\alpha^2 l_1 - 4\kappa_1} \\ \beta_2 &= \frac{\alpha l_2 W_M + \kappa_2 V_M}{2(l_2 - \kappa_2)}.\end{aligned}$$

Selecting $\kappa_1 > \alpha^2 l_1/4$, $\kappa_2 > l_2$ and using (13), we derive

$$\begin{aligned}\dot{J}(t) \leq & -\frac{\theta}{2}\lambda_{\min}\big[(C^+)^{\mathsf{T}}C^+\big]\|\tilde{y}\|^2 + \bigg\{\delta_M\|(C^+)^{\mathsf{T}}P\| \\ & + \Big(\frac{\kappa_1}{l_1} - \frac{\alpha^2}{4}\Big)\beta_1^2 + \Big(\frac{\kappa_2}{l_2} - 1\Big)\beta_2^2\bigg\}\|\tilde{y}\| \\ = & -\bigg(\frac{\theta}{2}\lambda_{\min}\big[(C^+)^{\mathsf{T}}C\big]\|\tilde{y}\| - \mathcal{B}\bigg)\|\tilde{y}\| \end{aligned} \tag{14}$$

where

$$\mathcal{B} = \delta_M\|(C^+)^{\mathsf{T}}P\| + \Big(\frac{\kappa_1}{l_1} - \frac{\alpha^2}{4}\Big)\beta_1^2 + \Big(\frac{\kappa_2}{l_2} - 1\Big)\beta_2^2. \tag{15}$$

Consequently, $\dot{J}(t)$ is negative as long as

$$\|\tilde{y}\| > \frac{2\mathcal{B}}{\theta\lambda_{\min}\big[(C^+)^{\mathsf{T}}C^+\big]} \tag{16}$$

where $\mathcal{B}$ is defined as in (15). Note that $\|\tilde{y}\| \leq \|C\|\|\tilde{x}\|$. Then, (16) is developed as

$$\|\tilde{x}\| > \frac{2\mathcal{B}}{\theta\|C\|\lambda_{\min}\big[(C^+)^{\mathsf{T}}C^+\big]}.$$

Hence, $\tilde{x}(t)$ converges to the compact set $\Omega_{\tilde{x}}$ defined as in (10). According to the standard Lyapunov extension theorem [21], this verifies the uniform ultimate boundedness of the observer NN weight estimation errors $\tilde{W}_1$ and $\tilde{V}_1$. ■

Remark 1: Noticing $\mathrm{rank}(C) = \mathrm{rank}(C^+)$, $\mathrm{rank}(C^+) = \mathrm{rank}[(C^+)^{\mathsf{T}}C^+]$ and using Assumption 2, we can obtain that $\mathrm{rank}[(C^+)^{\mathsf{T}}C^+] = p$. Observe that $[(C^+)^{\mathsf{T}}C^+] \in \mathbb{R}^{p\times p}$ is semipositive definite. Therefore, we derive that $(C^+)^{\mathsf{T}}C^+$ is positive definite. Then, we get $\lambda_{\min}[(C^+)^{\mathsf{T}}C^+] > 0$. This shows that $\Omega_{\tilde{x}}$ defined as in (10) makes sense.

B. Hamilton-Jacobi-Bellman Equation

In what follows we replace system (1) with (6), since system (1) can be approximated by (6) outside of $\Omega_{\tilde{x}}$. Meanwhile, we replace the actual state $x(t)$ with the estimated state $\hat{x}(t)$ due to the non-availability of $x(t)$. Then, system (1) can be represented as

$$\dot{\hat{x}}(t) = h(\hat{x}) + g(\hat{x})u \tag{17}$$

where $h(\hat{x}) = A\hat{x} + \hat{W}_1^{\mathsf{T}}\sigma(\hat{V}_1^{\mathsf{T}}\hat{x}) + B(y - C\hat{x})$. The value function (2) is rewritten as

$$V(\hat{x}(t)) = \int_t^\infty r\big(\hat{x}(s), u(s)\big)\mathrm{d}s. \tag{18}$$

where $r(\hat{x}, u) = Q_c(\hat{x}) + u^{\mathsf{T}}Ru$ with $Q_c(\hat{x}) = \hat{x}^{\mathsf{T}}C^{\mathsf{T}}QC\hat{x}$. If the control $u(\hat{x})$ is admissible and the value function $V(\hat{x}) \in \mathrm{C}^1(\Omega)$, then we have

$$V_{\hat{x}}^{\mathsf{T}}\big[h(\hat{x}) + g(\hat{x})u\big] + Q_c(\hat{x}) + u^{\mathsf{T}}Ru = 0$$

where $V_{\hat{x}} \in \mathbb{R}^n$ represents the partial derivative of $V(\hat{x})$ with respect to $\hat{x}$.

Define the Hamiltonian for the control $u(\hat{x})$ and the value function $V(\hat{x})$ as

$$H(\hat{x}, V_{\hat{x}}, u) = V_{\hat{x}}^{\mathsf{T}}\big[h(\hat{x}) + g(\hat{x})u\big] + Q_c(\hat{x}) + u^{\mathsf{T}}Ru.$$

Then, the optimal value $V^*(\hat{x})$ is obtained by solving the HJB equation

$$\min_{u(\hat{x})} H(\hat{x}, V_{\hat{x}}^*, u) = 0. \tag{19}$$

Accordingly, the closed-form expression for optimal control is derived as

$$u^*(\hat{x}) = -\frac{1}{2}R^{-1}g^{\mathsf{T}}(\hat{x})V_{\hat{x}}^*. \tag{20}$$

Substituting (20) into (19), we get the HJB equation as

$$V_{\hat{x}}^{*\mathsf{T}}h(\hat{x}) + Q_c(\hat{x}) - \frac{1}{4}V_x^{*\mathsf{T}}\mathfrak{A}(\hat{x})V_{\hat{x}}^* = 0 \tag{21}$$

where $\mathfrak{A}(\hat{x}) = g(\hat{x})R^{-1}g^{\mathsf{T}}(\hat{x})$.

Actually, (21) is difficult to solve by analytical methods. In what follows, we develop an online NN-based control scheme to derive the optimal control. Before presenting the control scheme, we provide a required assumption as follows.

Assumption 5: Assume that $L_1(\hat{x})$ is a continuously differentiable Lyapunov function candidate for system (17) and satisfies that $\dot{L}_1(\hat{x}) = L_{1\hat{x}}^{\mathsf{T}}\big(h(\hat{x}) + g(\hat{x})u^*\big) < 0$ with $L_{1\hat{x}}$ the partial derivative of $L_1(\hat{x})$ with respect to $\hat{x}$. Meanwhile,

there exists a positive definite matrix $\Lambda(\hat{x}) \in \mathbb{R}^{n\times n}$ defined on Ω such that

$$L_{1\hat{x}}^{\mathsf{T}}\big(h(\hat{x}) + g(\hat{x})u^*\big) = -L_{1\hat{x}}^{\mathsf{T}}\Lambda(\hat{x})L_{1\hat{x}}. \tag{22}$$

Remark 2: It should be emphasized that $h(\hat{x}) + g(\hat{x})u^*$ is often assumed to be bounded by a positive constant [13], [15], [16], i.e., there exists a constant $\rho > 0$ such that $\|h(\hat{x}) + g(\hat{x})u^*\| \leq \rho$. In order to relax the condition, in this paper, $h(\hat{x}) + g(\hat{x})u^*$ is bounded by a function with respect to x. Since $L_{1\hat{x}}$ is a function with respect to $\hat{x}$, without loss of generality, we assume that $\|h(\hat{x}) + g(\hat{x})u^*\| \leq \varrho\|L_{1\hat{x}}\|(\varrho > 0)$. In this sense, one can derive that $\|L_{1\hat{x}}^{\mathsf{T}}\big(h(\hat{x}) + g(\hat{x})u^*\big)\| \leq \varrho\|L_{1\hat{x}}\|^2$. Noticing $L_{1\hat{x}}^{\mathsf{T}}\big(h(\hat{x}) + g(\hat{x})u^*\big) < 0$, one shall find that (22) defined as in Assumption 5 is reasonable. In addition, $L_1(\hat{x})$ can be derived through proper selecting functions, such as polynomials.

C. Online Neuro-Optimal Control Scheme

In this subsection, an online optimal control scheme is constructed by using a unique critic NN. According to [20], the optimal value $V^*(\hat{x})$ can be represented as

$$V^*(\hat{x}) = W_c^{\mathsf{T}}\sigma(\nu_c^{\mathsf{T}}\hat{x}) + \varepsilon_2(\hat{x})$$

where $\nu_c \in \mathbb{R}^{n\times N}$ and $W_c \in \mathbb{R}^N$ denotes the weights for the input layer to the hidden layer and the hidden layer to the output layer respectively, N is the number of neurons in the hidden layer, and $\varepsilon_2(\hat{x})$ is the NN function reconstruction error. The derivative of $V^*(\hat{x})$ with respect to $\hat{x}$ is given by

$$V_{\hat{x}}^* = \nabla\sigma^{\mathsf{T}}(\hat{x})W_c + \nabla\varepsilon_2 \tag{23}$$

where $\nabla\sigma(\hat{x}) = \partial\sigma(\hat{x})/\partial\hat{x}$ and $\nabla\sigma(0) = 0$.

By utilizing (23), (20) can be represented as

$$u^*(\hat{x}) = -\frac{1}{2}R^{-1}g^{\mathsf{T}}(\hat{x})\nabla\sigma^{\mathsf{T}}W_c + \varepsilon_{u^*} \tag{24}$$

where $\varepsilon_{u^*} = -\frac{1}{2}R^{-1}g^{\mathsf{T}}(\hat{x})\nabla\varepsilon_2$.

Similarly, (21) can be rewritten as

$$\begin{aligned} W_c^{\mathsf{T}}\nabla\sigma h(\hat{x}) + Q_c(\hat{x}) + \varepsilon_{\mathrm{HJB}} \\ - \frac{1}{4}W_c^{\mathsf{T}}\nabla\sigma\mathfrak{A}(\hat{x})\nabla\sigma^{\mathsf{T}}W_c = 0 \end{aligned} \tag{25}$$

where $\varepsilon_{\mathrm{HJB}}$ is the residual error converging to zero as long as the number of NN nodes is large enough. That is, there exists $\varepsilon_a > 0$ such that $\|\varepsilon_{\mathrm{HJB}}\| \leq \varepsilon_a$.

Since the ideal critic NN weight W_c is unavailable, the control $u^*(\hat{x})$ in (24) cannot be implemented. Consequently, we use $\hat{V}(\hat{x})$ to approximate the value function in (18) as

$$\hat{V}(\hat{x}) = \hat{W}_c^{\mathsf{T}}\sigma(\hat{x}) \tag{26}$$

where $\hat{W}_c$ is the estimated weight of W_c. Define the estimation error for the critic NN as

$$\tilde{W}_c = W_c - \hat{W}_c. \tag{27}$$

By utilizing (26), the estimates of (20) is given by

$$\hat{u}(\hat{x}) = -\frac{1}{2}R^{-1}g^{\mathsf{T}}(\hat{x})\nabla\sigma^{\mathsf{T}}\hat{W}_c. \tag{28}$$

The approximated Hamiltonian is derived as

$$\begin{aligned} H(\hat{x}, \hat{W}_c) = \hat{W}_c^{\mathsf{T}}\nabla\sigma h(\hat{x}) + Q_c(\hat{x}) \\ - \frac{1}{4}\hat{W}_c^{\mathsf{T}}\nabla\sigma\mathfrak{A}(\hat{x})\nabla\sigma^{\mathsf{T}}\hat{W}_c \triangleq e. \end{aligned} \tag{29}$$

Combining (24), (25), and (29), we have

$$\begin{aligned} e = &-\tilde{W}_c^{\mathsf{T}}\nabla\sigma\Big(\mathfrak{C}(\hat{x}) + \frac{1}{2}\mathfrak{A}(\hat{x})\nabla\varepsilon_2\Big) \\ &- \frac{1}{4}\tilde{W}_c^{\mathsf{T}}\nabla\sigma\mathfrak{A}(\hat{x})\nabla\sigma^{\mathsf{T}}\tilde{W}_c - \varepsilon_{\mathrm{HJB}} \end{aligned} \tag{30}$$

with $\mathfrak{C}(\hat{x}) = h(\hat{x}) + g(\hat{x})u^*$.

In order to get the minimum value of e, it is desired to choose $\hat{W}_c$ to minimize the squared residual error $E = \frac{1}{2}e^{\mathsf{T}}e$. By using the gradient descent algorithm, the weight tuning law for the critic NN is generally given as [12], [15], [16]

$$\dot{\hat{W}}_c = -\eta\frac{\partial E}{\partial\hat{W}_c} = -\eta\frac{\phi}{(1+\phi^{\mathsf{T}}\phi)^2}e \tag{31}$$

where $\phi = \nabla\sigma\big[h(\hat{x}) + g(\hat{x})\hat{u}\big]$, $\eta > 0$ is a design parameter, and the term $\phi/(1+\phi^{\mathsf{T}}\phi)^2$ is employed for normalization.

However, there are two issues about the tuning rule (31):

1. Tuning the critic NN weights to minimize E alone cannot guarantee the stability of system (17) during the learning process of NNs.
2. The signal $\phi/(1+\phi^{\mathsf{T}}\phi)$ is required to be PE for guaranteeing the weights of the critic NN exponential converges to the actual optimal values. Nevertheless, the PE condition is intractable to verify due to the presence of hidden-layers involving in $\phi/(1+\phi^{\mathsf{T}}\phi)$.

For the sake of addressing above issues, a novel weight update law for the critic NN is developed as

$$\begin{aligned} \dot{\hat{W}}_c = &-\eta\bar{\phi}\Big(Y(\hat{x}) - \frac{1}{4}\hat{W}_c^{\mathsf{T}}\nabla\sigma\mathfrak{A}(\hat{x})\nabla\sigma^{\mathsf{T}}\hat{W}_c\Big) \\ &- \eta\sum_{j=1}^{N}\bar{\phi}_{(j)}\Big(Y(\hat{x}_{t_j}) - \frac{1}{4}\hat{W}_c^{\mathsf{T}}\nabla\sigma_{(j)}\mathfrak{A}(\hat{x}_{t_j})\nabla\sigma_{(j)}^{\mathsf{T}}\hat{W}_c\Big) \\ &+ \frac{\eta}{2}\Pi(\hat{x},\hat{u})\nabla\sigma\mathfrak{A}(\hat{x})L_{1\hat{x}} \end{aligned} \tag{32}$$

where $Y(\hat{x}) = \hat{W}_c^{\mathsf{T}}\nabla\sigma h(\hat{x}) + Q_c(\hat{x})$, $\bar{\phi} = \phi/m_s^2$, $m_s = 1+\phi^{\mathsf{T}}\phi$, $j \in \{1,\ldots,N\}$ denotes the index of a stored data point $\hat{x}(t_j)$ (for convenience, written as $\hat{x}_{t_j}$), $\bar{\phi}_{(j)} = \bar{\phi}(\hat{x}_{t_j})$, $m_{s_j} = 1+\phi^{\mathsf{T}}(\hat{x}_{t_j})\phi(\hat{x}_{t_j})$, $\nabla\sigma_{(j)} = \nabla\sigma(\hat{x}_{t_j})$, $L_{1\hat{x}}$ is defined as in Assumption 5, and $\Pi(\hat{x},\hat{u})$ is defined as

$$\Pi(\hat{x},\hat{u}) = \begin{cases} 0, \text{ if } L_{1\hat{x}}^{\mathsf{T}}\Big(h(\hat{x}) - \frac{1}{2}\mathfrak{A}(\hat{x})\nabla\sigma^{\mathsf{T}}\hat{W}_c\Big) < 0 \\ 1, \text{ otherwise.} \end{cases} \tag{33}$$

Remark 3: If there is no the second term in (32), one shall find $\dot{\hat{W}}_c = 0$ when there exists $\hat{x} = 0$. That is, the weights of the critic NN will not be updated. Under this circumstance, the critic NN might not be convergent. Hence, PE of the input signal is required. Nevertheless, by using (32), the PE condition is relaxed as long as $\{\bar{\phi}_{(j)}\}_1^N$ is selected to be linearly independent. Now we show this fact as follows:

Suppose that $\dot{\hat{W}}_c = 0$ when there exists $\hat{x} = 0$. From

(32), we can obtain that $\sum_{j=1}^{N}\bar{\phi}_{(j)}e_j = 0$, where $e_j = Y(\hat{x}_{t_j}) - \hat{W}_c^{\mathsf{T}}\nabla\sigma_{(j)}\mathfrak{A}(\hat{x}_{t_j})\nabla\sigma_{(j)}^{\mathsf{T}}\hat{W}_c/4$. Since $\{\bar{\phi}_{(j)}\}_1^N$ is linearly independent, we can conclude $e_j = 0$ $(j = 1, \ldots, N)$. However, this case will not happen until the system state stays at the equilibrium point. In other words, there exists at least $j_0 \in \{1, \ldots, N\}$ such that $e_{j_0} \neq 0$ during the learning process of NNs. Accordingly, we can draw the conclusion that the second term in (32) guarantees $\dot{\hat{W}}_c \neq 0$ during the learning process of NNs. That is, the PE condition is removed.

In order to guarantee the linear independence of $\{\bar{\phi}_{(j)}\}_1^N$, the following condition should be satisfied.

Condition 1: Let $\mathfrak{D} = \big[\sigma(\hat{x}_{t_1}), \ldots, \sigma(\hat{x}_{t_N})\big] \in \mathbb{R}^{N\times N}$ be the recorded data matrix. There exists sufficient large number of recorded data such that $\mathfrak{D}$ is nonsingular, that is, $\|\mathfrak{D}\| \neq 0$.

Remark 4: Condition 1 is first introduced in [22], which is used to derive adaptive control for tracking problems. *Condition 1* can be satisfied by selecting and recording data during the learning process of NNs over a finite time interval. Compared with the PE condition, a clear advantage of *Condition 1* is that it can be easily checked online.

By the definition of ϕ in (31) and using (24) , we have

$$\phi = \nabla\sigma\Big(\mathfrak{C}(\hat{x}) + \frac{1}{2}\mathfrak{A}(\hat{x})\nabla\varepsilon_2\Big) + \frac{1}{2}\nabla\sigma\mathfrak{A}(\hat{x})\nabla\sigma^{\mathsf{T}}\tilde{W}_c \quad (34)$$

with $\mathfrak{C}(\hat{x})$ defined as in (30). From (27), (30), (32), and (34), we derive

$$\begin{aligned}\dot{\tilde{W}}_c =& -\frac{\eta}{m_s^2}\Big(\nabla\sigma\mathfrak{L}(\hat{x}) + \frac{1}{2}\bar{\mathfrak{A}}(\hat{x})\tilde{W}_c\Big)\\ &\times\Big(\tilde{W}_c^{\mathsf{T}}\nabla\sigma\mathfrak{L}(\hat{x}) + \frac{1}{4}\tilde{W}_c^{\mathsf{T}}\bar{\mathfrak{A}}(\hat{x})\tilde{W}_c + \varepsilon_{\mathrm{HJB}}\Big)\\ &-\sum_{j=1}^{N}\frac{\eta}{m_{s_j}^2}\Big(\nabla\sigma_{(j)}\mathfrak{L}(\hat{x}_{t_j}) + \frac{1}{2}\bar{\mathfrak{A}}(\hat{x}_{t_j})\tilde{W}_c\Big)\\ &\times\Big(\tilde{W}_c^{\mathsf{T}}\nabla\sigma_{(j)}\mathfrak{L}(\hat{x}_{t_j}) + \frac{1}{4}\tilde{W}_c^{\mathsf{T}}\bar{\mathfrak{A}}(\hat{x}_{t_j})\tilde{W}_c + \varepsilon_{\mathrm{HJB}}\Big)\\ &-\frac{\eta}{2}\Pi(\hat{x},\hat{u})\nabla\sigma\mathfrak{A}(\hat{x})L_{1\hat{x}} \end{aligned} \quad (35)$$

where $\mathfrak{L}(\hat{x}) = \mathfrak{C}(\hat{x}) + \frac{1}{2}\mathfrak{A}(\hat{x})\nabla\varepsilon_2$, $\bar{\mathfrak{A}}(\hat{x}) = \nabla\sigma\mathfrak{A}(\hat{x})\nabla\sigma^{\mathsf{T}}$, and $\bar{\mathfrak{A}}(\hat{x}_{t_j}) = \nabla\sigma_{(j)}\mathfrak{A}(\hat{x}_{t_j})\nabla\sigma_{(j)}^{\mathsf{T}}$.

IV. Stability Analysis

In this section, we present our main results based on Lyapunov's direct method. Prior to giving the main theorem, we provide another assumption as follows:

Assumption 6: The derivative of $\sigma(\hat{x})$ with respect to $\hat{x}$ is bounded, that is, there exists $b_\sigma > 0$ such that $\|\nabla\sigma(\hat{x})\| < b_\sigma$. The derivative of the NN reconstruction error $\varepsilon_2(\hat{x})$ with respect to $\hat{x}$ is bounded as $\|\nabla\varepsilon_2(\hat{x})\| < \varepsilon_b$, where $\varepsilon_b > 0$.

With Assumptions 1–6 and Facts 1–2, our main theorem is developed as follows:

Theorem 2: Consider system (1) with the associated HJB equation (21). Let Assumptions 1–6 be satisfied and take the control input for system (1) as in (28). Meanwhile, let weight update laws for the observer NN be (8) and (9), and let weight tuning rule for the critic NN be (32). Then, the state observer error $\tilde{x}(t)$, NN weight estimation errors $\tilde{W}_1$, $\tilde{V}_1$, and $\tilde{W}_c$ are all guaranteed to be UUB.

Proof: We provide an outline of the proof due to the space limit. Consider the Lyapunov function candidate

$$L(t) = L_1(x(t)) + L_2(t) + \frac{1}{2}\tilde{W}_c^{\mathsf{T}}\eta^{-1}\tilde{W}_c \quad (36)$$

where $L_1(x(t))$ is defined as in Assumption 5, $L_2(t) = J(t)$ with $J(t)$ defined as in Theorem 1.

Taking the time derivative of (36) and by using Theorem 1, we derive

$$\begin{aligned}\dot{L}(t) \leq\ & L_{1\hat{x}}^{\mathsf{T}}\Big(h(\hat{x}) - \frac{1}{2}\mathfrak{A}(\hat{x})\nabla\sigma^{\mathsf{T}}\hat{W}_c\Big)\\ &-\frac{\theta}{2}\lambda_{\min}\big[(C^+)^{\mathsf{T}}C^+\big]\|C\tilde{x}\|^2 + \mathcal{B}\|C\tilde{x}\|\\ &+\tilde{W}_c^{\mathsf{T}}\eta^{-1}\dot{\tilde{W}}_c \end{aligned} \quad (37)$$

where $\mathcal{B}$ is defined as in (15).

By utilizing (35), we derive the last term in (37) as

$$\tilde{W}_c^{\mathsf{T}}\eta^{-1}\dot{\tilde{W}}_c = \mathfrak{F}_1 + \mathfrak{F}_2 - \frac{1}{2}\tilde{W}_c^{\mathsf{T}}\Pi(\hat{x},\hat{u})\nabla\sigma\mathfrak{A}(\hat{x})L_{1\hat{x}} \quad (38)$$

where

$$\begin{aligned}\mathfrak{F}_1 =& -\frac{1}{m_s^2}\Big(\tilde{W}_c^{\mathsf{T}}\nabla\sigma\mathfrak{L}(\hat{x}) + \frac{1}{2}\tilde{W}_c^{\mathsf{T}}\bar{\mathfrak{A}}(\hat{x})\tilde{W}_c\Big)\\ &\times\Big(\tilde{W}_c^{\mathsf{T}}\nabla\sigma\mathfrak{L}(\hat{x}) + \frac{1}{4}\tilde{W}_c^{\mathsf{T}}\bar{\mathfrak{A}}(\hat{x})\tilde{W}_c + \varepsilon_{\mathrm{HJB}}\Big)\\ \mathfrak{F}_2 =& -\sum_{j=1}^{N}\frac{1}{m_{s_j}^2}\Big(\tilde{W}_c^{\mathsf{T}}\nabla\sigma_{(j)}\mathfrak{L}(\hat{x}_{t_j}) + \frac{1}{2}\tilde{W}_c^{\mathsf{T}}\bar{\mathfrak{A}}(\hat{x}_{t_j})\tilde{W}_c\Big)\\ &\times\Big(\tilde{W}_c^{\mathsf{T}}\nabla\sigma_{(j)}\mathfrak{L}(\hat{x}_{t_j}) + \frac{1}{4}\tilde{W}_c^{\mathsf{T}}\bar{\mathfrak{A}}(\hat{x}_{t_j})\tilde{W}_c + \varepsilon_{\mathrm{HJB}}\Big).\end{aligned}$$

Note that $\mathfrak{F}_1$ and $\mathfrak{F}_2$ in (38) can be developed as

$$\begin{aligned}\mathfrak{F}_1 \leq& -\frac{1}{m_s^2}\Big\{\frac{1}{16}\Big(\tilde{W}_c^{\mathsf{T}}\bar{\mathfrak{A}}(\hat{x})\tilde{W}_c\Big)^2 - 4\Big(\tilde{W}_c^{\mathsf{T}}\nabla\sigma\mathfrak{L}(\hat{x})\Big)^2\\ &-\frac{5}{2}\varepsilon_{\mathrm{HJB}}^2\Big\}.\\ \mathfrak{F}_2 \leq& -\sum_{j=1}^{N}\frac{1}{m_{s_j}^2}\Big\{\frac{1}{16}\Big(\tilde{W}_c^{\mathsf{T}}\bar{\mathfrak{A}}(\hat{x}_{t_j})\tilde{W}_c\Big)^2\\ &-4\Big(\tilde{W}_c^{\mathsf{T}}\nabla\sigma_{(j)}\mathfrak{L}(\hat{x}_{t_j})\Big)^2 - \frac{5}{2}\varepsilon_{\mathrm{HJB}}^2\Big\}.\end{aligned} \quad (39)$$

Substituting (39) into (38), and noting that $1 \leq m_s^2 \leq 4$, $1 \leq m_{s_j}^2 \leq 4$, we have

$$\begin{aligned}\tilde{W}_c^{\mathsf{T}}\eta^{-1}\dot{\tilde{W}}_c \leq& -\frac{1}{64}\Big\{\sum_{j=1}^{N}\mu_{\inf}^2\Big(\bar{\mathfrak{A}}(\hat{x}_{t_j})\Big) + \mu_{\inf}^2\Big(\bar{\mathfrak{A}}(\hat{x})\Big)\Big\}\\ &\times\big\|\tilde{W}_c\big\|^4 + 4b_\sigma^2\Big\{\sum_{j=1}^{N}\vartheta_{\sup}^2\Big(\mathfrak{L}(\hat{x}_{t_j})\Big)\\ &+\vartheta_{\sup}^2\Big(\mathfrak{L}(\hat{x})\Big)\Big\}\big\|\tilde{W}_c\big\|^2 + \frac{5}{2}(N+1)\varepsilon_a^2\\ &-\frac{1}{2}\tilde{W}_c^{\mathsf{T}}\Pi(\hat{x},\hat{u})\nabla\sigma\mathfrak{A}(\hat{x})L_{1\hat{x}}\end{aligned} \quad (40)$$

where $\mu_{\inf}(\mathcal{Y})$ denotes the lower bound of $\mathcal{Y}$ $\big(\mathcal{Y} = \bar{\mathfrak{A}}(\hat{x}), \bar{\mathfrak{A}}(\hat{x}_{t_j})\big)$, and $\vartheta_{\sup}(\mathcal{Z})$ represents the upper bound of $\mathcal{Z}$ $(\mathcal{Z} = \mathfrak{L}(\hat{x}), \mathfrak{L}(\hat{x}_{t_j}))$, and N is the number of neurons in the hidden-layer.

Combining (37) and (40), we obtain

$$\begin{aligned}\dot{L}(t) \leq\ & L_{1\hat{x}}^{\mathsf{T}}\Big(h(\hat{x}) - \frac{1}{2}\mathfrak{A}(\hat{x})\nabla\sigma^{\mathsf{T}}\hat{W}_c\Big) \\ & - \frac{1}{2}\tilde{W}_c^{\mathsf{T}}\Pi(\hat{x},\hat{u})\nabla\sigma\mathfrak{A}(\hat{x})L_{1\hat{x}} \\ & - \frac{\mathfrak{T}_1}{64}\|\tilde{W}_c\|^4 + 4\mathfrak{T}_2\|\tilde{W}_c\|^2 \\ & - \frac{\gamma}{2}\Big(\|C\tilde{x}\| - \mathcal{B}/\gamma\Big)^2 + \frac{\mathcal{B}^2}{2\gamma} \\ & + \frac{5}{2}(N+1)\varepsilon_a^2 \end{aligned} \tag{41}$$

where

$$\begin{aligned}\mathfrak{T}_1 &= \mu_{\inf}^2\Big(\bar{\mathfrak{A}}(\hat{x})\Big) + \sum_{j=1}^{N}\mu_{\inf}^2\Big(\bar{\mathfrak{A}}(\hat{x}_{t_j})\Big) \\ \mathfrak{T}_2 &= b_\sigma^2\vartheta_{\sup}^2\Big(\mathfrak{L}(\hat{x})\Big) + b_\sigma^2\sum_{j=1}^{N}\vartheta_{\sup}^2\Big(\mathfrak{L}(\hat{x}_{t_j})\Big) \\ \gamma &= \theta\lambda_{\min}\big[(C^+)^{\mathsf{T}}C^+\big].\end{aligned}$$

Case I: $\Pi(\hat{x},\hat{u}) = 0$. In this sense, we derive that the first term in (41) is negative by using the definition of $\Pi(\hat{x},\hat{u})$ in (33). Noting that $L_{1\hat{x}}^{\mathsf{T}}\dot{\hat{x}} < 0$, based on Archimedean property of $\mathbb{R}$, one can conclude that there exists a constant $\tau > 0$ such that $-\|L_{1\hat{x}}\|\tau \geq L_{1\hat{x}}^{\mathsf{T}}\dot{\hat{x}}$. Then, (41) is developed as

$$\begin{aligned}\dot{L}(t) \leq & -\tau\|L_{1\hat{x}}\| - \frac{\gamma}{2}\Big(\|C\tilde{x}\| - \mathcal{B}/\gamma\Big)^2 \\ & - \frac{\mathfrak{T}_1}{64}\Big(\|\tilde{W}_c\|^2 - \frac{128\mathfrak{T}_2}{\mathfrak{T}_1}\Big)^2 + \frac{256\mathfrak{T}_2^2}{\mathfrak{T}_1} \\ & + \frac{1}{2\gamma}\big[\mathcal{B}^2 + 5\gamma(N+1)\varepsilon_a^2\big].\end{aligned} \tag{42}$$

Therefore, (42) yields $\dot{L}(t) < 0$ as long as one of the following conditions holds:

$$\|L_{1\hat{x}}\| > \frac{256\mathfrak{T}_2^2}{\tau\mathfrak{T}_1} + \frac{\mathcal{B}^2 + 5\gamma(N+1)\varepsilon_a^2}{2\tau\gamma}$$

or

$$\|\tilde{x}\| > \frac{1}{\|C\|}\sqrt{\frac{512\mathfrak{T}_2^2}{\gamma\mathfrak{T}_1} + \frac{\mathcal{B}^2 + 5\gamma(N+1)\varepsilon_a^2}{\gamma^2}} + \frac{\mathcal{B}}{\gamma\|C\|}$$

or

$$\|\tilde{W}_c\| > 2\sqrt{\frac{32\mathfrak{T}_2}{\mathfrak{T}_1} + \frac{\sqrt{2\mathfrak{T}_1\big[\mathcal{B}^2/\gamma + 5(N+1)\varepsilon_a^2\big] + 1024\mathfrak{T}_2^2}}{\mathfrak{T}_1}}.$$

Case II: $\Pi(\hat{x},\hat{u}) = 1$. By the definition of $\Pi(\hat{x},\hat{u})$ in (33), we find that, in this case, the first term in (41) is nonnegative which implies that the control defined as in (28) may not stabilize system (17). Then, (41) becomes

$$\begin{aligned}\dot{L}(t) \leq\ & L_{1\hat{x}}^{\mathsf{T}}\Big(\mathfrak{C}(\hat{x}) + \frac{1}{2}\mathfrak{A}(\hat{x})\nabla\varepsilon_2\Big) \\ & - \frac{\gamma}{2}\Big(\|C\tilde{x}\| - \mathcal{B}/\gamma\Big)^2 + \frac{\mathcal{B}^2}{2\gamma} \\ & - \frac{\mathfrak{T}_1}{64}\Big(\|\tilde{W}_c\|^2 - \frac{128\mathfrak{T}_2}{\mathfrak{T}_1}\Big)^2 \\ & + \frac{256\mathfrak{T}_2^2}{\mathfrak{T}_1} + \frac{5}{2}(N+1)\varepsilon_a^2\end{aligned} \tag{43}$$

where $\mathfrak{C}(\hat{x})$ is defined as in (30).

By using (22) and Assumption 6, (43) is developed as

$$\begin{aligned}\dot{L}(t) \leq & -\lambda_{\min}\big(\Lambda(\hat{x})\big)\left(\|L_{1\hat{x}}\| - \frac{\varepsilon_b\vartheta_{\sup}\big(\mathfrak{A}(\hat{x})\big)}{4\lambda_{\min}\big(\Lambda(\hat{x})\big)}\right)^2 \\ & - \frac{\gamma}{2}\Big(\|C\tilde{x}\| - \mathcal{B}/\gamma\Big)^2 - \frac{\mathfrak{T}_1}{64}\Big(\|\tilde{W}_c\|^2 - \frac{128\mathfrak{T}_2}{\mathfrak{T}_1}\Big)^2 \\ & + \frac{\varepsilon_b\vartheta_{\sup}\big(\mathfrak{A}(\hat{x})\big)}{16\lambda_{\min}\big(\Lambda(\hat{x})\big)} + \frac{256\mathfrak{T}_2^2}{\mathfrak{T}_1} \\ & + \frac{1}{2\gamma}\big[\mathcal{B}^2 + 5\gamma(N+1)\varepsilon_a^2\big]\end{aligned} \tag{44}$$

where $\lambda_{\min}\big(\Lambda(\hat{x})\big)$ represents the minimum eigenvalue of $\Lambda(\hat{x})$, $\vartheta_{\sup}(\cdot)$ is defined as in (40).

Hence, we obtain that (44) implies that $\dot{L}(t) < 0$ as long as one of the following conditions holds:

$$\|\tilde{x}\| > \frac{1}{\|C\|}\sqrt{\frac{2d}{\gamma}} + \frac{\mathcal{B}}{\gamma\|C\|}$$

or

$$\|\tilde{W}_c\| > 2\sqrt{\frac{32\mathfrak{T}_2}{\mathfrak{T}_1} + 2\sqrt{\frac{d}{\mathfrak{T}_1}}}$$

or

$$\|L_{1\hat{x}}\| > \frac{\varepsilon_b\vartheta_{\sup}\big(\mathfrak{A}(\hat{x})\big)}{4\lambda_{\min}\big(\Lambda(\hat{x})\big)} + \sqrt{\frac{d}{\lambda_{\min}\big(\Lambda(\hat{x})\big)}}$$

where

$$d = \frac{\varepsilon_b\vartheta_{\sup}\big(\mathfrak{A}(\hat{x})\big)}{16\lambda_{\min}\big(\Lambda(\hat{x})\big)} + \frac{256\mathfrak{T}_2^2}{\mathfrak{T}_1} + \frac{1}{2\gamma}\big[\mathcal{B}^2 + 5\gamma(N+1)\varepsilon_a^2\big].$$

Combining *Cases I* and *II* and using the standard Lyapunov extension theorem [21], one can derive that the state observer error $\tilde{x}(t)$, the NN weight estimation errors $\tilde{W}_1$, $\tilde{V}_1$, and $\tilde{W}_c$ are all UUB. ■

V. Simulation Results

Consider the nonlinear CT system described by

$$\begin{aligned}\dot{x} &= f(x) + g(x)u \\ y &= Cx\end{aligned} \tag{45}$$

where

$$f(x) = \begin{bmatrix} -x_1 + x_2 \\ -0.5x_1 - 0.5x_2 + 0.5x_2\big(\cos(2x_1) + 2\big)^2 \end{bmatrix}$$

$$g(x) = \begin{bmatrix} 0 \\ \cos(2x_1) + 2 \end{bmatrix} \quad C = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

The value function is defined as in (2), where Q and R are chosen as identity matrices of approximate dimensions. The prior knowledge of system states is assumed to be unavailable, and only the output $y(t)$ is measurable in system (45). In order to obtain the knowledge of system dynamics, an NN state observer defined as in (6) is employed. The gains for the observer are selected as

$$A = [-1 \quad 1; -0.5 \quad -0.5], \quad B = [1 \quad 0; -0.5 \quad 0],$$
$$l_1 = 20, \quad l_2 = 10, \quad k_1 = 6.1, \quad k_2 = 15, \quad N_1 = 8,$$

and the gain for the critic NN is chosen as $\eta = 2.5$. The activation function for the critic NN is selected with $N = 3$ neurons as

$$\sigma(x) = [x_1^2 \quad x_2^2 \quad x_1 x_2]^{\mathsf{T}}$$

and the weight of the critic NN is denoted as $\hat{W}_c = [W_c^1 \quad W_c^2 \quad W_c^3]^{\mathsf{T}}$.

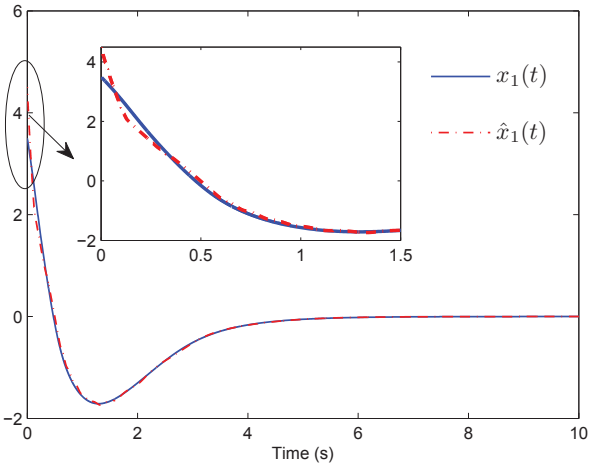

Fig. 1. Trajectories of real state $x_1(t)$ and observed state $\hat{x}_1(t)$

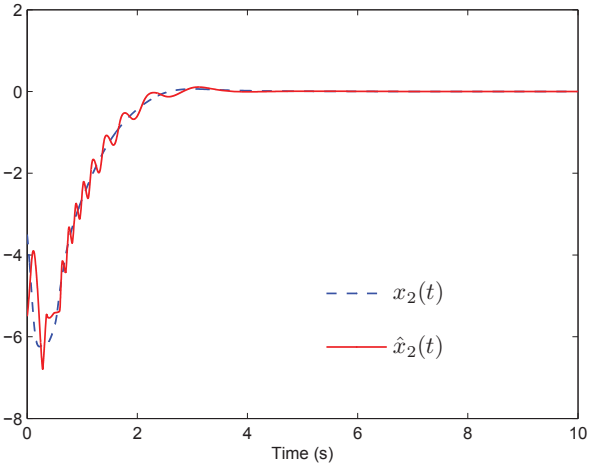

Fig. 2. Trajectories of real state $x_2(t)$ and observed state $\hat{x}_2(t)$

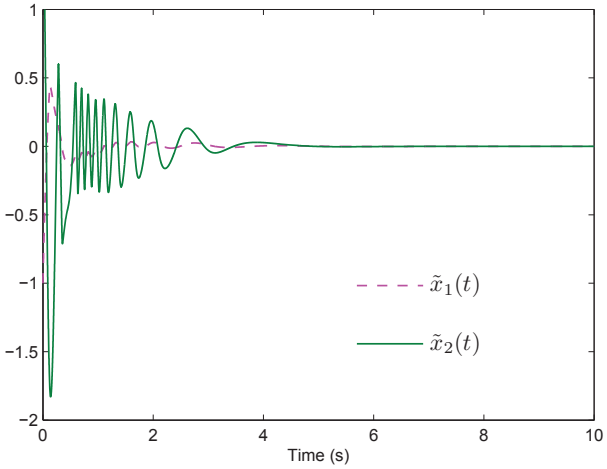

Fig. 3. NN observer errors $\tilde{x}_1(t)$ and $\tilde{x}_2(t)$

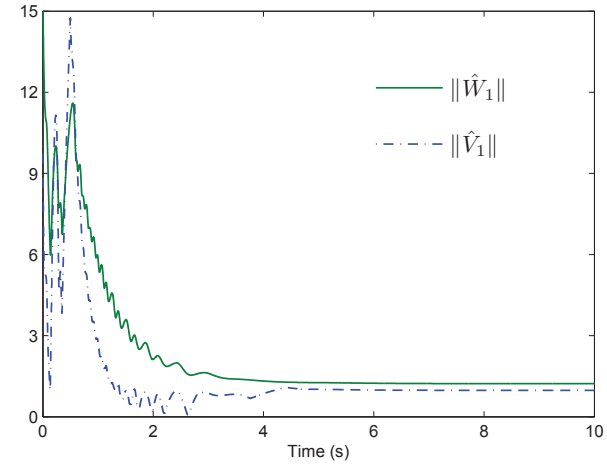

Fig. 4. 2-norm of observer NN weights $\|\hat{W}_1\|$ and $\|\hat{V}_1\|$

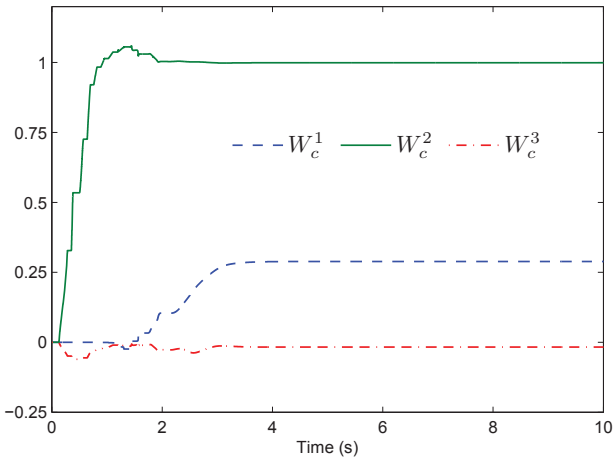

Fig. 5. Convergence of critic NN weight $\hat{W}_c$

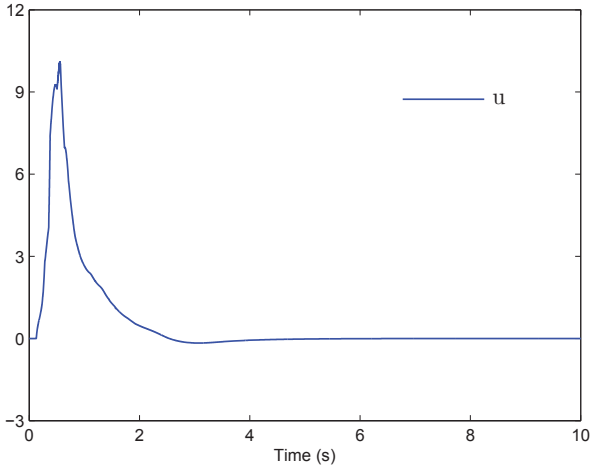

Fig. 6. Control input u

The initial weights $\hat{W}_1$ and $\hat{V}_1$ for the observer NN are selected randomly within an interval of $[-10, 10]$ and $[-5, 5]$, respectively. Meanwhile, the initial weights for the critic NN are chosen to be zeros, and the initial system state is selected to be $x_0 = [3.5, -3.5]^\mathsf{T}$. In this sense, one can find the initial control can not stabilize system (45). In other words, no initial stabilizing control is required for implementing the algorithm. In addition, by using the method proposed in [22], the recorded data can be easily made qualified for *Condition 1*. That is, the PE condition is removed.

The simulation results are presented in Figs. 1–6. Figs. 1 and 2 show the trajectories of system state $x_1(t)$ and observed state $\hat{x}_1(t)$, and the trajectories of system state $x_2(t)$ and observed state $\hat{x}_2(t)$, respectively. Fig. 3 illustrates the performance of the NN state observer errors $\tilde{x}_1(t)$ and $\tilde{x}_2(t)$. Fig. 4 indicates the 2-norm of the weights of the observer NN $\|\hat{W}_1\|$ and $\|\hat{V}_1\|$. Fig. 5 displays the performance of the convergence of the critic NN weights. Fig. 6 illustrates the optimal control u. From Figs. 1–3, it is observed that the NN observer can approximate the real system very fast and well. From Figs. 4 and 5, one can find that the observer NN and the critic NN are tuned simultaneously. Meanwhile, Fig. 5 indicates that no initial stabilizing control is required. Moreover, comparing Figs. 1–2 with simulation results in [15], one shall find that there is no probe noise added to get the PE signal. That is, the restrictive PE condition is relaxed. In addition, our algorithm ensures that the closed-loop system is stable in the sense of uniform ultimate boundedness and that learning is very fast.

VI. Conclusions

We have developed a new observer–critic architecture to derive the optimal control for uncertain nonlinear CT systems. Based on the present architecture, the observer NN and the critic NN are tuned simultaneously. Meanwhile, the restrictive conditions that the initial stabilizing control and PE are removed. In our future work, we shall focus on developing online algorithms for solving optimal control problems of nonaffine nonlinear CT systems.

References

[1] F. L. Lewis, D. Vrabie, and V. L. Syrmos, *Optimal Control (3rd edn.).* New Jersey: John Wiley & Sons, Inc., 2012.

[2] D. Liu and Q. Wei, "Finite-approximation-error-based optimal control approach for discrete-time nonlinear systems," *IEEE Trans. Cybern.*, vol. 43, no. 2, pp. 779–789, Apr. 2013.

[3] D. Liu, D. Wang, and X. Yang, "An iterative adaptive dynamic programming algorithm for optimal control of unknown discrete-time nonlinear systems with constrained inputs," *Inf. Sci.*, vol. 220, pp. 331–342, Jan. 2013.

[4] X. Yang, D. Liu, and Q. Wei, "Neuro-optimal control of unknown nonaffine nonlinear systems with saturating actuators," in *Proc. 3rd IFAC International Conference on Intelligent Control and Automation Science*, Chengdu, China, 2013, pp. 569–574.

[5] H. N. Wu and B. Luo, "Neural network based online simultaneous policy update algorithm for solving the HJI equation in nonlinear H^∞ control," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 12, pp. 1884–1895, Dec. 2012.

[6] R. E. Bellman, *Dynamic Programming.* New Jersey: Princeton University Press, 1957.

[7] P. J. Werbos, *Beyond Regression: New Tools for Prediction and Analysis in the Behavioral Sciences.* Ph.D. Dissertation, Harvard University, USA, 1974

[8] P. J. Werbos, "Approximate dynamic programming for real-time control and neural modeling," in *Handbook of Intelligent Control.* D. A. White and D. A. Sofge, Eds. Van Nostrand Reinhold, New York, 1992.

[9] R. S. Sutton and A. G. Barto, *Reinforcement Learning–An Introduction.* Cambridge, MA: MIT Press, 1998.

[10] F. L. Lewis, D. Vrabie, and K. G. Vamvoudakis, "Reinforcement learning and feedback control: Using natural decision methods to design optimal adaptive controllers," *IEEE Control Syst. Mag.*, vol. 32, no. 6, pp. 76–105, Nov. 2012.

[11] F. L. Lewis and D. Liu, *Reinforcement Learning and Approximate Dynamic Programming for Feedback Control.* Hoboken, New Jersey: Wiley, 2013.

[12] X. Yang, D. Liu, and D. Wang, "Reinforcement learning for adaptive optimal control of unknown continuous-time nonlinear systems with input constraints," *Int. J. Control*, vol. 87, no. 3, pp. 553–566, 2014.

[13] D. Liu, X. Yang, and H. Li, "Adaptive optimal control for a class of continuous-time affine nonlinear systems with unknown internal dynamics," *Neural Comput. Appl.*, vol. 23, no. 7–8, pp. 1843–1850, Dec. 2013.

[14] 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.

[15] K. G. Vamvoudakis and F. L. Lewis, "Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem," *Automatica*, vol. 46, no. 5, pp. 878–888, May 2010.

[16] S. Bhasin, R. Kamalapurkar, M. Johnson, K. G. Vamvoudakis, F. L. Lewis, and W. E. Dixon, "A novel actor-critic-identifier architecture for approximate optimal control of uncertain nonlinear systems," *Automatica*, vol. 49, no. 1, pp. 82–92, Jan. 2013.

[17] T. Dierks and S. Jagannathan, "Optimal control of affine nonlinear continuous-time systems," in *Amer. Control Conf.*, Baltimore, MD, USA, 2010, pp. 1568–1573.

[18] X. Yang, D. Liu, and Y. Huang, "Neural-network-based online optimal control for uncertain non-linear continuous-time systems with control constraints," *IET Contr. Theory Appl.*, vol. 7, no. 17, pp. 2037–2047, Nov. 2013.

[19] F. Abdollahi, H. A. Talebi, and R. V. Patel, "A stable neural network-based observer with application to flexible-joint manipulators," *IEEE Trans. Neural Netw.*, vol. 17, no. 1, pp. 118–129, Jan. 2006.

[20] 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.

[21] F. L. Lewis, S. Jagannathan, and A. Yesildirek, *Neural Network Control of Robot Manipulators and Nonlinear Systems.* London: Taylor & Francis, 1999.

[22] G. V. Chowdhary, *Concurrent Learning for Convergence in Adaptive Control without Persistency of Excitation.* Ph.D. Dissertation, Georgia Institute of Technology, USA, 2010.