Chapter 6
Optimal Feedback Control for Continuous-Time Systems via ADP

6.1 Introduction

In this chapter, we will study how to design a controller for continuous-time systems via the ADP method. Although many ADP methods have been proposed for continuous-time systems [1, 6, 9, 10, 12, 13, 15, 17–19], a suitable framework in which the optimal controller can be designed for a class of unknown general continuous-time systems is still not available. Therefore, in Sect. 6.2, we will develop a novel optimal robust feedback control scheme for a class of unknown general continuous-time systems using ADP method. The merit of present method is that we require only the availability of input/output data instead of exact system model. Moreover, the obtained control input can be guaranteed to be close to the optimal control input within a small bound.

As is known, in the real world, many practical control systems are described by nonaffine structure, such as chemical reactions, dynamic model in pendulum control, etc. The difficulty associated with ADP for nonaffine nonlinear system is that the nonlinear function is an implicit function with respect to the control variable. To overcome this difficulty, in Sect. 6.3, we will extend the ADP method to a class of nonaffine nonlinear systems. Through the present two methods, optimal control problems of a quite wide class of continuous-time nonlinear systems can be solved.

6.2 Optimal Robust Feedback Control for Unknown General Nonlinear Systems

In this section, a robust approximate optimal tracking control scheme is developed for a class of unknown general nonlinear systems by using the ADP method. In the design of the controller, only available input/output data are required instead of known system dynamics. First, a data-based model is established by a *recurrent neural network* (RNN) to reconstruct the unknown system dynamics using available input/output data. Then, based on the obtained data-based model, the ADP method is utilized to design the approximate optimal tracking controller.

H. Zhang et al., *Adaptive Dynamic Programming for Control*,
Communications and Control Engineering, DOI 10.1007/978-1-4471-4757-2_6,

6.2.1 Problem Formulation

Consider the following general continuous-time nonlinear systems:

$$\dot{x}(t) = f(x(t), u(t)), \tag{6.1}$$

where $x(t) = [x_1(t), x_2(t), \dots, x_n(t)]^{\mathrm{T}} \in \mathbb{R}^n$ is the state vector, $u(t) = [u_1(t), u_2(t), \dots, u_m(t)]^{\mathrm{T}} \in \mathbb{R}^m$ is the input vector, and $f(\cdot, \cdot)$ is an unknown smooth nonlinear function with respect to $x(t)$ and $u(t)$.

In this section, our control objective is to design an optimal controller for (6.1), which ensures that the state vector $x(t)$ tracks the specified trajectory $x_d(t)$ while minimizing the infinite-horizon cost functional as follows:

$$J(e(t), u) = \int_t^{\infty} l(e(\tau), u(\tau)) d\tau, \tag{6.2}$$

where $e(t) = x(t) - x_d(t)$ denotes the state tracking error, $l(e(t), u(t)) = e^{\mathrm{T}}(t) Q e(t) + u^{\mathrm{T}}(t) R u(t)$ is the utility function, and Q and R are symmetric positive definite matrices with appropriate dimensions.

Since the system dynamics is completely unknown, we cannot apply existing ADP methods to (6.1) directly. Therefore, it is now desirable to propose a novel control scheme that does not need the exact system dynamics but only the input/output data which can be obtained during the operation of the system. Therefore, we propose a *data-based* optimal robust tracking control scheme using ADP method for unknown general nonlinear continuous-time systems. Specifically, the design of the present controller is divided into two steps:

1. Establishing a data-based model based on an RNN by using available input/output data to reconstruct the unknown system dynamics, and
2. Designing the robust approximate optimal tracking controller based on the obtained data-based model

In the following, the establishment of the data-based model and the controller design will be discussed in detail.

6.2.2 Data-Based Robust Approximate Optimal Tracking Control

Although we cannot obtain the exact system model in general, fortunately, we can access input–output data of the unknown general nonlinear systems in many practical control processes. So it is desirable to use available input–output data in the design of the controller. The historical input–output data could be incorporated indirectly in the form of a data-based model. The data-based model could extract useful information contained in the input–output data and capture input–output mapping. Markov models, neural network models, well structured filters, wavelet models, and other function approximation models can be regarded as data-based models [14, 20–24, 26]. In this section, we develop a data-based model based on a recurrent neural network (RNN) to reconstruct the unknown system dynamics by using available input–output data.

To begin with the development, the system dynamics (6.1) is rewritten in the form of an RNN as follows [16]:

$$\dot{x}(t) = C^{*\mathrm{T}}x(t) + A^{*\mathrm{T}}h(x(t)) + C_u^{*\mathrm{T}}u(t) + A_u^{*\mathrm{T}} + \varepsilon_m(t), \tag{6.3}$$

where $\varepsilon_m(t)$ is assumed to be bounded, $C^{*\mathrm{T}}$, $A^{*\mathrm{T}}$, $C_u^{*\mathrm{T}}$, and $A_u^{*\mathrm{T}}$ are unknown ideal weight matrices. The activation function $h(\cdot)$ is selected as a monotonically increasing function and satisfies

$$0 \le h(x) - h(y) \le k(x-y), \tag{6.4}$$

for any $x, y \in \mathbb{R}$ and $x \ge y, k > 0$, such as $h(x) = \tanh(x)$.

Based on (6.3), the data-based model is then constructed as

$$\dot{\hat{x}}(t) = \hat{C}^{\mathrm{T}}(t)\hat{x}(t) + \hat{A}^{\mathrm{T}}(t)h(\hat{x}(t)) + \hat{C}_u^{\mathrm{T}}(t)u(t) + \hat{A}_u^{\mathrm{T}}(t) - \upsilon(t), \tag{6.5}$$

where $\hat{x}(t)$ is the estimated system state vector, $\hat{C}(t)$, $\hat{A}(t)$, $\hat{C}_u(t)$, and $\hat{A}_u(t)$ are the estimates of the ideal weight matrices C^*, A^*, C_u^*, and A_u^*, respectively, and $\upsilon(t)$ is defined as

$$\upsilon(t) = Se_m(t) + \frac{\hat{\theta}(t)e_m(t)}{e_m^{\mathrm{T}}(t)e_m(t) + \eta}, \tag{6.6}$$

where $e_m(t) = x(t) - \hat{x}(t)$ is the system modeling error, $S \in \mathbb{R}^{n\times n}$ is a design matrix, $\hat{\theta}(t) \in \mathbb{R}$ is an additional tunable parameter, and $\eta > 1$ is a constant.

Assumption 6.1 The term $\varepsilon_m(t)$ is assumed to be upper bounded by a function of modeling error such that

$$\varepsilon_m^{\mathrm{T}}(t)\varepsilon_m(t) \le \varepsilon_M(t) = \theta^* e_m^{\mathrm{T}}(t)e_m(t), \tag{6.7}$$

where θ^* is the bounded constant target value.

The modeling error dynamics is written as

$$\begin{aligned}\dot{e}_m(t) = {} & C^{*\mathrm{T}}e_m(t) + \tilde{C}^{\mathrm{T}}(t)\hat{x}(t) + A^{*\mathrm{T}}\tilde{h}(e_m(t)) \\ & + \tilde{A}^{\mathrm{T}}(t)h(\hat{x}(t)) + \tilde{C}_u^{\mathrm{T}}(t)u(t) + \tilde{A}_u^{\mathrm{T}}(t) + \varepsilon_a(t) \\ & + Se_m(t) - \frac{\tilde{\theta}(t)e_m(t)}{e_m^{\mathrm{T}}(t)e_m(t) + \eta} + \frac{\theta^* e_m(t)}{e_m^{\mathrm{T}}(t)e_m(t) + \eta},\end{aligned} \tag{6.8}$$

where $\tilde{C}(t) = C^* - \hat{C}(t)$, $\tilde{A}(t) = A^* - \hat{A}(t)$, $\tilde{C}_u(t) = C_u^* - \hat{C}_u(t)$, $\tilde{A}_u(t) = A_u^* - \hat{A}_u(t)$, $\tilde{h}(e_m(t)) = h(x(t)) - h(\hat{x}(t))$, and $\tilde{\theta}(t) = \theta^* - \hat{\theta}(t)$.

Theorem 6.2 (cf. [25]) *The modeling error $e_m(t)$ will be asymptotically convergent to zero as $t \to \infty$ if the weight matrices and the tunable parameter of the data-based model* (6.5) *are updated through the following equations*:

$$\begin{aligned}\dot{\hat{C}}(t) &= \Gamma_1 \hat{x}(t) e_m^{\mathrm{T}}(t),\\ \dot{\hat{A}}(t) &= \Gamma_2 f(\hat{x}(t)) e_m^{\mathrm{T}}(t),\\ \dot{\hat{C}}_u(t) &= \Gamma_3 u(t) e_m^{\mathrm{T}}(t),\\ \dot{\hat{A}}_u(t) &= \Gamma_4 e_m^{\mathrm{T}}(t),\\ \dot{\hat{\theta}}(t) &= -\Gamma_5 \frac{e_m^{\mathrm{T}}(t) e_m(t)}{e_m^{\mathrm{T}}(t) e_m(t) + \eta},\end{aligned} \tag{6.9}$$

where Γ_i is a positive definite matrix such that $\Gamma_i = \Gamma_i^{\mathrm{T}} > 0$, $i = 1, 2, \ldots, 5$.

Proof Choose the following Lyapunov function candidate:

$$J_3(t) = J_1(t) + J_2(t), \tag{6.10}$$

where

$$\begin{aligned}J_1(t) &= \frac{1}{2} e_m^{\mathrm{T}}(t) e_m(t),\\ J_2(t) &= \frac{1}{2} \mathrm{tr}\{\tilde{C}^{\mathrm{T}}(t) \Gamma_1^{-1} \tilde{C}(t) + \tilde{A}^{\mathrm{T}}(t) \Gamma_2^{-1} \tilde{A}(t)\\ &\quad + \tilde{C}_u^{\mathrm{T}}(t) \Gamma_3^{-1} \tilde{C}_u(t) + \tilde{A}_u^{\mathrm{T}}(t) \Gamma_4^{-1} \tilde{A}_u(t)\} + \frac{1}{2} \tilde{\theta}^{\mathrm{T}}(t) \Gamma_5^{-1} \tilde{\theta}(t).\end{aligned}$$

Then, the time derivative of the *Lyapunov function* candidate (6.10) along the trajectories of the error system (6.8) is computed as

$$\begin{aligned}\dot{J}_1(t) = {} & e_m^{\mathrm{T}}(t) C^{*\mathrm{T}} e_m(t) + e_m^{\mathrm{T}}(t) \tilde{C}^{\mathrm{T}}(t) \hat{x}(t)\\ & + e_m^{\mathrm{T}}(t) A^{*\mathrm{T}} \tilde{h}(e_m(t)) + e_m^{\mathrm{T}}(t) \tilde{A}^{T}(t) h(\hat{x}(t))\\ & + e_m^{\mathrm{T}}(t) \tilde{C}_u^{\mathrm{T}}(t) u(t) + e_m^{\mathrm{T}}(t) \tilde{A}_u^{\mathrm{T}}(t) + e_m^{\mathrm{T}}(t) \varepsilon_m(t)\\ & + e_m^{\mathrm{T}}(t) S e_m(t) - \frac{e_m^{\mathrm{T}}(t) \tilde{\theta}(t) e_m(t)}{e_m^{\mathrm{T}}(t) e_m(t) + \eta} + \frac{e_m^{\mathrm{T}}(t) \theta^* e_m(t)}{e_m^{\mathrm{T}}(t) e_m(t) + \eta}.\end{aligned} \tag{6.11}$$

From (6.4), we can obtain

$$e_m^{\mathrm{T}}(t) A^{*\mathrm{T}} \tilde{h}(e_m(t)) \le \frac{1}{2} e_m^{\mathrm{T}}(t) A^{*\mathrm{T}} A^* e_m(t) + \frac{1}{2} k^2 e_m^{\mathrm{T}}(t) e_m(t). \tag{6.12}$$

According to Assumption 6.1, we have

$$e_m^{\mathrm{T}}(t)\varepsilon_a(t) \le \frac{1}{2}e_m^{\mathrm{T}}(t)e_m(t) + \frac{1}{2}\varepsilon_m^{\mathrm{T}}(t)\varepsilon_m(t)$$
$$\le \frac{1}{2}e_m^{\mathrm{T}}(t)e_m(t) + \frac{1}{2}\theta^* e_m^{\mathrm{T}}(t)e_m(t). \tag{6.13}$$

Therefore, (6.11) can be rewritten as

$$\begin{aligned}\dot{J}_1(t) \le\, & e_m^{\mathrm{T}}(t)C^{*\mathrm{T}}e_m(t) + e_m^{\mathrm{T}}(t)\tilde{C}^{\mathrm{T}}(t)\hat{x}(t) \\ & + \frac{1}{2}e_m^{\mathrm{T}}(t)A^{*\mathrm{T}}A^*e_m(t) + \left(\frac{1}{2} + \frac{1}{2}\theta^* + \frac{1}{2}k^2\right)e_m^{\mathrm{T}}(t)e_m(t) \\ & + e_m^{\mathrm{T}}(t)\tilde{A}^{\mathrm{T}}(t)h(\hat{x}(t)) + e_m^{\mathrm{T}}(t)\tilde{C}_u^{\mathrm{T}}(t)u(t) + e_m^{\mathrm{T}}(t)\tilde{A}_u^{\mathrm{T}}(t) \\ & + e_m^{\mathrm{T}}(t)Se_m(t) - \frac{e_m^{\mathrm{T}}(t)\tilde{\theta}(t)e_m(t)}{e_m^{\mathrm{T}}(t)e_m(t)+\eta} + \frac{e_m^{\mathrm{T}}(t)\theta^*e_m(t)}{e_m^{\mathrm{T}}(t)e_m(t)+\eta}.\end{aligned} \tag{6.14}$$

By computing the time derivative of $J_2(t)$, we have

$$\begin{aligned}\dot{J}_2(t) =\, & \mathrm{tr}\{\tilde{C}^{\mathrm{T}}(t)\varGamma_1^{-1}\dot{\tilde{C}}(t) + \tilde{A}^{\mathrm{T}}(t)\varGamma_2^{-1}\dot{\tilde{A}}(t) \\ & + \tilde{C}_u^{\mathrm{T}}(t)\varGamma_3^{-1}\dot{\tilde{C}}_u(t) + \tilde{A}_u^{\mathrm{T}}(t)\varGamma_4^{-1}\dot{\tilde{A}}_u(t)\} + \tilde{\theta}^{\mathrm{T}}(t)\varGamma_5^{-1}\dot{\tilde{\theta}}(t).\end{aligned} \tag{6.15}$$

Combining (6.14) with (6.15), we have

$$\begin{aligned}\dot{J}_3(t) \le\, & e_m^{\mathrm{T}}(t)C^{*\mathrm{T}}e_m(t) + \frac{1}{2}e_m^{\mathrm{T}}(t)A^{*\mathrm{T}}A^*e_m(t) \\ & + e_m^{\mathrm{T}}(t)\left(\left(\frac{1}{2} + \frac{1}{2}\theta^* + \frac{1}{2}k^2\right)I_n + S\right)e_m(t) + \frac{e_m^{\mathrm{T}}(t)\theta^*e_m(t)}{e_m^{\mathrm{T}}(t)e_m(t)+\eta} \\ \le\, & e_m^{\mathrm{T}}(t)\varXi e_m(t),\end{aligned} \tag{6.16}$$

where I_n denotes a $n \times n$ identity matrix and

$$\varXi = C^{*\mathrm{T}} + \frac{1}{2}A^{*\mathrm{T}}A^* + \left(\frac{1}{2} + \frac{3}{2}\theta^* + \frac{1}{2}k^2\right)I_n + S,$$

and S is selected to make $\varXi < 0$. Therefore, it can be concluded that $\dot{J}_3(t) < 0$. Since $J_3(t) > 0$, it follows from [3] that $e_m(t) \to 0$ as $t \to \infty$.

This completes the proof. □

Remark 6.3 According to the results of Theorem 6.2, since $e_m(t) \to 0$ as $t \to \infty$, the term $\upsilon(t) \to 0$ as $t \to \infty$. In addition, $\dot{\hat{C}}(t) \to 0$, $\dot{\hat{A}}(t) \to 0$, $\dot{\hat{C}}_u(t) \to 0$, and $\dot{\hat{A}}_u(t) \to 0$ as $e_m(t) \to 0$. It means that $\hat{C}(t)$, $\hat{A}(t)$, $\hat{C}_u(t)$, and $\hat{A}_u(t)$ all tend to be constant matrices which are denoted C, A, C_u, and A_u, respectively.

Consequently, the nonlinear system (6.1) can be rewritten as

$$\dot{x}(t) = C^{\mathrm{T}} x(t) + A^{\mathrm{T}} h(x(t)) + C_u^{\mathrm{T}} u(t) + A_u^{\mathrm{T}}. \tag{6.17}$$

In this way, the original optimal tracking control problem of (6.1) is transformed into the optimal tracking control problem of (6.17). Next, the controller design based on (6.17) will be given in detail.

It is assumed that the desired trajectory $x_d(t)$ has the following form:

$$\dot{x}_d(t) = C^{\mathrm{T}} x_d(t) + A^{\mathrm{T}} h(x_d(t)) + C_u^{\mathrm{T}} u_d(t) + A_u^{\mathrm{T}}, \tag{6.18}$$

where $u_d(t)$ is the control input of the desired system.

By using (6.17) and (6.18), the error system can be formulated as

$$\dot{e}(t) = C^{\mathrm{T}} e(t) + A^{\mathrm{T}} h_e(t) + C_u^{\mathrm{T}} u_e(t), \tag{6.19}$$

where $h_e(t) = h(x(t)) - h(x_d(t))$ and $u_e(t) = u(t) - u_d(t)$. It is noted that the controller $u(t)$ is composed of two parts, the steady-state controller $u_d(t)$ and the feedback controller $u_e(t)$.

The steady-state controller $u_d(t)$ can be obtained from (6.18) as follows:

$$u_d(t) = C_u^{-\mathrm{T}} (\dot{x}_d(t) - C^{\mathrm{T}} x_d(t) - A^{\mathrm{T}} h(x_d(t)) - A_u^{\mathrm{T}}), \tag{6.20}$$

where C_u^{-1} stands for the pseudo-inverse of C_u. The steady-state controller is used to maintain the tracking error close to zero at the steady-state stage.

Next, the feedback controller $u_e(t)$ will be designed to stabilize the state tracking error dynamics at transient stage in an optimal manner. In the following, for brevity, the denotations $e(t)$, $u_d(t)$, $u_e(t)$, $u(t)$, and $V(e(t))$ are rewritten as e, u_d, u_e, u, and $V(e)$.

The infinite-horizon cost functional (6.2) is transformed into

$$J(e, u) = \int_t^{\infty} l(e(\tau), u_e(\tau)) \mathrm{d}\tau, \tag{6.21}$$

where $l(e, u_e) = e^{\mathrm{T}} Q e + u_e^{\mathrm{T}} R u_e$ is the utility function; Q and R are symmetric positive definite matrices with appropriate dimensions.

It is desirable to find the optimal feedback control u_e^* which stabilizes the system (6.19) and minimizes the cost functional (6.21). The kind of control is called admissible control.

Define the *Hamilton function* as

$$H(e, u_e, J_e) = J_e^{\mathrm{T}} (C^{\mathrm{T}} e + A^{\mathrm{T}} h_e + C_u^{\mathrm{T}} u_e) + e^{\mathrm{T}} Q e + u_e^{\mathrm{T}} R u_e, \tag{6.22}$$

where $J_e = \partial J(e) / \partial e$.

The optimal value function $J^*(e)$ is defined as

$$J^*(e) = \min_{u_e \in \psi(\Omega)} \int_t^{\infty} l(e(\tau), u_e(e(\tau))) \mathrm{d}\tau, \tag{6.23}$$

and satisfies

$$0 = \min_{u_e \in \psi(\Omega)} (H(e, u_e, J_e^*)). \tag{6.24}$$

Further, we can obtain the optimal control u_e^* by solving $\partial H(e, u_e, J_e^*)/\partial u_e = 0$ as

$$u_e^* = -\frac{1}{2} R^{-1} C_u J_e^*, \tag{6.25}$$

where $J_e^* = \partial J^*(e)/\partial e$. Then, the overall optimal control input can be rewritten as $u^* = u_d + u_e^*$.

In the following, we will focus on the optimal feedback controller design using the ADP method, which is implemented by employing the critic NN and the action NN.

A neural network is utilized to approximate $J(e)$ as

$$J(e) = W_c^{\mathrm{T}} \phi_c(e) + \varepsilon_c, \tag{6.26}$$

where W_c is the unknown ideal constant weights and $\phi_c(e) : \mathbb{R}^n \to \mathbb{R}^{N_1}$ is called the critic NN activation function vector, N_1 is the number of neurons in the hidden layer, and ε_c is the critic NN approximation error.

The derivative of the cost function $J(e)$ with respect to e is

$$J_e = \nabla\phi_c^{\mathrm{T}} W_c + \nabla\varepsilon_c, \tag{6.27}$$

where $\nabla\phi_c \triangleq \partial\phi_c(e)/\partial e$ and $\nabla\varepsilon_c \triangleq \partial\varepsilon_c/\partial e$.

Let $\hat{W}_c$ be an estimate of W_c, then we have the estimate of $J(e)$ as follows:

$$\hat{J}(e) = \hat{W}_c^{\mathrm{T}} \phi_c(e). \tag{6.28}$$

Then, the approximate Hamilton function can be derived as follows:

$$\begin{aligned} H(e, u_e, \hat{W}_a) &= \hat{W}_c^{\mathrm{T}} \nabla\phi_c (C^{\mathrm{T}} e + A^{\mathrm{T}} h_e + C_u^{\mathrm{T}} u_e) + e^{\mathrm{T}} Q e + u_e^{\mathrm{T}} R u_e \\ &= e_c. \end{aligned} \tag{6.29}$$

Given any admissible control law u_e, we desire to select $\hat{W}_c$ to minimize the squared residual error $E_c(\hat{W}_c)$ as follows:

$$E_c(\hat{W}_c) = \frac{1}{2} e_c^{\mathrm{T}} e_c. \tag{6.30}$$

The weight update law for the critic NN is a gradient descent algorithm, which is given by

$$\dot{\hat{W}}_c = -\alpha_c \sigma_c (\phi_c^{\mathrm{T}} \hat{W}_c + e^{\mathrm{T}} Q e + u_e^{\mathrm{T}} R u_e), \tag{6.31}$$

where $\alpha_c > 0$ is the adaptive gain of the critic NN, $\sigma_c = \sigma/(\sigma^{\mathrm{T}}\sigma + 1)$, $\sigma = \nabla\phi_c(C^{\mathrm{T}}e + A^{\mathrm{T}}h_e + C_u^{\mathrm{T}}u_e)$. Therefore, there exists a positive constant $\sigma_{cM} > 1$ such that $\|\sigma_c\| \le \sigma_{cM}$. Define the weight estimation error of critic NN to be $\tilde{W}_c = \hat{W}_c - W_c$, and note that, for a fixed control law u_e, the Hamilton function (6.22) becomes

$$\begin{aligned} H(e, u_e, W_c) &= W_c^{\mathrm{T}}\nabla\phi_c(C^{\mathrm{T}}e + A^{\mathrm{T}}h_e + C_u^{\mathrm{T}}u_e) + e^{\mathrm{T}}Qe + u_e^{\mathrm{T}}Ru_e \\ &= \varepsilon_{\mathrm{HJB}}, \end{aligned} \tag{6.32}$$

where the residual error due to the NN approximation error is $\varepsilon_{\mathrm{HJB}} = -\nabla\varepsilon_c(C^{\mathrm{T}}e + A^{\mathrm{T}}h_e + C_u^{\mathrm{T}}u_e)$.

Rewriting (6.31) by using (6.32), we have

$$\dot{\tilde{W}}_c = -\alpha_c\sigma_c(\phi_c^{\mathrm{T}}\tilde{W}_c + \varepsilon_{\mathrm{HJB}}). \tag{6.33}$$

To begin the development of the feedback control law, u_e is approximated by the action NN as

$$u_e = W_a^{\mathrm{T}}\phi_a(e) + \varepsilon_a, \tag{6.34}$$

where W_a is the matrix of unknown ideal constant weights and $\phi_a(e) : \mathbb{R}^n \to \mathbb{R}^{N_2}$ is called the action NN activation function vector, N_2 is the number of neurons in the hidden layer, and ε_a is the action NN approximation error.

Let $\hat{W}_a$ be an estimate of W_a, the actual output can be expressed as

$$\hat{u}_e = \hat{W}_a^{\mathrm{T}}\phi_a(e). \tag{6.35}$$

The feedback error signal used for tuning action NN is defined to be the difference between the real feedback control input applied to the error system (6.19) and the desired control signal input minimizing (6.28) as

$$e_a = \hat{W}_a^{\mathrm{T}}\phi_a + \frac{1}{2}R^{-1}C_u\nabla\phi_c^{\mathrm{T}}\hat{W}_c. \tag{6.36}$$

The objective function to be minimized by the action NN is defined as

$$E_a(\hat{W}_a) = \frac{1}{2}e_a^{\mathrm{T}}e_a. \tag{6.37}$$

The weight update law for the action NN is a gradient descent algorithm, which is given by

$$\dot{\hat{W}}_a = -\alpha_a\phi_a\left(\hat{W}_a^{\mathrm{T}}\phi_a + \frac{1}{2}R^{-1}C_u\nabla\phi_c^{\mathrm{T}}\hat{W}_c\right)^{\mathrm{T}}, \tag{6.38}$$

where $\alpha_a > 0$ is the adaptive gain of the action NN.

Define the weight estimation error of action NN to be $\tilde{W}_a = \hat{W}_a - W_a$. Since the control law in (6.34) minimizes the infinite-horizon cost functional (6.26), from (6.25) we have

$$\varepsilon_a + W_a^{\mathrm{T}}\phi_a + \frac{1}{2}R^{-1}C_u\nabla\phi_c^{\mathrm{T}}W_c + \frac{1}{2}R^{-1}C_u\nabla\varepsilon_c = 0. \tag{6.39}$$

Combining (6.38) with (6.39), we have

$$\dot{\tilde{W}}_a = -\alpha_a\phi_a\left(\tilde{W}_a^{\mathrm{T}}\phi_a + \frac{1}{2}R^{-1}C_u\nabla\phi_c^{\mathrm{T}}\tilde{W}_c + \varepsilon_{12}\right)^{\mathrm{T}}, \tag{6.40}$$

where $\varepsilon_{12} = -(\varepsilon_a + R^{-1}C_u\nabla\varepsilon_c/2)$.

Remark 6.4 It is important to note that the tracking error must be persistently excited: enough for tuning the critic NN and action NN. In order to satisfy the *persistent excitation condition*, probing noise is added to the control input [17]. Further, the persistent excitation condition ensures $\|\sigma_c\| \geq \sigma_{cm}$ and $\|\phi_a\| \geq \phi_{am}$ with σ_{cm} and ϕ_{am} being positive constants.

Based on the above analysis, the optimal tracking controller is composed of the steady-state controller u_d and the optimal feedback controller u_e. As a result, the control input is written as

$$u = u_d + \hat{u}_e. \tag{6.41}$$

According to (6.35) and the error system (6.19), we have

$$\dot{e} = C^{\mathrm{T}}e + A^{\mathrm{T}}h_e + C_u^{\mathrm{T}}\hat{W}_a^{\mathrm{T}}\phi_a. \tag{6.42}$$

Subtracting and adding $C_u^{\mathrm{T}}W_a\phi_a$ to (6.42), and then recalling (6.34), (6.42) is rewritten as

$$\dot{e} = C^{\mathrm{T}}e + A^{\mathrm{T}}h_e + C_u^{\mathrm{T}}\tilde{W}_a^{\mathrm{T}}\phi_a + C_u^{\mathrm{T}}u_e - C_u^{\mathrm{T}}\varepsilon_a. \tag{6.43}$$

In the following, the stability analysis will be performed. First, the following assumption is made, which can reasonably be satisfied under the current problem settings.

Assumption 6.5

(a) The unknown ideal constant weights for the critic NN and the action NN, i.e., W_c and W_a, are upper bounded so that $\|W_c\| \leq W_{cM}$ and $\|W_a\| \leq W_{aM}$ with W_{cM} and W_{aM} being positive constants, respectively.
(b) The NN approximation errors ε_c and ε_a are upper bounded so that $\|\varepsilon_c\| \leq \varepsilon_{cM}$ and $\|\varepsilon_a\| \leq \varepsilon_{aM}$ with ε_{cM} and ε_{aM} being positive constants, respectively.
(c) The vectors of the activation functions of the critic NN and the action NN, i.e., ϕ_c and ϕ_a, are upper bounded so that $\|\phi_c(\cdot)\| \leq \phi_{cM}$ and $\|\phi_a(\cdot)\| \leq \phi_{aM}$ with ϕ_{cM} and ϕ_{aM} being positive constants, respectively.

(d) The gradients of the critic NN approximation error and the activation function vector are upper bounded so that $\|\nabla\varepsilon_c\| \le \varepsilon'_{cM}$ and $\|\nabla\phi_a\| \le \phi_{dM}$ with ε'_{cM} and ϕ_{dM} being positive constants. The residual error is upper bounded so that $\|\varepsilon_{\text{HJB}}\| \le \varepsilon_{\text{HJBM}}$ with $\varepsilon_{\text{HJBM}}$ being positive constant.

Now we are ready to prove the following theorem.

Theorem 6.6 (cf. [25]) *Consider the system given by* (6.17) *and the desired trajectory* (6.18). *Let the control input be provided by* (6.41). *The weight updating laws of the critic NN and the action NN are given by* (6.31) *and* (6.38), *respectively. Let the initial action NN weights be chosen to generate an initial admissible control. Then, the tracking error e, the weight estimate errors $\tilde{W}_c$ and $\tilde{W}_a$ are uniformly ultimately bounded* (*UUB*) *with the bounds specifically given by* (6.51)–(6.53). *Moreover, the obtained control input u is close to the optimal control input u^* within a small bound ε_u, i.e., $\|u - u^*\| \le \varepsilon_u$ as $t \to \infty$ for a small positive constant ε_u.*

Proof Choose the following Lyapunov function candidate:

$$L(t) = L_1(t) + L_2(t) + L_3(t), \tag{6.44}$$

where $L_1(t) = \text{tr}\{\tilde{W}_c^{\text{T}}\tilde{W}_c\}/2\alpha_c$, $L_2(t) = \alpha_c\,\text{tr}\{\tilde{W}_a^{\text{T}}\tilde{W}_a\}/2\alpha_a$, and $L_3(t) = \alpha_c\alpha_a(e^{\text{T}}e + \Gamma J(e))$ with $\Gamma > 0$.

According to Assumption 6.5 and using (6.21), (6.33), and (6.40), the time derivative of the Lyapunov function candidate (6.44) along the trajectories of the error system (6.43) is computed as follows:

$$\dot{L}(t) = \dot{L}_1(t) + \dot{L}_2(t) + \dot{L}_3(t), \tag{6.45}$$

where

$$\begin{aligned}
\dot{L}_1(t) &= \frac{1}{\alpha_c}\text{tr}\{\tilde{W}_c^{\text{T}}\dot{\tilde{W}}_c\} \\
&= \frac{1}{\alpha_c}\text{tr}\{\tilde{W}_c^{\text{T}}(-\alpha_c\sigma_c(\phi_c^{\text{T}}\tilde{W}_c + \varepsilon_{\text{HJB}}))\} \\
&\le -\left(\sigma_{cm}^2 - \frac{\alpha_c}{2}\sigma_{cM}^2\right)\|\tilde{W}_c\|^2 + \frac{1}{2\alpha_c}\varepsilon_{\text{HJB}}^2, \\
\dot{L}_2(t) &= \frac{\alpha_c}{\alpha_a}\text{tr}\{\tilde{W}_a^{\text{T}}\dot{\tilde{W}}_a\} \\
&= \frac{\alpha_c}{\alpha_a}\text{tr}\left\{\tilde{W}_a^{\text{T}}\left(-\alpha_a\phi_a\left(\tilde{W}_a^{\text{T}}\phi_a + \frac{1}{2}R^{-1}C_u\nabla\phi_c^{\text{T}}\tilde{W}_c + \varepsilon_{12}\right)^{\text{T}}\right)\right\} \\
&\le -\left(\alpha_c\phi_{am}^2 - \frac{3}{4}\alpha_c\alpha_a\phi_{aM}^2\right)\|\tilde{W}_a\|^2 \\
&\quad + \frac{\alpha_c}{4\alpha_a}\|R^{-1}\|^2\|C_u\|^2\phi_{dM}^2\|\tilde{W}_c\|^2 + \frac{\alpha_c}{2\alpha_a}\varepsilon_{12}^{\text{T}}\varepsilon_{12},
\end{aligned}$$

$$\begin{aligned}\dot{L}_3(t) &= 2\alpha_c\alpha_a e^{\mathrm{T}}\dot{e} + \alpha_c\alpha_a \Gamma(-e^{\mathrm{T}}Qe - u_e^{\mathrm{T}}Ru_e)\\ &= 2\alpha_c\alpha_a e^{\mathrm{T}}(C^{\mathrm{T}}e + A^T h_e + C_u^{\mathrm{T}}\tilde{W}_a^{\mathrm{T}}\phi_a + C_u^{\mathrm{T}}u_e\\ &\quad - C_u^{\mathrm{T}}\varepsilon_a) + \alpha_c\alpha_a\Gamma(-e^{\mathrm{T}}Qe - u_e^{\mathrm{T}}Ru_e)\\ &\le \alpha_c\alpha_a(2\|C\| + 3 + \|A\|^2 + k^2 - \Gamma\lambda_{\min}(Q))\|e\|^2\\ &\quad + \alpha_c\alpha_a\phi_{aM}^2\|C_u\|^2\|\tilde{W}_a\|^2 + \alpha_c\alpha_a\|C_u\|^2\varepsilon_a^{\mathrm{T}}\varepsilon_a\\ &\quad + \alpha_c\alpha_a(\|C_u\|^2 - \Gamma\lambda_{\min}(R))\|u_e\|^2.\end{aligned}$$

Then we obtain

$$\begin{aligned}\dot{L}(t) \le &-\left(\sigma_{cm}^2 - \frac{\alpha_c}{2}\sigma_{cM}^2 - \frac{\alpha_c}{4\alpha_a}\|R^{-1}\|^2\|C_u\|^2\phi_{dM}^2\right)\|\tilde{W}_c\|^2\\ &-\left(\alpha_c\phi_{am}^2 - \frac{3}{4}\alpha_c\alpha_a\phi_{aM}^2 - \alpha_c\alpha_a\phi_{aM}^2\|C_u\|^2\right)\|\tilde{W}_a\|^2\\ &-\alpha_c\alpha_a(-\|C_u\|^2 + \Gamma\lambda_{\min}(R))\|u_e\|^2\\ &-\alpha_c\alpha_a(-2\|C\| - 3 - \|A\|^2 - k^2 + \Gamma\lambda_{\min}(Q))\|e\|^2\\ &+\frac{1}{2\alpha_c}\varepsilon_{\mathrm{HJB}}^2 + \frac{\alpha_c}{2\alpha_a}\varepsilon_{12}^{\mathrm{T}}\varepsilon_{12} + \alpha_c\alpha_a\|C_u\|^2\varepsilon_a^{\mathrm{T}}\varepsilon_a.\end{aligned} \tag{6.46}$$

By using Assumption 6.5, we have $\|\varepsilon_{12}\| \le \varepsilon_{12M}$, where $\varepsilon_{12M} = \varepsilon_{aM} + R^{-1}C_u\varepsilon'_{cM}/2$. Then, we have

$$\frac{1}{2\alpha_c}\varepsilon_{\mathrm{HJB}}^2 + \frac{\alpha_c}{2\alpha_a}\varepsilon_{12}^{\mathrm{T}}\varepsilon_{12} + \alpha_c\alpha_a\|C_u\|^2\varepsilon_{12}^{\mathrm{T}}\varepsilon_{12} \le D_M, \tag{6.47}$$

where $D_M = \varepsilon_{\mathrm{HJBM}}^2/(2\alpha_c) + \alpha_c\varepsilon_{12M}^2/(2\alpha_a) + \alpha_c\alpha_a\|C_u\|^2\varepsilon_{12M}^2$.

If Γ, α_c, and α_a are selected to satisfy

$$\Gamma > \max\left\{\frac{\|C_u\|^2}{\lambda_{\min}(R)}, \frac{2\|C\| + 3 + \|A\|^2 + k^2}{\lambda_{\min}(Q)}\right\}, \tag{6.48}$$

$$\alpha_c < \frac{4\alpha_a\sigma_{cm}^2}{2\alpha_a\sigma_{cM}^2 + \|R^{-1}\|^2\|C_u\|^2\phi_{dM}^2}, \tag{6.49}$$

$$\alpha_a < \frac{4\phi_{am}^2}{3\phi_{aM}^2 + 4\phi_{aM}^2\|C_u\|^2}, \tag{6.50}$$

and given the following inequalities:

$$\begin{aligned}\|e\| &> \sqrt{\frac{D_M}{\alpha_c\alpha_a(-2\|C\| - 3 - \|A\|^2 - k^2 + \Gamma\lambda_{\min}(Q))}}\\ &\triangleq b_e,\end{aligned} \tag{6.51}$$

or

$$\|\tilde{W}_c\| > \sqrt{\frac{4\alpha_a D_M}{4\alpha_a\sigma_{cm}^2 - 2\alpha_a\alpha_c\sigma_{cM}^2 - \alpha_c\|R^{-1}\|^2\|C_u\|^2\phi_{dM}^2}} \triangleq b_{\tilde{W}_c}, \tag{6.52}$$

or

$$\|\tilde{W}_a\| > \sqrt{\frac{4D_M}{4\alpha_c\phi_{am}^2 - 3\alpha_c\alpha_a\phi_{aM}^2 - 4\alpha_c\alpha_a\phi_{aM}^2\|C_u\|^2}} \triangleq b_{\tilde{W}_a}, \tag{6.53}$$

then we can conclude $\dot{L}(t) < 0$. Therefore, using Lyapunov theory [7], it can be concluded that the tracking error e, and the NN weight estimation errors $\tilde{W}_c$ and $\tilde{W}_a$ are UUB.

Next, we will prove $\|u - u^*\| \le \varepsilon_u$ as $t \to \infty$. Recalling the expression of u^* together with (6.34) and (6.41), we have

$$u - u^* = \tilde{W}_a^{\mathrm{T}}\phi_a + \varepsilon_a. \tag{6.54}$$

When $t \to \infty$, the upper bound of (6.54) is

$$\|u - u^*\| \le \varepsilon_u, \tag{6.55}$$

where $\varepsilon_u = b_{\tilde{W}_a}\phi_{aM} + \varepsilon_{aM}$.

This completes the proof. □

Remark 6.7 From (6.31) and (6.38), it is noted that the weights of critic NN and action NN are updated simultaneously in contrast with some standard ADP methods in which the weights of critic NN and action NN are updated sequentially.

Remark 6.8 If the NN approximation errors ε_c and ε_a are considered to be negligible, then from (6.47) we have $D_M = 0$, with $u \to u^*$. Otherwise, the obtained control input u is close to the optimal input u^* within a small bound ε_u.

Due to the presence of the NN approximation errors ε_c and ε_a, the tracking error is UUB instead of asymptotically convergent to zero. In the following, for improving the tracking performance, an additional robustifying term is developed to attenuate the NN approximation errors such that tracking error converges to zero asymptotically, which can be constructed in the form

$$u_r = \frac{K_r e}{e^{\mathrm{T}}e + \zeta}, \tag{6.56}$$

where $\zeta > 0$ is a constant, $K_r > K_{r\,\min}$ is a designed parameter. $K_{r\,\min}$ is selected to satisfy the following inequality:

$$K_{r\,\min} \ge \frac{D_M(e^{\mathrm{T}}e + \zeta)}{2\alpha_c\alpha_a\|C_u\|e^{\mathrm{T}}e}. \tag{6.57}$$

Then, the overall control input is given as

$$u_{ad} = u - u_r, \tag{6.58}$$

where u is the same as (6.41).

Applying (6.58) to the error system (6.17) and using (6.18), a new error system is obtained as follows:

$$\dot{e} = C^{\mathrm{T}} e + A^{\mathrm{T}} h_e + C_u^{\mathrm{T}} \tilde{W}_a^{\mathrm{T}} \phi_a + C_u^{\mathrm{T}} u_e - C_u^{\mathrm{T}} \varepsilon_a - C_u^{\mathrm{T}} u_r. \tag{6.59}$$

Theorem 6.9 (cf. [25]) *Consider the system given by* (6.17) *and the desired trajectory* (6.18). *Let the control input be provided by* (6.58). *The weight updating laws of the critic NN and the action NN are given by* (6.31) *and* (6.38), *respectively. Let the initial action NN weights be chosen to generate an initial admissible control. Then, the tracking error e and the weight estimation errors $\tilde{W}_c$ and $\tilde{W}_a$ will asymptotically converge to zero. Moreover, the obtained control input u_{ad} is close to the optimal control input u^* within a small bound δ_u, i.e., $\|u_{ad} - u^*\| \le \delta_u$ as $t \to \infty$ for a small positive constant δ_u.*

Proof Choose the same Lyapunov function candidate as that in Theorem 6.6. Differentiating the Lyapunov function candidate in (6.44) along the trajectories of the error system in (6.59), similar to the proof of Theorem 6.6, by using (6.56) and (6.57), we obtain

$$\begin{aligned}
\dot{L}(t) \le & -\left(\sigma_{cm}^2 - \frac{\alpha_c}{2}\sigma_{cM}^2 - \frac{\alpha_c}{4\alpha_a}\|R^{-1}\|^2\|C_u\|^2\phi_{dM}^2\right)\|\tilde{W}_c\|^2 \\
& - \left(\alpha_c\phi_{am}^2 - \frac{3}{4}\alpha_c\alpha_a\phi_{aM}^2 - \alpha_c\alpha_a\phi_{aM}^2\|C_u\|^2\right)\|\tilde{W}_a\|^2 \\
& - \alpha_c\alpha_a(-\|C_u\|^2 + \Gamma\lambda_{\min}(R))\|u_e\|^2 \\
& - \alpha_c\alpha_a(-2\|C\| - 3 - \|A\|^2 - k^2 + \Gamma\lambda_{\min}(Q))\|e\|^2.
\end{aligned} \tag{6.60}$$

Choosing Γ, α_c, and α_a as Theorem 6.6, we have $\dot{L}(t) \le 0$. Equations (6.44) and (6.60) guarantee that the tracking error e, NN weight estimation errors $\tilde{W}_c$ and $\tilde{W}_a$ are bounded, since L is nonincreasing. Because all the variables on the right-hand side of (6.59) are bounded, $\dot{e}$ is also bounded. From (6.60), we have

$$\dot{L}(t) \le -B_e\|e\|^2, \tag{6.61}$$

where $B_e = \alpha_c\alpha_a(-2\|C\| - 3 - \|A\|^2 - k^2 + \Gamma\lambda_{\min}(Q))$.

Integrating both sides of (6.61) and after some manipulations, we have

$$\int_0^\infty \|e\|^2 \mathrm{d}t \le B_e^{-1}(L(0) - L(\infty)). \tag{6.62}$$

Since the right side of (6.59) is bounded, $\|e\| \in \mathcal{L}_2$. Using Barbalat's lemma [7], we have $\lim_{t\to\infty}\|e\| = 0$. Similarly, we can prove that $\lim_{t\to\infty}\|\tilde{W}_c\| = 0$ and $\lim_{t\to\infty}\|\tilde{W}_a\| = 0$.

Next, we will prove $\|u_{ad} - u^*\| \leq \delta_u$ as $t \to \infty$. From (6.34) and (6.58), we can have

$$u_{ad} - u^* = \tilde{W}_a^{\mathrm{T}} \phi_a + \varepsilon_a + u_r. \tag{6.63}$$

Since $\|e\| \to 0$ as $t \to \infty$, the robustifying control input $\|u_r\| \to 0$ as $t \to \infty$. Then, the upper bound of (6.63) is

$$\|u_{ad} - u^*\| \leq \delta_u, \tag{6.64}$$

where $\delta_u = \varepsilon_{aM}$.

This completes the proof. □

Remark 6.10 From (6.55) and (6.64), it can be seen that δ_u is smaller than ε_u, which reveals the role of the robustifying term in making the obtained control input closer to the optimal control input.

6.2.3 Simulations

In this subsection, two examples are provided to demonstrate the effectiveness of the present approach.

Example 6.11 Consider the following affine nonlinear continuous-time system:

$$\begin{aligned} \dot{x}_1 &= -x_1 + x_2, \\ \dot{x}_2 &= -0.5x_1 - 0.5x_2(1 - (\cos(2x_1) + 2)^2) \\ &\quad + (\cos(2x_1) + 2)u. \end{aligned} \tag{6.65}$$

The cost functional is defined by (6.21) where Q and R are chosen as identity matrices of appropriate dimensions. The control object is to make x_1 follow the desired trajectory $x_{1d} = \sin(t)$. It is assumed that the system dynamics is unknown and input/output data are available.

First, a data-based model is established to estimate the nonlinear system dynamics. Let us select the RNN as (6.5) with $S = -30I_2$ and $\eta = 1.5$. The activation function $h(\hat{x})$ is selected as hyperbolic tangent function $\tanh(\hat{x})$. Select the design parameters in Theorem 6.2 as $\Gamma_1 = [1, 0.1; 0.1, 1]$, $\Gamma_2 = [1, 0.2; 0.2, 1]$, $\Gamma_3 = [1, 0.1; 0.1, 1]$, $\Gamma_4 = 0.2$, and $\Gamma_5 = 0.1$. Then, we can obtain the trajectories of the modeling error as shown in Fig. 6.1. It is observed that the obtained data-based model can reconstruct the nonlinear system dynamics successfully, as Theorem 6.2 predicted.

Then, based on the obtained data-based model, the approximate *optimal robust* controller is implemented for the unknown affine nonlinear continuous-time system (6.65). The activation function of the critic NN is selected as $\phi_c = [e_1^2 \; e_1 e_2 \; e_2^2]^{\mathrm{T}}$, the

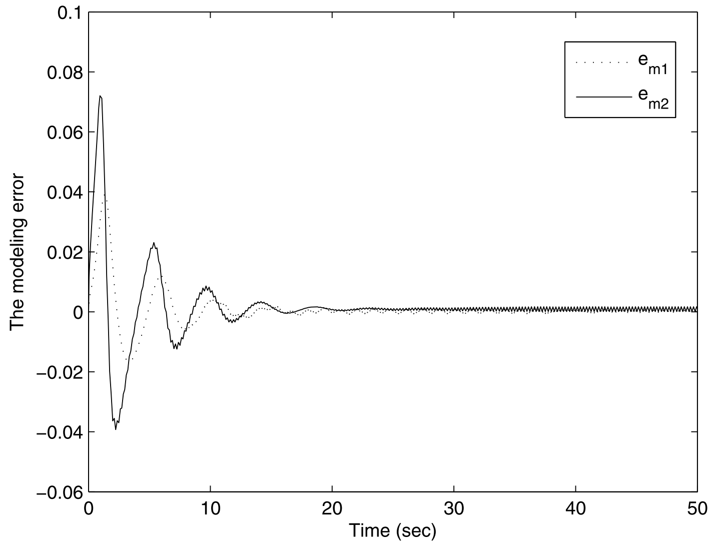

Fig. 6.1 The modeling error for the affine nonlinear system

critic NN weights are denoted $\hat{W}_c = [W_{c1}\ W_{c2}\ W_{c3}]^{\mathrm{T}}$. The activation function of the action NN ϕ_a is chosen as the gradient of the critic NN, the action NN weights are denoted $\hat{W}_a = [W_{a1}\ W_{a2}\ W_{a3}]^{\mathrm{T}}$. The adaptive gains for the critic NN and action NN are selected as $\alpha_c = 0.8$ and $\alpha_a = 0.5$, and the design parameters of the robustifying term are selected as $K_r = [20, 20]$, $\zeta = 1.2$. Additionally, the critic NN weights are set as $[1, 1, 1]^{\mathrm{T}}$ at the beginning of the simulation with the initial weights of the action NN chosen to reflect the initial admissible control. To maintain the excitation condition, probing noise is added to the control input for the first 5000 s.

After simulation, the trajectory of the tracking error is shown in Fig. 6.2. The convergence trajectories of the critic NN weights and action NN weights are shown in Figs. 6.3 and 6.4, respectively. For comparing the tracking performance, we apply the obtained optimal robust controller and the initial admissible controller to system (6.65) under the same initial state, and obtain the trajectories of tracking error as shown in Fig. 6.5, respectively. It can be seen from Fig. 6.5 that the present robust approximate optimal controller yields a better tracking performance than the initial admissible controller.

Example 6.12 Consider the following continuous-time nonaffine nonlinear system:

$$\begin{aligned} \dot{x}_1 &= x_2, \\ \dot{x}_2 &= x_1^2 + 0.15u^3 + 0.1(4 + x_2^2)u + \sin(0.1u). \end{aligned} \tag{6.66}$$

The cost functional is defined as Example 6.11. The control objective is to make x_1 follow the desired trajectory $x_{1d} = \sin(t)$. It is assumed that the system dynamics is unknown and input/output data are available.

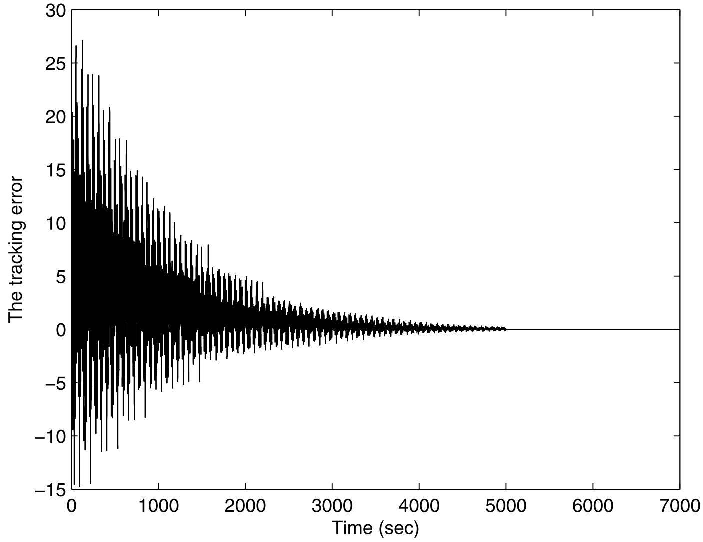

Fig. 6.2 The tracking error for the affine nonlinear system

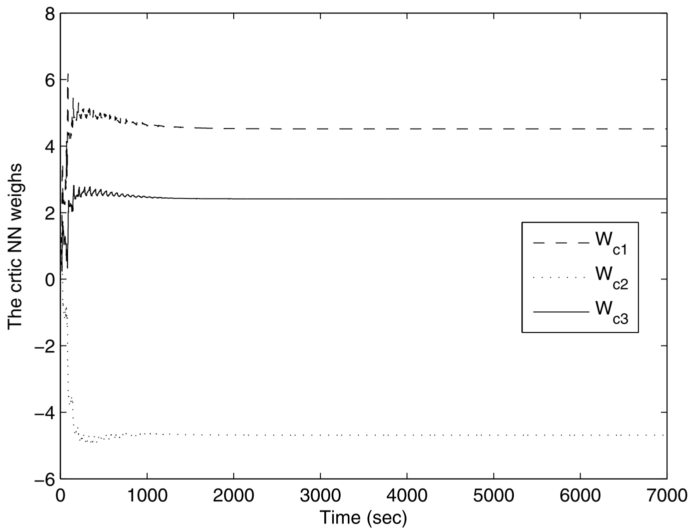

Fig. 6.3 The critic NN weights

Using a similar method as shown in Example 6.11, we can obtain the trajectories of modeling error as shown in Fig. 6.6. It is observed that the obtained data-based model learns the nonlinear system dynamics successfully, as Theorem 6.2 predicted. Then, based on the obtained data-based model we design the robust approximate optimal controller, which is then applied to the unknown nonaffine nonlinear system

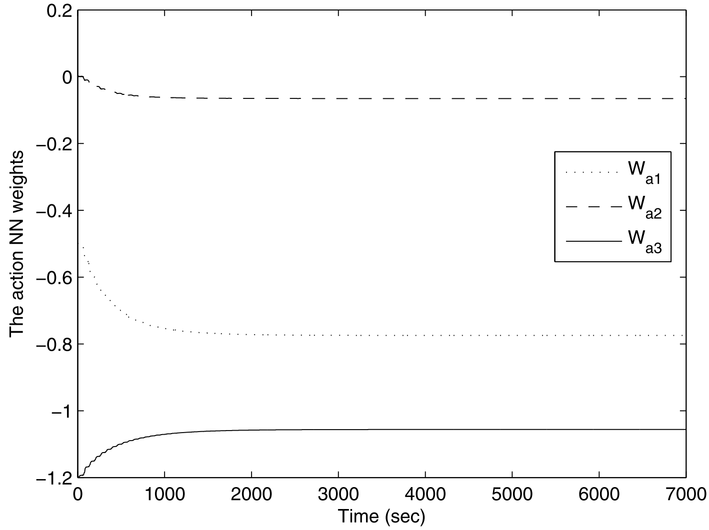

Fig. 6.4 The action NN weights

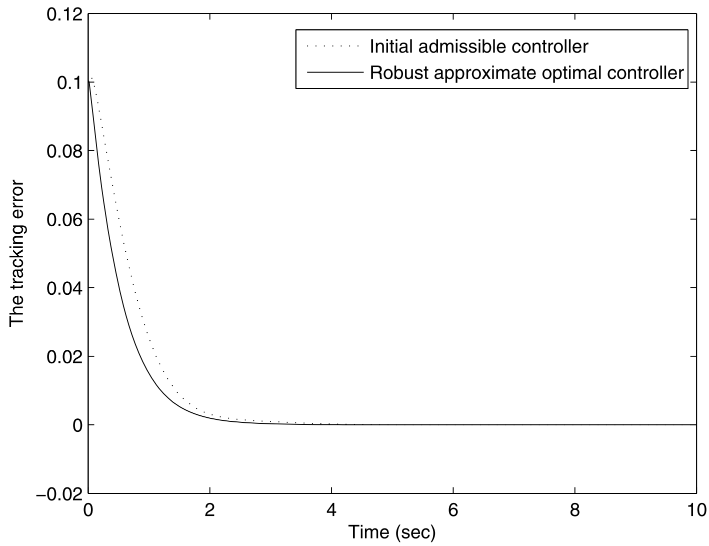

Fig. 6.5 The comparison result between initial admissible controller and robust approximate optimal controller

(6.66). The activation functions of the critic NN and action NN are the same as the ones in Example 6.11. The adaptive gains for critic NN and action NN are selected as $\alpha_c = 0.5$ and $\alpha_a = 0.2$ and the parameters of the robustifying term are selected as $K_r = [10, 10]$, $\zeta = 1.2$. Additionally, the critic NN weights are set as $[1, 1, 1]^{\mathrm{T}}$ at

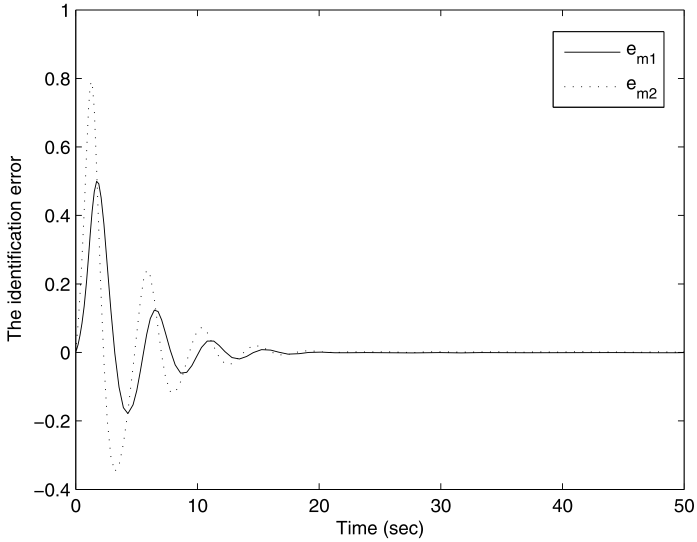

Fig. 6.6 The modeling error for the continuous-time nonaffine nonlinear system

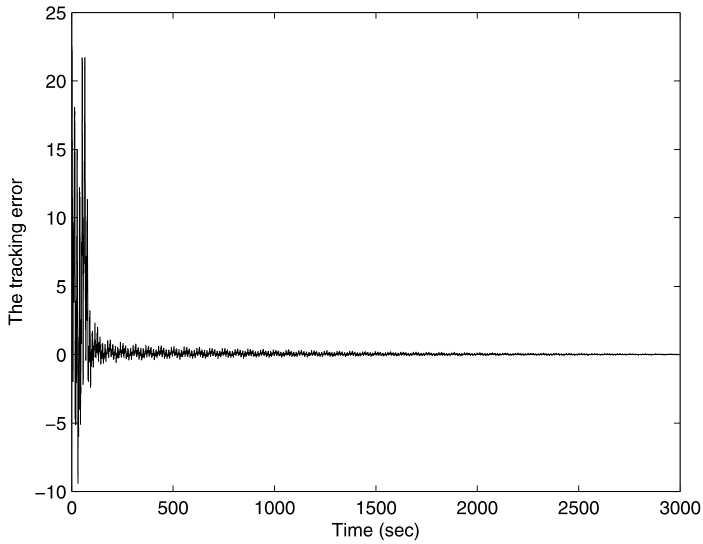

Fig. 6.7 The tracking error for the nonaffine nonlinear system

the beginning of the simulation with the initial weights of the action NN chosen to reflect the initial admissible control. Similarly, to maintain the excitation condition, probing noise is added to the control input for the first 1500 s.

After simulation, the trajectory of the tracking error is shown in Fig. 6.7. The convergence trajectories of the critic NN weights and action NN weights are shown

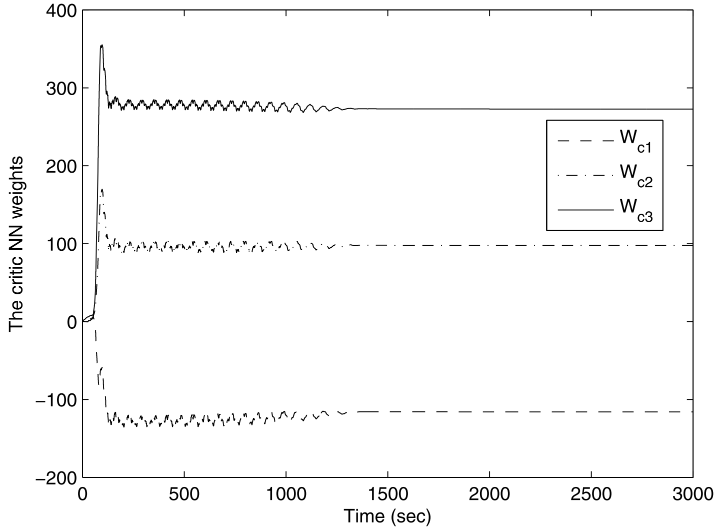

Fig. 6.8 The critic NN weights

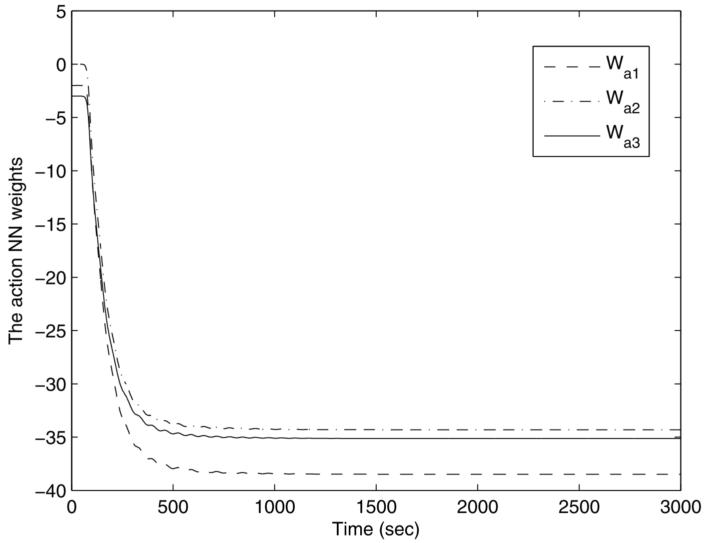

Fig. 6.9 The action NN weights

in Figs. 6.8 and 6.9, respectively. Similarly, for comparing the tracking performance, we apply the obtained robust optimal controller and the initial admissible controller to system (6.66) for the same initial state, and obtain the trajectories of tracking error as shown in Fig. 6.10, respectively. It can be seen from Fig. 6.10 that the present robust approximate optimal controller yields better tracking performance

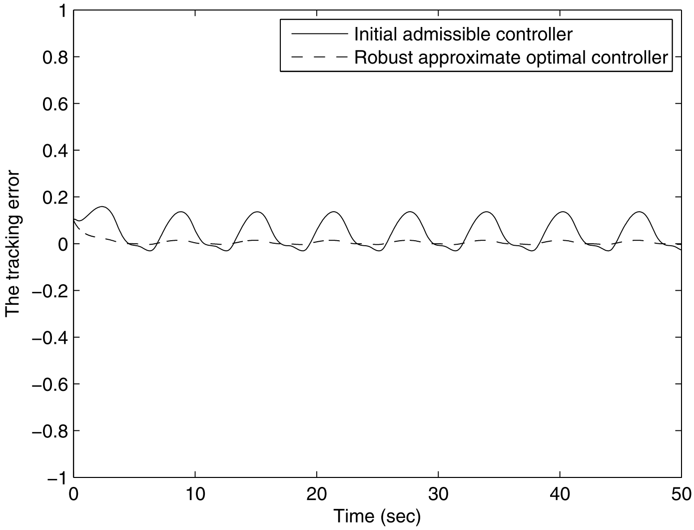

Fig. 6.10 The comparison result between initial admissible controller and robust approximate optimal controller

than the initial admissible controller. The simulation results reveal that the present controller can be applied to nonaffine nonlinear systems and we obtain satisfactory tracking performance even for the unknown system dynamics.

6.3 Optimal Feedback Control for Nonaffine Nonlinear Systems

In this section, we will study the optimal feedback control problem of a class of continuous-time nonaffine nonlinear systems via the ADP method.

6.3.1 Problem Formulation

Consider a class of continuous-time nonaffine nonlinear systems described as follows:

$$\begin{aligned}\dot{x}_i &= x_{i+1}, i = 1, \dots, n-1,\\ \dot{x}_n &= f(x, u),\\ y &= x_1,\end{aligned} \tag{6.67}$$

where $x = [x_1, \dots, x_n]^{\mathrm{T}} \in \mathbb{R}^n$ is the system state vector, $u \in \mathbb{R}$ is the control input, and $y \in \mathbb{R}$ is the output of the system; $f(x, u)$ is an unknown smooth function satisfying $f(0, 0) = 0$.

Assumption 6.13 The control effectiveness term $f_u(x,u) \triangleq \partial f(x,u)/\partial u$ has a known sign and is bounded away from zero, i.e., there exists $d_f > 0$ such that $|f_u(x,u)| \geq d_f$. Without loss of generality, we assume $f_u > 0$.

The control objective is to force the system output y to follow the desired trajectory y_d while ensuring that all the signals of the closed-loop system are bounded. Assume that y_d and its up to n times derivatives, namely $\dot{y}_d, y_d^{(2)}, \ldots,$ and $y_d^{(n)}$ are smooth, bounded, and available for design.

6.3.2 Robust Approximate Optimal Control Based on ADP Algorithm

Define the tracking error $\tilde{y} = y - y_d$, desired trajectory vector $\bar{y}_d = [y_d, \ldots, y_d^{(n-1)}]^{\mathrm{T}}$, tracking error vector $\tilde{x} = x - \bar{y}_d$, and the filtered tracking error as

$$r = \tilde{y}^{(n-1)} + \lambda_{n-1}\tilde{y}^{(n-2)} + \cdots + \lambda_2\tilde{y}^{(1)} + \lambda_1\tilde{y} = [\Lambda^{\mathrm{T}} \;\; 1]\tilde{x}, \tag{6.68}$$

where $\lambda_i, i = 1, \ldots, n-2$, are chosen such that the polynomial $H(s) = s^{(n-1)} + \lambda_{n-1}s^{(n-2)} + \cdots + \lambda_1$ is Hurwitz, and $\Lambda = [\lambda_1, \ldots, \lambda_{n-1}]^{\mathrm{T}}$.

With these definitions, the tracking error dynamics can be given as follows:

$$\dot{r} = -y_d^{(n)} + [0 \;\; \Lambda^{\mathrm{T}}]\tilde{x} + f(x,u). \tag{6.69}$$

Then, feedback linearization is performed by introducing a so-called pseudocontrol

$$\upsilon = \hat{f}(x,u), \tag{6.70}$$

where $\hat{f}(x,u)$ is an available approximation of $f(x,u)$ satisfying Assumption 6.13.

Assumption 6.14 $\hat{f}_u \triangleq \partial\hat{f}(x,u)/\partial u \geq d_{\hat{f}} > 0$ for some $d_{\hat{f}}$, and there exist some constants $b_l, b_u > 0$ such that $b_l \leq f_u/\hat{f}_u \leq b_u$.

By adding and subtracting υ on the right-hand side of (6.69), the tracking error dynamics can be rewritten as follows:

$$\dot{r} = -y_d^{(n)} + [0 \;\; \Lambda^{\mathrm{T}}]\tilde{x} + \hat{f}(x,u) + \Delta, \tag{6.71}$$

where $\Delta = f(x,u) - \hat{f}(x,u)$ is the modeling error.

The pseudocontrol υ is designed as

$$\upsilon = \upsilon_{rm} + \upsilon_{dc} - \upsilon_{ad} + \upsilon_r, \tag{6.72}$$

where $\upsilon_{rm} = y_d^{(n)} - [0 \ \ \Lambda^{\mathrm{T}}]\tilde{x}$, $\upsilon_{dc} = -Kr$ is to stabilize the tracking error dynamics in the absence of a modeling error, υ_{ad} is used to approximately cancel the modeling error, and υ_r is a robustifying term.

By inverting (6.70), we have the following control law:

$$u = \hat{f}^{-1}(x, \upsilon). \tag{6.73}$$

Then, applying (6.73) to (6.71), we have

$$\dot{r} = -Kr + \Delta - \upsilon_{ad} + \upsilon_r. \tag{6.74}$$

The term $\Delta - \upsilon_{ad}$ can be expressed as

$$\Delta - \upsilon_{ad} = f(x, \hat{f}^{-1}(x, \upsilon_{rm} + \upsilon_{dc} - \upsilon_{ad} + \upsilon_r)) - \upsilon_{rm} - \upsilon_{dc} - \upsilon_r. \tag{6.75}$$

Reference [2] points out that Δ depends on υ_{ad} through (6.72) and (6.73), while υ_{ad} is designed to cancel Δ. A contraction mapping assumption on Δ with respect to υ_{ad} is required to guarantee the existence of a smooth function $\upsilon_{ad}^* = \Delta(\cdot, \upsilon_{ad}^*)$. To remove the contraction mapping assumption, we follow the work of [8]. Define $\upsilon_l \triangleq \upsilon_{rm} + \upsilon_{dc}$, $\upsilon^* \triangleq \hat{f}(x, f^{-1}(x, \upsilon_l))$. Then we have $\upsilon_l = f(x, \hat{f}^{-1}(x, \upsilon^*))$.

Using the mean value theorem [5], (6.75) can be expressed as

$$\begin{aligned}
\Delta - \upsilon_{ad} &= f(x, u) - \hat{f}(x, u) - \upsilon_{ad} \\
&= f(x, \hat{f}^{-1}(x, \upsilon)) - \upsilon_l + \upsilon_{ad} - \upsilon_r - \upsilon_{ad} \\
&= f(x, \hat{f}^{-1}(x, \upsilon)) - f(x, \hat{f}^{-1}(x, \upsilon^*)) - \upsilon_r \\
&= f_\upsilon(\bar{\upsilon})(\upsilon - \upsilon^*) - \upsilon_r \\
&= f_\upsilon(\bar{\upsilon})(\upsilon_l - \upsilon_{ad} + \upsilon_r - \hat{f}(x, f^{-1}(x, \upsilon_l))) - \upsilon_r \\
&= f_\upsilon(\bar{\upsilon})(-\upsilon_{ad} + \upsilon_r + \bar{\Delta}) - \upsilon_r,
\end{aligned} \tag{6.76}$$

where $\bar{\Delta} = \upsilon_l - \hat{f}(x, f^{-1}(x, \upsilon_l)) = f(x, \hat{f}^{-1}(x, \upsilon^*)) - f(x, f^{-1}(x, \upsilon_l))$ denotes the unknown uncertain term, $f_\upsilon(\bar{\upsilon}) \triangleq (\partial f/\partial u)(\partial u/\partial \upsilon)|_{\upsilon=\bar{\upsilon}} = (\partial f/\partial u)/(\partial \hat{f}/\partial u)|_{u=\hat{f}^{-1}(x,\bar{\upsilon})}$, $\bar{\upsilon} = \lambda\upsilon + (1-\lambda)\upsilon^*$ and $0 \le \lambda(\upsilon) \le 1$.

Then, substituting (6.76) into (6.74), we have

$$\dot{r} = -Kr + f_\upsilon(\bar{\upsilon})(-\upsilon_{ad} + \bar{\Delta}) + f_\upsilon(\bar{\upsilon})\upsilon_r. \tag{6.77}$$

If $\bar{\Delta}$ is known, υ_{ad} can be chosen as $\upsilon_{ad} = \bar{\Delta}$; then we let $\upsilon_r = 0$. Since $\bar{\Delta}$ is unknown, the desired υ_{ad} cannot be implemented directly. Instead, an action NN is employed to approximate $\bar{\Delta}$ as follows:

$$\bar{\Delta} = W_a^{\mathrm{T}} \phi_a(\bar{x}) + \varepsilon_a(\bar{x}), \tag{6.78}$$

where $\bar{x} = [1, x, \upsilon_l, \bar{y}_d]$, $W_a \in \mathbb{R}^N$ is the ideal weight of the action NN, $\phi_a(\bar{x}) \in \mathbb{R}^N$ is the basis function of the action NN, and $\varepsilon_a(\bar{x})$ is the reconstruction error of the action NN satisfying $\|\varepsilon_a(\bar{x})\| \le \varepsilon_a^*$.

Remark 6.15 Because the unknown nonlinear function $f(x,u)$ of (6.67) is an implicit function with respect to the control input u, traditional ADP methods can rarely be applied. To overcome this difficulty, the action NN is employed to approximate the derived unknown uncertain term $\bar{\Delta}$ instead of modeling the unknown system (6.67) directly.

The adaptive signal υ_{ad} is designed as

$$\upsilon_{ad} = \hat{W}_a^{\mathrm{T}} \phi_a(\bar{x}), \tag{6.79}$$

where $\hat{W}_a$ is the estimate of W_a.

Substituting (6.78) and (6.79) into (6.77), the following is immediate:

$$\begin{aligned}\dot{r} = &-Kr + (f_\upsilon(\bar{\upsilon}) - b_u)\tilde{W}_a^{\mathrm{T}} \phi_a(\bar{x}) + b_u \tilde{W}_a^{\mathrm{T}} \phi_a \\ &+ f_\upsilon(\bar{\upsilon})\upsilon_r + f_\upsilon(\bar{\upsilon})\varepsilon_a(\bar{x}),\end{aligned} \tag{6.80}$$

where $\tilde{W}_a = W_a - \hat{W}_a$.

Using adaptive bounding technique, the robustifying term υ_r is designed as

$$\upsilon_r = -\frac{b_r}{1-b_r}|\hat{\kappa}|\psi \tanh\left(\frac{\mathcal{R}\psi}{\alpha}\right), \tag{6.81}$$

where $b_r = 1-(b_l/b_u) < 1$ and α is a design parameter, $\hat{\kappa}$ is the adaptive parameter, $\mathcal{R}$ and ψ will be defined later. Applying (6.81) to (6.80), the tracking error dynamics can be rewritten as

$$\begin{aligned}\dot{r} = &-Kr + b_u \tilde{W}_a^{\mathrm{T}} \phi_a + b_u \left[\upsilon_r + \left(\frac{f_\upsilon(\bar{\upsilon})}{b_u} - 1\right)\left(\upsilon_r + \tilde{W}_a^{\mathrm{T}} \phi_a\right)\right] \\ &+ f_\upsilon(\bar{\upsilon})\varepsilon_a(\bar{x}).\end{aligned} \tag{6.82}$$

Remark 6.16 Since $\tanh(\cdot)$ can maintain a continuous control signal while $\mathrm{sgn}(\cdot)$ will result in a chattering phenomenon due to its discontinuity, the robustifying term is designed based on $\tanh(\cdot)$ rather than $\mathrm{sgn}(\cdot)$.

Next, we choose the critic signal as [13]

$$\mathcal{R}_n = \mathcal{R} + |\mathcal{R}| W_c^{\mathrm{T}} \phi_c(r). \tag{6.83}$$

The first term $\mathcal{R}$ is called the primary critic signal which is defined in terms of the performance measure as

$$\mathcal{R} = \frac{\chi}{1+e^{-mr}} - \frac{\chi}{1+e^{mr}}, \tag{6.84}$$

where m is a positive constants and the value of $\mathcal{R}$ is bounded in the interval $[-\chi, \chi]$ with $\chi > 0$ being the critic slope gain. The second term $|\mathcal{R}| W_c^{\mathrm{T}} \phi_c(r)$ is called the second critic signal, where W_c is the ideal weight of critic NN and $\phi_c(r)$ is the basis function of critic NN. The actual output of the critic NN is $\hat{W}_c^{\mathrm{T}}\phi_c(r)$ where $\hat{W}_c$ is the estimate of W_c. Then, the actual critic signal can be expressed as $\hat{\mathcal{R}}_n = \mathcal{R} + |\mathcal{R}|\hat{W}_c^{\mathrm{T}}\phi_c(r)$. Define $\tilde{W}_c = W_c - \hat{W}_c$. It should be noted that $\mathcal{R}$ will approach zero when r approaches zero. Therefore we can conclude that $\hat{\mathcal{R}}_n$ will approach zero. As a learning signal, the critic signal $\hat{\mathcal{R}}_n$ is more informative than the filtered tracking error r. Consequently, a larger control input can be yielded and better tracking performance can be obtained.

Next, the uniformly ultimate boundedness of the closed-loop system is demonstrated by the Lyapunov method.

Assumption 6.17 The ideal weights of the action NN and the critic NN, i.e., W_a and W_c, are bounded above by unknown positive constants so that $\|W_a\| \le W_a^*$, $\|W_c\| \le W_c^*$.

Assumption 6.18 The activation functions of the action NN and critic NN, ϕ_a and ϕ_c, are bounded above by known positive constants, so that $\|\phi_a\| \le \phi_a^*$, $\|\phi_c\| \le \phi_c^*$.

Lemma 6.19 (cf. [4]) *The following inequality holds*:

$$\begin{aligned}\|\delta\| &\triangleq \|\mathcal{R} f_\upsilon(\bar{\upsilon})\varepsilon_a(\bar{x}) + b_u b_r|\mathcal{R}|\|\tilde{W}_a\|\|\phi_a\| \\ &\quad + b_u|\mathcal{R}|\mathrm{tr}\{\hat{W}_a^{\mathrm{T}}\phi_a\phi_c^{\mathrm{T}}W_a - W_a^{\mathrm{T}}\phi_a(\phi_c^{\mathrm{T}}\hat{W}_c)\}\| \\ &\le b_u b_r|\mathcal{R}|\kappa^*\psi,\end{aligned} \tag{6.85}$$

where κ^* is an unknown constant and $\psi = 1 + \|\hat{W}_a\| + \|\hat{W}_c\|$.

Proof Using Assumptions 6.13, 6.14, 6.17, and 6.18, the boundedness of $\varepsilon_a(\bar{x})$ and the inequality $\|\tilde{W}_a\| \le \|\hat{W}_a\| + \|W_a^*\|$, we have

$$\begin{aligned}\|\delta\| &\le |\mathcal{R}|b_u\varepsilon_a^* + b_u b_r|\mathcal{R}|\|\hat{W}_a\|\|\phi_a\| + b_u b_r|\mathcal{R}|W_a^*\|\phi_a\| \\ &\quad + b_u|\mathcal{R}|\phi_a^*\phi_c^*W_c^*\|\hat{W}_a\| + b_u|\mathcal{R}|\phi_a^*\phi_c^*W_a^*\|\hat{W}_c\| \\ &\le b_u b_r|\mathcal{R}|\kappa^*\psi,\end{aligned} \tag{6.86}$$

where $\kappa^* \triangleq \max\{\varepsilon_a^*/b_r + W_a^*\phi_a^*, \phi_a^*(1+\phi_c^*W_c^*/b_r), \phi_a^*\phi_c^*W_a^*/b_r\}$, $\psi = 1 + \|\hat{W}_a\| + \|\hat{W}_c\|$.

This completes the proof. □

Theorem 6.20 (cf. [4]) *Under Assumptions 6.13, 6.14 and 6.17, considering the closed-loop system consisting of system (6.67) and the control u provided by (6.73), the weights tuning laws of the action NN and critic NN are*

$$\dot{\hat{W}}_a = \alpha_1(\phi_a(\mathcal{R} + |\mathcal{R}|\hat{W}_c^{\mathrm{T}}\phi_c)^{\mathrm{T}} - K_{W_a}|\mathcal{R}|\hat{W}_a), \tag{6.87}$$

$$\dot{\hat{W}}_c = -\alpha_2(|\mathcal{R}|\phi_c(\hat{W}_a^{\mathrm{T}}\phi_a)^{\mathrm{T}} + K_{W_c}|\mathcal{R}|\hat{W}_c), \tag{6.88}$$

and the tuning law of the adaptive parameter is

$$\dot{\hat{\kappa}} = \alpha_3\left(\mathcal{R}\psi \tanh\left(\frac{\mathcal{R}\psi}{\alpha}\right) - K_\kappa\hat{\kappa}\right), \tag{6.89}$$

where α_1, α_2, α_3, K_{W_a}, K_{W_c}, *and* K_κ *are the positive design parameters. If* $K_\kappa > 1/(b_u b_r)$, *then all the closed-loop signals are uniformly ultimately bounded.*

Proof Choose a Lyapunov function candidate as

$$L = \rho(r) + \frac{b_u}{2\alpha_1}\mathrm{tr}\{\tilde{W}_a^{\mathrm{T}}\tilde{W}_a\} + \frac{b_u}{2\alpha_2}\mathrm{tr}\{\tilde{W}_c^{\mathrm{T}}\tilde{W}_c\} + \frac{b_u b_r}{2\alpha_3}\tilde{\kappa}^2, \tag{6.90}$$

where $\rho(r) = \chi(\ln(1+e^{mr}) + \ln(1+e^{-mr}))/m$, $\tilde{\kappa} = \kappa^* - \hat{\kappa}$. The time derivative of (6.90) can be expressed as

$$\dot{L} = \mathcal{R}\dot{r} + \frac{b_u}{\alpha_1}\mathrm{tr}\{\tilde{W}_a^{\mathrm{T}}\dot{\tilde{W}}_a\} + \frac{b_u}{\alpha_2}\mathrm{tr}\{\tilde{W}_c^{\mathrm{T}}\dot{\tilde{W}}_c\} + \frac{b_u b_r}{\alpha_3}\tilde{\kappa}\dot{\tilde{\kappa}}. \tag{6.91}$$

Substituting (6.82) into (6.91), we have

$$\begin{aligned}\dot{L} = \mathcal{R}&\left\{-Kr + b_u\tilde{W}_a^{\mathrm{T}}\phi_a + b_u\left[\upsilon_r + \left(\frac{f_\upsilon(\bar{\upsilon})}{b_u} - 1\right)\left(f_\upsilon(\bar{\upsilon}) + \tilde{W}_a^{\mathrm{T}}\phi_a\right)\right]\right\}\\ &+ \mathcal{R}f_\upsilon(\bar{\upsilon})\varepsilon_a(\bar{x}) + \frac{b_u}{\alpha_1}\mathrm{tr}\{\tilde{W}_a^{\mathrm{T}}\dot{\tilde{W}}_a\} + \frac{b_u}{\alpha_2}\mathrm{tr}\{\tilde{W}_c^{\mathrm{T}}\dot{\tilde{W}}_c\} + \frac{b_u b_r}{\alpha_3}\tilde{\kappa}\dot{\tilde{\kappa}}.\end{aligned} \tag{6.92}$$

Applying (6.87) to (6.92), we have

$$\begin{aligned}\dot{L} = \mathcal{R}&\left\{-Kr + b_u\tilde{W}_a^{\mathrm{T}}\phi_a + b_u\left[\upsilon_r + \left(\frac{f_\upsilon(\bar{\upsilon})}{b_u} - 1\right)\left(f_\upsilon(\bar{\upsilon}) + \tilde{W}_a^{\mathrm{T}}\phi_a\right)\right]\right\}\\ &+ \mathcal{R}f_\upsilon(\bar{\upsilon})\varepsilon_a(\bar{x}) - b_u\mathrm{tr}\left\{\tilde{W}_a^{\mathrm{T}}(\phi_a(\mathcal{R} + |\mathcal{R}|\hat{W}_c^{\mathrm{T}}\phi_c)^{\mathrm{T}} - K_{W_a}|\mathcal{R}|\hat{W}_a)\right\}\\ &+ \frac{b_u}{\alpha_2}\mathrm{tr}\{\tilde{W}_c^{\mathrm{T}}\dot{\tilde{W}}_c\} + \frac{b_u b_r}{\alpha_3}\tilde{\kappa}\dot{\tilde{\kappa}}.\end{aligned} \tag{6.93}$$

Due to the fact that

$$\begin{aligned} b_u\mathrm{tr}\{\tilde{W}_a^{\mathrm{T}}\phi_a\mathcal{R}\} &= \mathcal{R}b_u\tilde{W}_a^{\mathrm{T}}\phi_a,\\ \mathrm{tr}\{\tilde{W}_a^{\mathrm{T}}\phi_a|\mathcal{R}|(\hat{W}_c^{\mathrm{T}}\phi_c)^{\mathrm{T}}\} &= |\mathcal{R}|\tilde{W}_a^{\mathrm{T}}\phi_a\phi_c^{\mathrm{T}}\hat{W}_c\\ &= |\mathcal{R}|(-\hat{W}_a^{\mathrm{T}}\phi_a\phi_c^{\mathrm{T}}W_a + W_a^{\mathrm{T}}\phi_a\phi_c^{\mathrm{T}}\hat{W}_c + \hat{W}_a^{\mathrm{T}}\phi_a\phi_c^{\mathrm{T}}\tilde{W}_c),\end{aligned} \tag{6.94}$$

and using (6.88), (6.93) can be rewritten as

$$\begin{aligned}\dot{L} \leq &-\mathcal{R}Kr + \mathcal{R}b_u\left[\upsilon_r + \left(\frac{f_\upsilon(\bar{\upsilon})}{b_u} - 1\right)\left(f_\upsilon(\bar{\upsilon}) + \tilde{W}_a^{\mathrm{T}}\phi_a\right)\right] + \mathcal{R}f_\upsilon(\bar{\upsilon})\varepsilon_a(\bar{x})\\ &+ b_u K_{W_a}|\mathcal{R}|\mathrm{tr}\{\tilde{W}_a^{\mathrm{T}}\hat{W}_a\} + b_u K_{W_c}|\mathcal{R}|\mathrm{tr}\{\tilde{W}_c^{\mathrm{T}}\hat{W}_c\}\\ &+ b_u|\mathcal{R}|(-\hat{W}_a^{\mathrm{T}}\phi_a(\phi_c^{\mathrm{T}}W_a) + W_a^{\mathrm{T}}\phi_a\phi_c^{\mathrm{T}}\hat{W}_c) + \frac{b_u b_r}{\alpha_3}\tilde{\kappa}\dot{\tilde{\kappa}}.\end{aligned} \tag{6.95}$$

With the robustifying term in (6.81), we have

$$\mathcal{R}\upsilon_r = -\frac{b_r}{1-b_r}|\hat{\kappa}|\mathcal{R}\psi\,\mathrm{sgn}(\mathcal{R}\psi) + \frac{b_r}{1-b_r}|\hat{\kappa}|\left(|\mathcal{R}\psi| - \mathcal{R}\psi\tanh\left(\frac{\mathcal{R}\psi}{\alpha}\right)\right). \tag{6.96}$$

According to Assumptions 6.13 and 6.14, we have

$$\left|\left(\frac{f_\upsilon(\bar{\upsilon})}{b_u} - 1\right)\left(\upsilon_r + \tilde{W}_a^{\mathrm{T}}\phi_a\right)\right| \leq b_r|\upsilon_r| + b_r\|\tilde{W}_a\|\|\phi_a\|. \tag{6.97}$$

Substituting (6.96) and (6.97) into (6.95), we have

$$\begin{aligned}\dot{L} \leq &-\mathcal{R}Kr + b_u\left[-\frac{b_r}{1-b_r}|\hat{\kappa}|\mathcal{R}\psi\,\mathrm{sgn}(\mathcal{R}\psi) + b_r\mathcal{R}|\upsilon_r|\right.\\ &\left.+\frac{b_r}{1-b_r}|\hat{\kappa}|\left(|\mathcal{R}\psi| - \mathcal{R}\psi\tanh\left(\frac{\mathcal{R}\psi}{\alpha}\right)\right)\right]\\ &+ b_u K_{W_a}|\mathcal{R}|\mathrm{tr}\{\tilde{W}_a^{\mathrm{T}}\hat{W}_a\} + b_u K_{W_c}|\mathcal{R}|\mathrm{tr}\{\tilde{W}_c^{\mathrm{T}}\hat{W}_c\}\\ &+ \delta + \frac{b_u b_r}{\alpha_3}\tilde{\kappa}\dot{\tilde{\kappa}}.\end{aligned} \tag{6.98}$$

According to Lemma 6.19 and the inequality $0 \leq |\mathcal{R}\psi| - \mathcal{R}\psi\tanh(\mathcal{R}\psi/\alpha) \leq \alpha$ for $\forall\alpha > 0$, $\mathcal{R}\psi \in \mathbb{R}$, we have

$$\begin{aligned}\dot{L} \leq &-\mathcal{R}Kr + b_u\left[-b_r|\hat{\kappa}|\mathcal{R}\psi\tanh\left(\frac{\mathcal{R}\psi}{\alpha}\right)\right.\\ &\left.+\frac{b_r}{1-b_r}|\hat{\kappa}|\left(|\mathcal{R}\psi| - \mathcal{R}\psi\tanh\left(\frac{\mathcal{R}\psi}{\alpha}\right)\right)\right] + b_u K_{W_a}|\mathcal{R}|tr\{\tilde{W}_a^{\mathrm{T}}\hat{W}_a\}\\ &+ b_u K_{W_c}|\mathcal{R}|\mathrm{tr}\{\tilde{W}_c^{\mathrm{T}}\hat{W}_c\} + b_u b_r\kappa^*\left(\alpha + \mathcal{R}\psi\tanh\left(\frac{\mathcal{R}\psi}{\alpha}\right)\right)\\ &- b_u b_r\tilde{\kappa}\mathcal{R}\psi\tanh\left(\frac{\mathcal{R}\psi}{\alpha}\right) + b_u b_r K_\kappa\tilde{\kappa}\hat{\kappa}\end{aligned}$$

$$\leq -\mathcal{R}Kr + \frac{b_u b_r}{1-b_r}|\hat{\kappa}|\alpha + b_u b_r \kappa^* \alpha + b_u K_{W_a}|\mathcal{R}|\text{tr}\{\tilde{W}_a^{\mathrm{T}} \hat{W}_a\}$$
$$+ b_u K_{W_c}|\mathcal{R}|\text{tr}\{\tilde{W}_c^{\mathrm{T}} \hat{W}_c\} + b_u b_r K_\kappa \tilde{\kappa}\hat{\kappa}. \tag{6.99}$$

Let $b^* = b_u - b_l$. Using the relation $b_u b_r/(1-b_r) = (b_u/b_l)b^*$ and the inequality $|\hat{\kappa}| \leq |\tilde{\kappa}| + \kappa^*$, $\dot{L}$ is further derived as

$$\dot{L} \leq -\mathcal{R}Kr + \frac{b_u}{b_l}b^*|\tilde{\kappa}|\alpha + \frac{b_u}{b_l}b^*\kappa^*\alpha + b_u b_r \kappa^* \alpha$$
$$+ b_u K_{W_a}|\mathcal{R}|\text{tr}\{\tilde{W}_a^{\mathrm{T}} \hat{W}_a\} + b_u K_{W_c}|\mathcal{R}|\text{tr}\{\tilde{W}_c^{\mathrm{T}} \hat{W}_c\} + b_u b_r K_\kappa \tilde{\kappa}\hat{\kappa}. \tag{6.100}$$

Since $\text{tr}\{\tilde{W}_a^{\mathrm{T}} \hat{W}_a\} \leq -\|\tilde{W}_a\|^2/2 + \|W_a\|^2/2$, $\text{tr}\{\tilde{W}_c^{\mathrm{T}} \hat{W}_c\} \leq -\|\tilde{W}_2\|^2/2 + \|W_c\|^2/2$, $\tilde{\kappa}\hat{\kappa} \leq -|\tilde{\kappa}|^2/2 + \kappa^{*2}/2$, $(b_u/b_l)b^*|\tilde{\kappa}|\alpha \leq ((b_u/b_l)b^*\alpha)^2/2 + |\tilde{\kappa}|^2/2$, we have

$$\dot{L} \leq -\mathcal{R}Kr - \frac{b_u K_{W_a}|\mathcal{R}|}{2}\|\tilde{W}_a\|^2 - \frac{b_u K_{W_c}|\mathcal{R}|}{2}\|\tilde{W}_c\|^2$$
$$- \frac{b_u b_r}{2}\left(K_\kappa - \frac{1}{b_u b_r}\right)|\tilde{\kappa}|^2 + D, \tag{6.101}$$

where

$$D = \frac{b_u K_{W_a}|\mathcal{R}|}{2}\|W_a\|^2 + \frac{b_u K_{W_c}|\mathcal{R}|}{2}\|W_c\|^2 + \frac{b_u b_r K_\kappa}{2}\kappa^{*2}$$
$$+ \frac{1}{2}\left(\frac{b_u}{b_l}b^*\alpha\right)^2 + \frac{b_u}{b_l}b^*\kappa^*\alpha + b_u b_r \kappa^* \alpha. \tag{6.102}$$

Because $\mathcal{R}r > 0$ for any $r \neq 0$ and $\mathcal{R} \in [-\chi, \chi]$ and using Assumption 6.17, (6.101) becomes

$$\dot{L} \leq -\chi K|r| - \frac{b_u K_{W_a}|\mathcal{R}|}{2}\|\tilde{W}_a\|^2 - \frac{b_u K_{W_c}|\mathcal{R}|}{2}\|\tilde{W}_c\|^2$$
$$- \frac{b_u b_r}{2}\left(K_\kappa - \frac{1}{b_u b_r}\right)|\tilde{\kappa}|^2 + D_M, \tag{6.103}$$

where $\lambda_M \triangleq b_u K_{W_a}\chi W_a^{*2}/2 + b_u K_{W_c}\chi W_c^{*2}/2 + (b_u/b_l)b^*\kappa^*\alpha + b_u b_r \kappa^*\alpha + ((b_u/b_l)b^*\alpha)^2/2$.

With K_κ picked so that $K_\kappa > 1/(b_u b_r)$ is satisfied, the time derivative of L is guaranteed to be negative as long as the following hold:

$$|\tilde{r}| \geq \sqrt{\frac{D_M}{K\chi}}, \tag{6.104}$$

or

$$\|\tilde{W}_a\| \geq \sqrt{\frac{2D_M}{b_u K_{W_a}|\mathcal{R}|}} \geq \sqrt{\frac{2D_M}{b_u K_{W_a}\chi}}, \tag{6.105}$$

or

$$\|\tilde{W}_c\| \geq \sqrt{\frac{2D_M}{b_u K_{W_c} |\mathcal{R}|}} \geq \sqrt{\frac{2D_M}{b_u K_{W_c} \chi}}, \tag{6.106}$$

or

$$|\tilde{\kappa}| \geq \sqrt{\frac{2\lambda_M}{b_u b_r (K_\kappa - \frac{1}{b_u b_r})}}. \tag{6.107}$$

Therefore, according to the standard Lyapunov extensions [11], this demonstrates that the filtered tracking error, the weights estimation errors of the critic NN and action NN are uniformly ultimately bounded. □

Remark 6.21 It is interesting to note from (6.104)–(6.107) that arbitrarily small $|\tilde{r}|$, $\|\tilde{W}_a\|$, $\|\tilde{W}_c\|$ and $|\tilde{\kappa}|$ may be achieved by selecting the large fixed gain K, K_{W_a}, K_{W_c}, and K_κ or the critic slope gain χ, respectively.

6.3.3 Simulations

Example 6.22 Consider the following continuous-time nonaffine nonlinear system:

$$\begin{aligned}
\dot{x}_1 &= x_2, \\
\dot{x}_2 &= x_1 x_2^2 + x_2 e^{-1-x_1^2} - x_1 x_2 + (2 + 0.3\sin x_2^2)u + \cos(0.1u). \\
y &= x_1.
\end{aligned}$$

The control objective is to make the output y follow the desired trajectory y_d. The reference signal is selected as $y_d = \sin(t) + \cos(0.5t)$. We assume that the control design is performed by using the approximate model $\hat{f}(x, u) = 10u$. The controller parameters are chosen as $K = 30$, $\Lambda = 20$, $b_r = 1/3$, and $\chi = 20$. The critic NN and action NN consist of five and six hidden-layer nodes, respectively. The activation functions are selected as sigmoidal activation functions. The first-layer weights of both action NN and critic NN are selected randomly over an internal of $[-1, 1]$. The threshold weights for the first layer of both action NN and critic NN are uniformly randomly distributed between -10 and 10. The second-layer weights of action NN $\hat{W}_a$ is uniformly randomly initialized over an internal of $[-1, 1]$. The second-layer weights of the critic NN $\hat{W}_c$ is initialized at zero. The parameters of the critic signal are selected as $m = 2$. For weights and adaptive parameters updating, the design parameters are selected as $\alpha_1 = \alpha_2 = \alpha_3 = K_{W_a} = K_{W_c} = 0.1$, and $K_\kappa = 80$, which satisfies $K_\kappa > 1/(b_u b_r)$. Then, for the initial states $x(0) = [0, 0]^{\mathrm{T}}$, we apply the present controller to the system for 100 s. The simulation results are shown in Figs. 6.11–6.16. From Fig. 6.11, it is observed that the system output

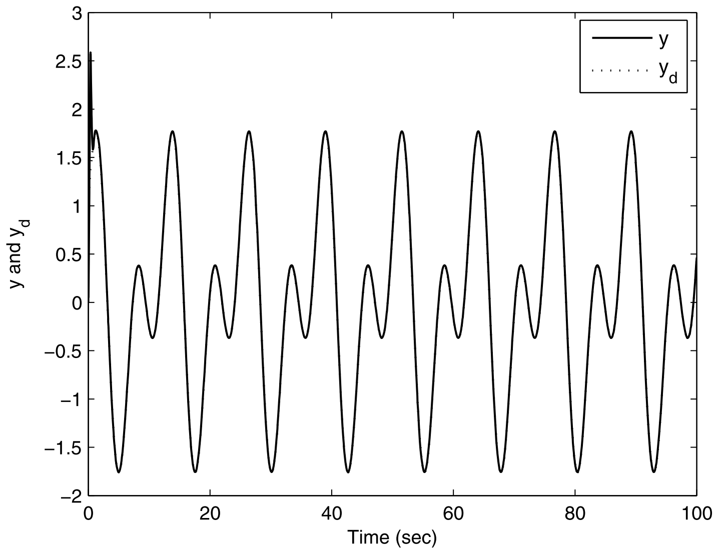

Fig. 6.11 The trajectories of y_d and y

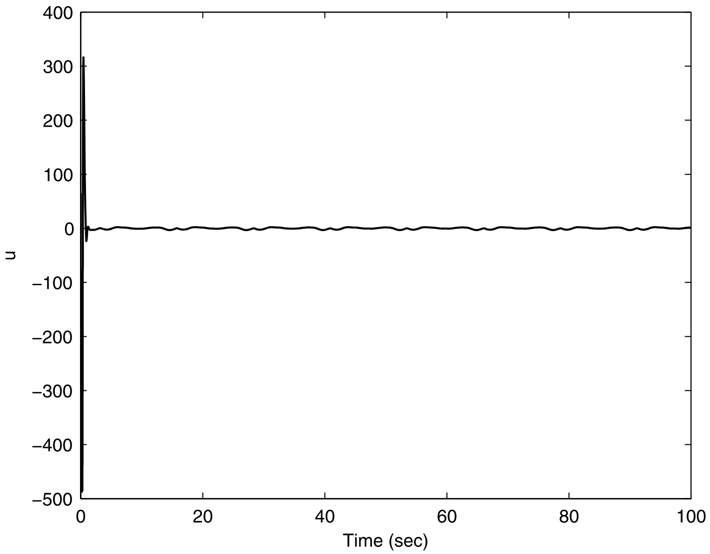

Fig. 6.12 The trajectory of control input u

y tracks the desired trajectory y_d fast and well. Figures 6.12, 6.13, 6.14 clearly show that the control input u is bounded. Figure 6.15 displays that the critic signal $\hat{\mathcal{R}}_n$ is bounded. From Fig. 6.16, it is observed that the adaptive parameter $\hat{\kappa}$ is

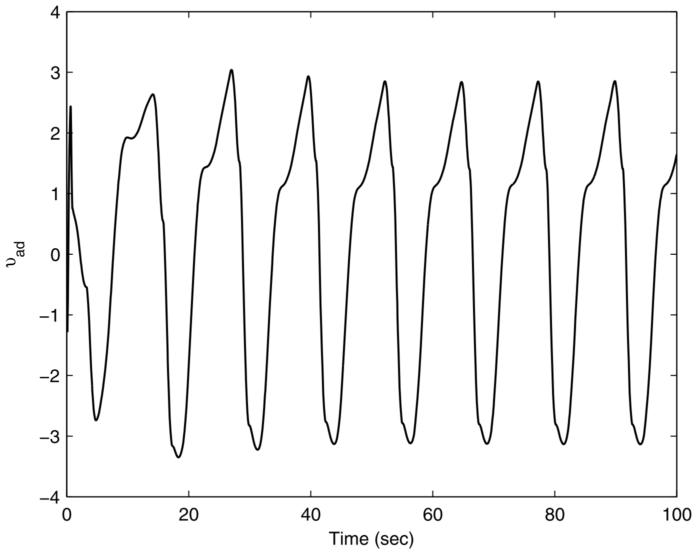

Fig. 6.13 The trajectory of υ_{ad}

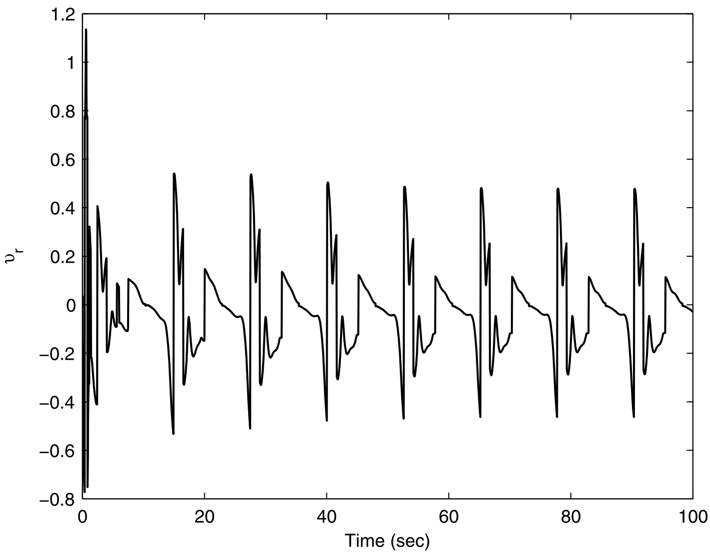

Fig. 6.14 The trajectory of υ_r

bounded. The simulation results show that the present ADP method can perform successful control and achieve the desired performance for the nonaffine nonlinear system.

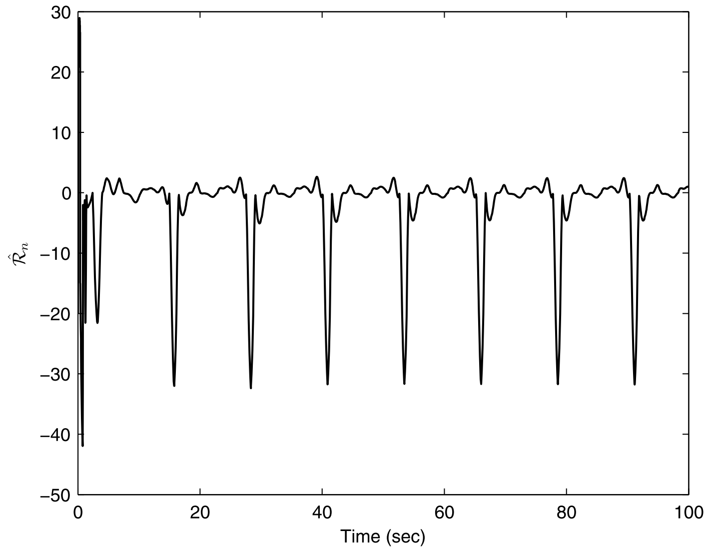

Fig. 6.15 The trajectory of $\hat{\mathcal{R}}_n$

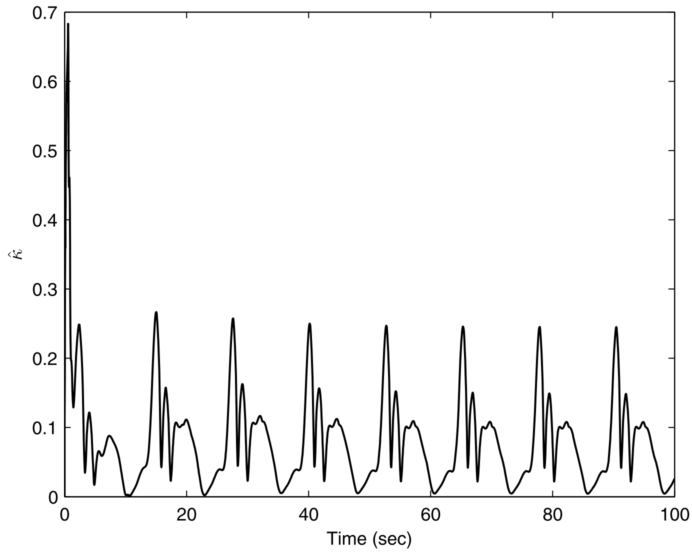

Fig. 6.16 The trajectory of $\hat{\kappa}$

6.4 Summary

In this chapter, we investigated the optimal feedback control problems of a class of unknown general continuous-time nonlinear systems and a class of continuous-time nonaffine nonlinear systems via the ADP approach. In Sect. 6.2, we developed an effective scheme to design the data-based optimal robust tracking controller for unknown general continuous-time nonlinear systems, in which only input/output data are required instead of the system dynamics. In Sect. 6.3, a novel ADP based robust neural network controller was developed for a class of continuous-time nonaffine nonlinear systems, which is the first attempt to extend the ADP approach to continuous-time nonaffine nonlinear systems. Numerical simulations have shown that the present methods are effective and can be used for a quite wide class of continuous-time nonlinear systems.

References

1. Al-Tamimi A, Lewis FL, Abu-Khalaf M (2007) Model-free Q-learning designs for linear discrete-time zero-sum games with application to H_∞ control. Automatica 43:473–481
2. Calise AJ, Hovakimyan N, Idan M (2001) Adaptive output feedback control of nonlinear systems using neural networks. Automatica 37:1201–1211
3. Chellaboina V, Haddad WM (2002) A unification between partial stability and stability theory for time-varying systems. IEEE Control Syst Mag 22:66–75
4. Cui LL, Luo YH, Zhang HG (2011) Adaptive critic design based robust neural network control for a class of continuous-time nonaffine nonlinear system. In: Proceedings of international conference on modelling, identification and control, Shanghai, pp 26–29
5. Ge S, Zhang J (2003) Neural-network control of nonaffine nonlinear system with zero dynamics by state and output feedback. IEEE Trans Neural Netw 14:900–918
6. Hanselmann T, Noakes L, Zaknich A (2007) Continuous-time adaptive critics. IEEE Trans Neural Netw 3:631–647
7. Khalil HK (2002) Nonlinear system. Prentice Hall, Englewood Cliffs
8. Kim N, Calise AJ (2007) Several extensions in methods for adaptive output feedback control. IEEE Trans Neural Netw 18:482–494
9. Kim YH, Lewis FL (2000) Reinforcement adaptive learning neural-net-based friction compensation control for high speed and precision. IEEE Trans Control Syst Technol 8:118–126
10. Kuljaca O, Lewis FL (2003) Adaptive critic design using non-linear network structure. Int J Adapt Control Signal Prog 17:431–445
11. Lewis FL, Jagannathan S, Yesildirek A (1999) Neural network control of robot manipulators and nonlinear systems. Taylor & Francis, London
12. Lin CK (2005) Adaptive critic autopilot design of bank-to-turn missiles using fuzzy basis function networks. IEEE Trans Syst Man Cybern, Part B, Cybern 35:197–206
13. Lin CK (2009) H_∞ reinforcement learning control of robot manipulators using fuzzy wavelet networks. Fuzzy Sets Syst 160:1765–1786
14. Liu ZW, Zhang HG, Zhang QL (2010) Novel stability analysis for recurrent neural networks with multiple delays via line integral-type L-K functional. IEEE Trans Neural Netw 21:1710–1718
15. Murray JJ, Cox CJ, Lendaris GG, Saeks R (2002) Adaptive dynamic programming. IEEE Trans Syst Man Cybern, Part B, Cybern 32:140–153
16. Rubio JDJ, Yu W (2007) Stability analysis of nonlinear system identification via delayed neural networks. IEEE Trans Circuits Syst II, Express Briefs 54:161–195
17. Vamvoudakis KG, Lewis FL (2010) Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem. Automatica 46:878–888
18. Vrabie D, Lewis FL (2009) Neural network approach to continuous-time direct adaptive optimal control for partially unknown nonlinear system. Neural Netw 22:237–246

19. Vrabie D, Pastravanu O, Abu-Khalaf M, Lewis FL (2009) Adaptive optimal control for continuous-time linear systems based on policy iteration. Automatica 45:477–484
20. Wang ZS, Zhang HG, Yu W (2009) Robust stability of Cohen–Grossberg neural networks via state transmission matrix. IEEE Trans Neural Netw 20:169–174
21. Wang ZS, Zhang HG, Jiang B (2011) LMI-based approach for global asymptotic stability analysis of recurrent neural networks with various delays and structures. IEEE Trans Neural Netw 22:1032–1045
22. Zhang HG, Wang ZS, Liu DR (2008) Global asymptotic stability of recurrent neural networks with multiple time-varying delays. IEEE Trans Neural Netw 19:855–873
23. Zhang HG, Li M, Yang J (2009) Fuzzy model-based robust networked control for a class of nonlinear systems. IEEE Trans Syst Man Cybern, Part A, Syst Hum 39:437–447
24. Zhang HG, Liu ZW, Huang GB (2010) Novel weighting-delay-based stability criteria for recurrent neural networks with time-varying delay. IEEE Trans Neural Netw 21:91–106
25. Zhang HG, Cui LL, Zhang X, Luo YH (2011) Data-driven robust approximate optimal tracking control for unknown general nonlinear systems using adaptive dynamic programming method. IEEE Trans Neural Netw 22(12):2226–2236
26. Zhang HG, Xie XP, Tong SC (2011) Homogeneous polynomially parameter-dependent H_∞ filter designs of discrete-time fuzzy systems. IEEE Trans Syst Man Cybern, Part B, Cybern 41:1313–1322