Neural Computing and Applications (2023) 35:23987–23996
https://doi.org/10.1007/s00521-023-08993-0

ORIGINAL ARTICLE

Reinforcement learning-based robust optimal tracking control for disturbed nonlinear systems

Zhong-Xin Fan[1,2] · Lintao Tang[3] · Shihua Li[1,2] · Rongjie Liu[3]

Received: 29 May 2023 / Accepted: 22 August 2023 / Published online: 12 September 2023

Abstract
This paper concludes a robust optimal tracking control law for a class of nonlinear systems. A characteristic of this paper is that the designed controller can guarantee both robustness and optimality under nonlinearity and mismatched disturbances. Optimal controllers for nonlinear systems are difficult to obtain, hence a reinforcement learning method is adopted with two neural networks (NNs) approximating the cost function and optimal controller, respectively. We designed weight update laws for critic NN and actor NN based on gradient descent and stability, respectively. In addition, matched and mismatched disturbances are estimated by fixed-time disturbance observers and an artful transformation based on backstepping method is employed to convert the system into a filtered error nonlinear system. Through a rigorous analysis using the Lyapunov method, we demonstrate states and estimation errors remain uniformly ultimately bounded. Finally, the effectiveness of the proposed method is verified through two illustrative examples.

Keywords Reinforcement learning · Actor-critic neural network · Fixed-time disturbance observer · Robust optimal control

1 Introduction

The objective of optimal tracking control is to develop a controller that ensures the system's output tracks a specified reference signal, while minimizing a specific performance index. This field has earned significant attention and research, finding applications in practical domains such as chaotic systems, helicopters, permanent magnet synchronous motors, dispatch and electric vehicles [1–5]. Optimal control techniques rely on the principles of Pontryagin's minimum principle. In the case of linear systems, the optimal control involves solving the algebraic Riccati equation, as suggested in the work by [6]. For the nonlinear systems, the optimal control necessitates the solution of the nonlinear Hamilton-Jacobi-Bellman (HJB) equation. Despite the practical utility of optimal control, the conventional methodology encounters a significant challenge, namely, the difficulty of solving the nonlinear HJB equation for higher-order systems [7–10].

In recent years, numerous efforts have been made to obtain the optimal controller, including inverse optimal control, θ-D techniques, numerical approximation methods, and others [11, 13, 14]. The inverse optimal control method, presented in [11, 12], offers a solution that avoids the need to solve the HJB equations. For nonlinear systems, a suboptimal control approach was proposed in [13]. Another approach, described in [14], employed a θ-D approximation method to solve the HJB equation by transforming it into state-dependent Lyapunov equations. It is important to note that these methods, although effective, are typically performed offline. Consequently, when there are changes in the system parameters, there may be

✉ Rongjie Liu
rliu3@fsu.edu

Zhong-Xin Fan
zhxfan@seu.edu.cn

Lintao Tang
lt20ca@fsu.edu

Shihua Li
lsh@seu.edu.cn

[1] School of Automation, Southeast University, Sipailou, Nanjing 210096, Jiangsu, China

[2] Key Laboratory of Measurement and Control of Complex Systems of Engineering, Ministry of Education, Southeast University, Sipailou, Nanjing 210096, Jiangsu, China

[3] Department of Statistics, Florida State University, Tallahassee, FL 32304, USA

fluctuations in the control effectiveness. To address this issue, researchers have explored the integration of reinforcement learning and adaptive control with optimal control [7, 15–21].

Approximate dynamic programming (ADP), proposed by [7] in 1992, utilizes function approximation structures to approximate the cost function and control strategy in the dynamic programming equation. ADP has been developed in subsequent works [15–17] using neural networks (NNs) to achieve optimal tracking control. These methods have been thoroughly studied and widely adopted [18, 24]. Furthermore, advancements in hardware have paved the way for data-driven approaches in optimal control. For example, [22] introduced a computational adaptive optimal controller for linear systems with completely unknown dynamics. Nonlinear adaptive optimal control was achieved through value iteration and ADP, as described in [23].

Inspired by this, we have incorporated the principles of adaptive and reinforcement learning to develop efficient tracking controllers using an actor-critic approach. Nevertheless, previous studies such as [25, 26] have highlighted a limitation of optimal tracking control, which involves the introduction of a discount factor into the performance index. This factor is intended to prevent the index from growing indefinitely, but it can hinder the convergence of the system state to zero. To address this issue, our paper proposes a reinforcement learning-based tracking control technique that utilizes a filtered error system, thereby eliminating the need for a discount factor.

In practical systems, the presence of disturbances is an inevitable issue [27, 28, 35]. These disturbances encompass both internal environmental factors, such as unmodeled dynamics, perturbed model parameters, and structural perturbations, as well as external environmental disturbances [37]. To achieve desired control outcomes, including improved disturbance rejection, fast dynamic response, and minimal steady-state error, it is crucial to explore high reliability controllers. Extensive research has been conducted on various anti-disturbance control methods, such as robust control [29], sliding mode control [30, 31], and output regulation theory [32]. Among these methods, two approaches have gained attention for their ability to achieve fast disturbance suppression based on system dynamics: disturbance observer-based control and active disturbance rejection control [33–35]. By employing disturbance observers or extended state observers to estimate and actively compensate for disturbances, their influence can be effectively mitigated [35].

However, mismatched disturbances are difficult to handle, as highlighted in [36, 37]. In [37], the authors proposed a composite control strategy based on the backstepping method for higher-order nonlinear systems with non-vanishing disturbances. By incorporating estimation information of the disturbance at each step of the virtual control, output is regulated to 0. While this method effectively handles mismatched disturbances, it is not optimal due to two reasons. Firstly, nonlinearity is subtracted at each step of the virtual control process. Secondly, the gain of the virtual control is artificially assigned and only satisfies the condition for making the derivative of the Lyapunov function negative definite. Therefore, we employ the concept of backstepping to construct a filtered error system that retains the nonlinear terms, ensuring optimality in dealing with mismatched disturbances.

Furthermore, the majority of existing studies focus on achieving asymptotic estimates of disturbances, implying that estimation errors persist even as the system converges. To mitigate the impact of disturbances, researchers have proposed fixed-time observers [38–40]. This approach involves estimating unknown disturbances within a predetermined time period, thereby minimizing their subsequent effects. In our study, we also employ a fixed-time disturbance observer (FTDOB) to estimate disturbances and reduce their influence on the neural network training process.

Therefore, this paper aims to address the limitations of existing optimal control methods and anti-disturbance methods in order to tackle more complex scenarios. The primary contributions of this paper are as follows:

- Two neural networks are utilized to implement an actor-critic network, enabling the approximation of both the optimal control and cost function.
- The fixed-time algorithm is employed in the design of the observer, allowing for the estimation of disturbances over a predetermined time interval, thereby enhancing the reliability of the control strategy.
- Filtered error systems are constructed to attain an optimal controller for high-order nonlinear systems affected by mismatched disturbances.

The rest of the paper are organized as follows. In Sect. 2, system description and some necessary definitions are given. Section 3 concludes the main results about disturbance observer design and controller design. Simulation examples are given in Sect. 4 and conclusion is given in Sect. 5.

2 System descriptions and some preliminaries

Consider the following disturbed nonlinear system,

$$\begin{cases} \dot{x}_i = x_{i+1} + f_i + d_i, \quad i = 1, 2, \ldots, n-1, \\ \dot{x}_n = f_n + u + d_n, \end{cases} \tag{1}$$

where x_i, d_i, f_i, $i = 1, 2, \ldots, n$ denote system states, disturbances and nonlinear functions, u is the control input. Assuming complete state information is available.

Assumption 1 Assuming there exists a small enough constant ξ such that $\|\dot{d}\| < \xi$.

Here, we recall the optimal control theory [6]. For the nominal system, i. e., we do not consider the disturbance here, a cost function is given as

$$J = \int_0^\infty [Q(x) + u^{\mathsf{T}} R u] \mathrm{d}t, \tag{2}$$

where $Q(x)$ is positive definite function and R is symmetric positive definite constant matrix. Define $\frac{\partial J}{\partial x} = \nabla J$ and choose the Hamilton function as $H = \nabla J^{\mathsf{T}} \dot{x} + Q + u^{\mathsf{T}} R u$. Then, optimal value function J^* meets $0 = \min_u [H(x, u, \nabla J^*)]$. With optimal control policy u^*, the HJB equation becomes

$$0 = Q + u^{*\mathsf{T}} R u^* + \nabla J^{*\mathsf{T}} (f + g u^*). \tag{3}$$

Then, we have the optimal control input u^* as

$$u^* = \arg\min_u [H(x, u, \nabla J^*)] = -\frac{1}{2} R^{-1} g^{\mathsf{T}} \nabla J^*. \tag{4}$$

The existing optimal control methods faces two challenges: (1) robustness in the presence of disturbances, especially in the presence of mismatched disturbances; (2) complex nonlinear HJB equation, given that the solution is very resource-intensive. Hence, we proposed a robust optimal control strategy based on NNs and disturbance observers, which will be detailed given in Sect. 3. Next, we provide one definition for the latter process.

Definition 1 The equilibrium x_e of system (1) is uniformly ultimately bounded (UUB) if there is a compact set $S \subset \mathbb{R}^n$, and for any initial value x_0 that belongs to that compact set, initial time t_0, there is an upper bound B and a time $T(B, x_0)$ such that $\|x(t) - x_e\| \leq B$ for all $t > t_0 + T$.

3 Main results

The classic control method usually adopts the idea of feedback control plus feedforward control [35], but it has the following two shortcomings: (1) The asymptomatically convergent observer will cause the estimation error to persist. (2) Feedback control can only stabilize the system with not optimality. This paper avoids these shortcomings by fusing fixed-time estimation with reinforcement learning. The accompanying Fig. 1 visually represents the core concepts discussed in this paper. The output of the system is directly used as the input of the disturbance observer. By choosing the observer gain reasonably, the complete tracking of the disturbance can be realized in any fixed time. Then, the original with disturbance estimation is transformed into a filter error system, which enables us to deal with mismatched disturbance well. Under the framework of optimal control, reinforcement learning methods relying on actor and critic NNs are proposed. By training the NN, the optimal controller of the error system is obtained.

Firstly, we design the fixed-time disturbance observers. With the disturbance estimation in hand, a filtered error system is then transformed.

3.1 Fixed-time disturbance observer design

The fixed-time disturbance observer is designed for each channel as

$$\begin{cases} \dot{z}_{i1} = z_{i2} - \lambda_1 (z_{i1} - x_i)^{\alpha 1} - \lambda_2 (z_{i1} - x_i)^{\beta 1} + x_{i+1} + f_i, \\ \dot{z}_{i2} = -\lambda_3 (z_{i1} - x_i)^{\alpha 2} - \lambda_4 (z_{i1} - x_i)^{\beta 2}, \end{cases} \tag{5}$$

where $i = 1, 2, \ldots, n$. z_{i1}, z_{i2} are estimations of x_i and d_i, λ_1, λ_2, λ_3, λ_4 are observer gains to be designed, α_1, α_2, β_1, β_2 are observer internal parameters.

Theorem 1 *Given system (1) if the observer gain is chosen properly, the disturbance can be estimated in a fixed time T_d, which is independent of the initial values.*

Proof Define the estimation error as $e_{i1} = x_1 - z_{i1}$, $e_{i2} = d_i - z_{i2}$. Derivation of e_{i1} and e_{i2} along time gives

$$\begin{cases} \dot{e}_{i1} = e_{i2} - \lambda_1 (e_1)^{\alpha 1} - \lambda_2 (e_1)^{\beta 1}, \\ \dot{e}_{i2} = -\lambda_3 (e_1)^{\alpha 2} - \lambda_4 (e_1)^{\beta 2} + \dot{d}_i. \end{cases} \tag{6}$$

As long as the observer gain is chosen carefully, then the estimation error is fixed-time convergent, and can be written as $\dot{e} = \Lambda(e) + D$, $D = [0, \quad \dot{d}_i]^T$. The rest proof is similar to [31] and is omitted here. □

Under the designed observer, the mismatched disturbance can be handled. With the help of backstepping method, the filtered error is obtained as $\dot{z}_1 = x_2 + f_1 + d_1 - \dot{r}$, where r is reference signal. Here, we denote $z_2 = x_2 - x_2^*$, choose $x_2^* = -k_1 z_1 - \hat{d}_1 + \dot{r}$, then $\dot{z}_1 = \dot{x}_1 - \dot{r} = z_2 - k_1 z_1 + f_1 + e_1$. Likewise, we have

$$\begin{cases} \dot{z}_i = z_{i+1} - k_i z_i + f_i + e_i, i = 1, \ldots, n-1, \\ \dot{z}_n = u_o + f_n + e_n. \end{cases} \tag{7}$$

Then (7) is rewritten as

$$\dot{Z} = F(Z) + G u_o, \tag{8}$$

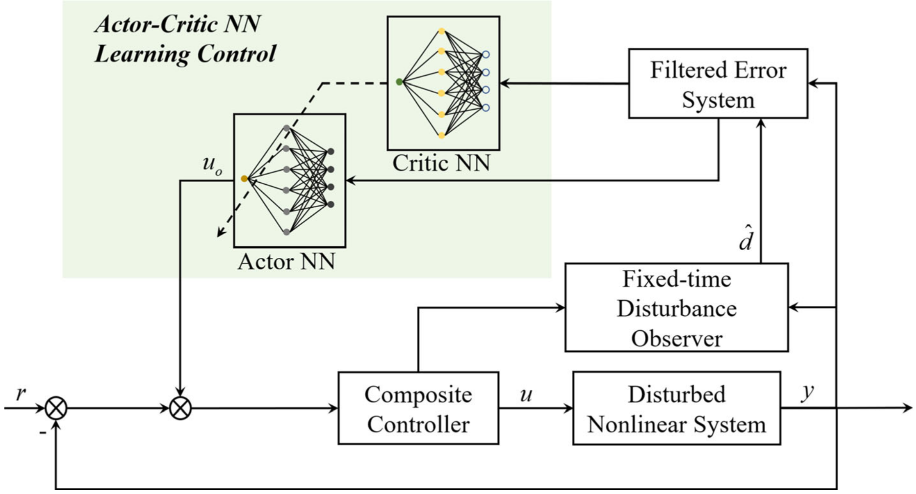

Fig. 1 Reinforcement learning based robust optimal control strategy. The fixed-time observer provides an accurate estimate of the disturbance. By compensating it back to the original system, filtered error systems are constructed. Actor and critic NN is used to achieve reinforcement learning optimal control

where $Z = [z_1, z_2, \ldots, z_n]^{\mathsf{T}}$, $F(Z) = [z_2 + f_1 - k_1 z_1, \cdots, f_n - k_n z_n]^{\mathsf{T}}$, $G = [0, 0, \ldots, 1]^T$.

Remark 1 Subtracting the nonlinear in backstepping method will lead to a nonoptimal controller as the nonlinearity may be actually beneficial in meeting the stabilization and/or performance objectives [11].

Remark 2 During the actual production process, the controlled system often encounters abrupt disturbances that can be characterized as lumped disturbance [35]. These types of disturbances do not satisfy the assumption we initially made (referred to as Assumption 1). Nevertheless, the proposed control strategy exhibits the capability to stabilize the system and demonstrates a certain level of robustness. This is attributed to the fact that even in the presence of sudden disturbance changes, the designed observer is able to estimate the disturbance at a fixed time. It is worth noting that the nonlinear function employed in the controller design is represented as $f + e$. However, since the term e exists only momentarily and eventually diminishes to zero, the overall effect on the controller's performance is minimal.

According to the former section, we define $\frac{\partial J}{\partial Z} = \nabla J$ and the Hamilton function is chosen as $H = \nabla J^{\mathsf{T}} \dot{Z} + Z^{\mathsf{T}} Q Z + u_o^{\mathsf{T}} R u_o$. Then, we have the optimal control as $u^* = \arg\min_{u_o}[H(Z, u_o, \nabla J^*)] = -\frac{1}{2} R^{-1} g^{\mathsf{T}} \nabla J^*$, satisfying $0 = Q + u_o^{*\mathsf{T}} R u_o^* + \nabla J^{*\mathsf{T}}(F + G u_o^*)$.

3.2 Critic NN design

The cost function is approximated by a critic neural network,

$$J = W^{\mathsf{T}} \phi(Z) + \epsilon(Z), \tag{9}$$

where W denotes the ideal neuron weights, $\phi(Z) : R^n \to R^N$ is the NN activation function vector, N stands for the number of neurons in the hidden layer, $\epsilon(Z)$ is the approximation error. As $N \to \infty$, it has $\epsilon(Z) \to 0$. As a result of the unknown nature of neural network weight W, the output of the neural network can be expressed as

$$\hat{J} = \hat{W}_c^{\mathsf{T}} \phi(Z), \tag{10}$$

where $\hat{W}_c$ is the estimation of W.

Considering (9) and (10), the corresponding Hamilton functions are rewritten as

$$H(Z, u_o, W) = W^{\mathsf{T}} \nabla\phi(F + G u_o) + Q + u_o^{\mathsf{T}} R u_o + \upsilon_H \tag{11}$$

and

$$H(Z, u_o, \hat{W}_c) = \hat{W}_c^{\mathsf{T}} \nabla\phi(F + G u_o) + Q + u_o^{\mathsf{T}} R u_o, \tag{12}$$

where $\upsilon_H = \nabla\epsilon(F + G u_o)$.

Define critic NN approximation error $\tilde{W}_c = W - \hat{W}_c$, then we have

$$e_H = H(Z, u_o, W) - H(Z, u_o, \hat{W}_c) = \tilde{W}_c^{\mathsf{T}} \nabla\phi_1(F + G u_o) + \upsilon_H. \tag{13}$$

Given any admissible control policy, it is desired to select $\hat{W}_c$ to minimize the quadratic error

$$E = \frac{1}{2} e_H^{\mathsf{T}} e_H. \tag{14}$$

The normalized gradient algorithm is adopted to tune the critic weights

$$\dot{\hat{W}}_c = -a_1 \frac{\partial E}{\partial \hat{W}_c} = -a_1 \frac{\sigma_1}{(\sigma_1^{\mathsf{T}}\sigma_1 + 1)^2}[\sigma_1^{\mathsf{T}}\hat{W}_c + Q + u_o^{\mathsf{T}} R u_o], \tag{15}$$

where $\sigma_1 = \nabla\phi(F + Gu_o)$, $(\sigma_1^T\sigma_1 + 1)^2$ is used for normalization, a_1 is scalar to be designed.

As mentioned in [2, 18], the identification of the critic parameter needs to fulfill the persistent excitation (PE) condition. In order to satisfy this condition, there are numerous options available for the signal selection, as long as the PE condition outlined in [18] is met.

3.3 Actor NN design

According to (3), we know the optimal control could be $-\frac{1}{2}R^{-1}G^{\mathsf{T}}(\nabla\phi^{\mathsf{T}}W + \nabla\epsilon)$. Due the parameter W is unknown, here we utilize an actor NN to approximate the control input. Then, the controller is represented as

$$\hat{u}_o = -\frac{1}{2}R^{-1}G^{\mathsf{T}}\nabla\phi^{\mathsf{T}}\hat{W}_a, \tag{16}$$

where $\hat{W}_a$ denotes the estimated value of W.

Similarly, $\hat{W}_a$ should be designed to approach W as closely as possible. Here, the tuning law of the actor NN is

$$\dot{\hat{W}}_a = -a_2\{F_2\hat{W}_a - F_2\hat{W}_c - \frac{1}{4}\bar{D}_1\hat{W}_a m^{\mathsf{T}}\hat{W}_c\}, \tag{17}$$

where $\bar{D}_1 = \nabla\phi G R^{-1} G^{\mathsf{T}} \nabla\phi^{\mathsf{T}}$, $m = \frac{\sigma_2}{(\sigma_2^{\mathsf{T}}\sigma_2+1)^2}$, $\sigma_2 = \nabla\phi(F + G\hat{u}_o)$, a_2 is scalar to be designed.

The following is the online algorithm that facilitates the simultaneous tuning of the actor NN and the critic NN.

Algorithm 1 Actor-Critic Learning-based Algorithm

```
Require: reference signal $r$, $\alpha_1 \in (0,1)$, $\alpha_2 \in (0,1)$, $\beta_1 > 1$, $\beta_2 > 1$, $\begin{bmatrix} -\lambda_1 & 1 \\ -\lambda_3 & 0 \end{bmatrix}$
    and $\begin{bmatrix} -\lambda_2 & 1 \\ -\lambda_4 & 0 \end{bmatrix}$ are Hurwitz matrices, an initial stabilizing control law $u$,
    $k_i > 0$, $i = 1, 2, ..., i-1$.
Ensure: $z_1 = 0$, i. e., $x_1 \to r$
Step 1: FTDOB activation
if $d_i \neq 0$ then
    (5) is activated and the estimations are obtained in fixed-time
else
    Go Step 2
end if
Step 2: Filtered error system transformation
    Based on (5), filtered error system is obtained as (8)
Step 3: Update weight $W_c$ in critic NN
    $\dot{\hat{W}}_c \leftarrow -a_1 \frac{\sigma_2}{(\sigma_2^{\mathsf{T}}\sigma_2+1)^2}[\sigma_2^{\mathsf{T}}\hat{W}_c + Q + \hat{u}_o^{\mathsf{T}} R \hat{u}_o]$
Step 4: Update weight $W_a$ in actor NN
    $\dot{\hat{W}}_a \leftarrow -a_2\{F_2\hat{W}_a - F_2\hat{W}_c - \frac{1}{4}\bar{D}_1\hat{W}_a m^{\mathsf{T}}\hat{W}_c\}$
Step 5: Update $u$ in Theorem 2
```

3.4 Stability analysis

The following assumption is necessary for stability analysis in Theorem 2.

Assumption 2 [18] In equation (9), the NN approximate error, NN activation functions and their gradient are bounded on a compact set, i.e., $\|\epsilon\| < b_\epsilon$, $\|\phi\| < b_\phi$, $\|\nabla\epsilon\| < b_{\epsilon_x}$, $\|\nabla\phi\| < b_{\phi_x}$.

Theorem 2 *Given system* (8), *critic NN updating law* (15), *actor NN updating law* (17), *controller u=u_o, there exist a positive integer N_0 such that the number of the hidden layer units $N > N_0$, the closed-loop system states, the critic NN approximate error, and the actor NN approximate error are UUB.*

Proof Choose the Lyapunov function as

$$V = J + \frac{1}{2}a_1^{-1}\tilde{W}_c^{\mathsf{T}}\tilde{W}_c + \frac{1}{2}a_2^{-1}\tilde{W}_a^{\mathsf{T}}\tilde{W}_a + \frac{1}{2}e^{\mathsf{T}}e, \tag{18}$$

where $e = [e_{i1}, e_{i2}]^T$. Taking the derivative, it has

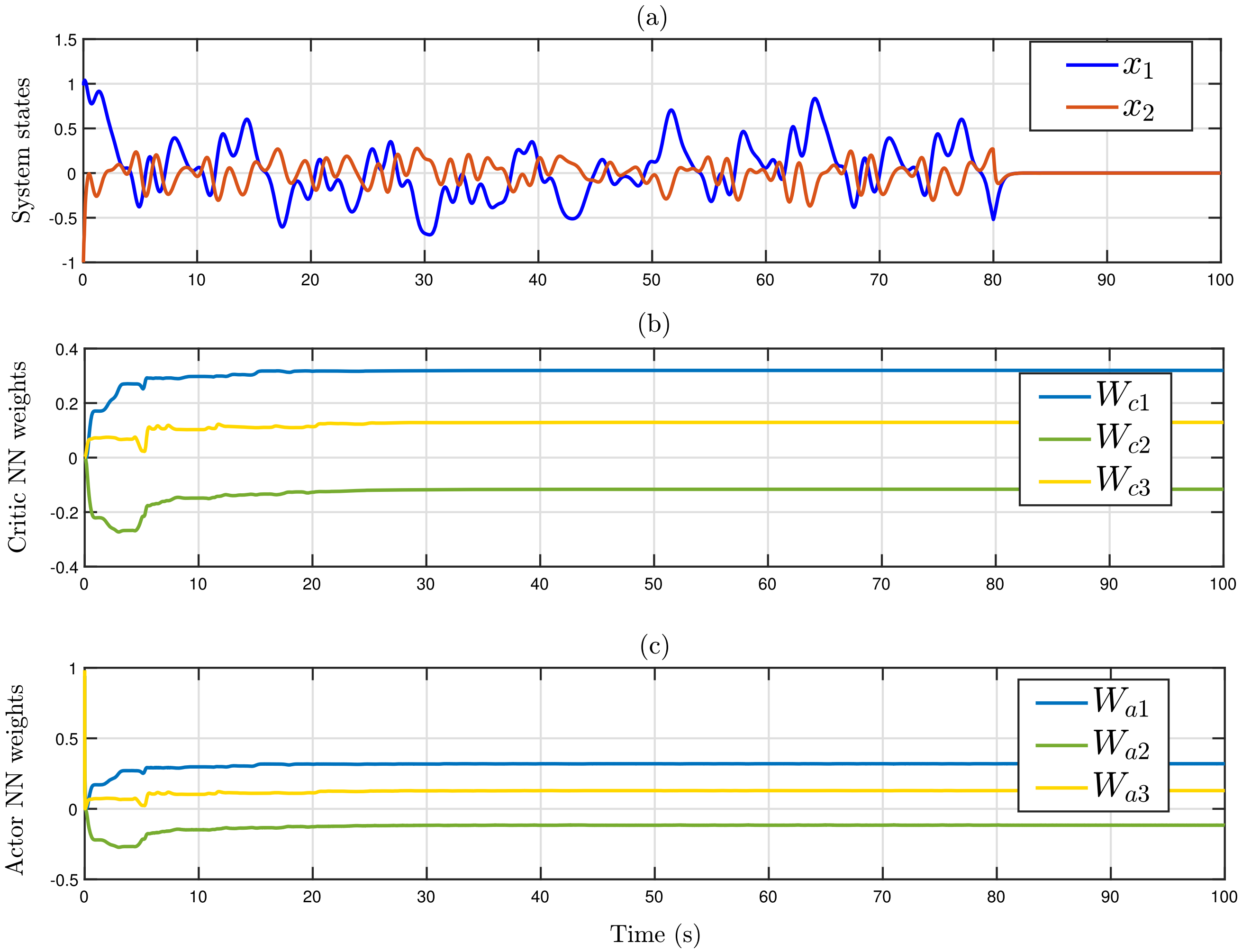

Fig. 2 **a** Trajectories of system states under (16) with reference signal $r = 0$. Before 80 s, states constantly fluctuates due to the presence of the excitation signal. After the excitation signal is removed, system states reach a bound near the equilibrium point $(0,0)^T$. **b** CNN weights. Driven by update law (15), the weights of the CNN eventually converge to within a bounded range of ideal weights. **c** ANN weights. Driven by update law (17), the weights of the ANN eventually converge to within a bounded range of ideal weights

$$\dot{V} = \dot{J} + a_1^{-1}\tilde{W}_c^{\mathsf{T}}\dot{\tilde{W}}_c + a_2^{-1}\tilde{W}_a^{\mathsf{T}}\dot{\tilde{W}}_a + e^{\mathsf{T}}\dot{e}. \tag{19}$$

Firstly, we have

$$\begin{aligned}\dot{J} &= W_c^{\mathsf{T}}\nabla\phi\dot{x} + \nabla\epsilon^{\mathsf{T}}\dot{x}\\ &= W_c^{\mathsf{T}}\nabla\phi(F - \frac{1}{2}GR^{-1}G^T\nabla\phi^{\mathsf{T}}\hat{W}_a)\\ &\quad + \nabla\epsilon^{\mathsf{T}}(F - \frac{1}{2}GR^{-1}G^{\mathsf{T}}\nabla\phi^{\mathsf{T}}\hat{W}_a).\end{aligned} \tag{20}$$

Here we define $\nabla\phi GR^{-1}G^{\mathsf{T}}\nabla\phi^{\mathsf{T}}$ as $\bar{D}_1$, $\nabla\epsilon^{\mathsf{T}}(F - \frac{1}{2}GR^{-1}G^{\mathsf{T}}\nabla\phi^{\mathsf{T}}\hat{W}_a)$ as μ_1 and we have $\dot{J} = W_c^{\mathsf{T}}\sigma_1 + \frac{1}{2}W_c^{\mathsf{T}}\bar{D}_1\tilde{W}_a + \mu_1$. From the HJB Eq. (11), we have $W^T\sigma_1 = -Q - \frac{1}{4}W^{\mathsf{T}}\bar{D}_1W + \upsilon_H$. Then, it has

$$\dot{J} = -Q - \frac{1}{4}W_c^{\mathsf{T}}\bar{D}_1W_c + \frac{1}{2}W_c^{\mathsf{T}}\bar{D}_1\tilde{W}_a + \upsilon_H + \mu_1. \tag{21}$$

In addition, we have

$$\begin{aligned}a_1^{-1}\tilde{W}_c^{\mathsf{T}}\dot{\tilde{W}}_c =& \tilde{W}_c^{\mathsf{T}}\frac{\sigma_2}{(\sigma_2^{\mathsf{T}}\sigma_2+1)^2}[\sigma_2^{\mathsf{T}}\hat{W}_c + Q + \hat{u}_o^{\mathsf{T}}R\hat{u}_o])\\ =& \tilde{W}_c^{\mathsf{T}}\frac{\sigma_2}{(\sigma_2^{\mathsf{T}}\sigma_2+1)^2}[-\sigma_2^{\mathsf{T}}\tilde{W}_c\\ &+ \frac{1}{4}\tilde{W}_a^{\mathsf{T}}\bar{D}_1\tilde{W}_a + \upsilon_H].\end{aligned} \tag{22}$$

Based on the FTDOB, the error becomes 0 after T_d seconds. As $\dot{\hat{W}}_a$ is given in Theorem 2, we have

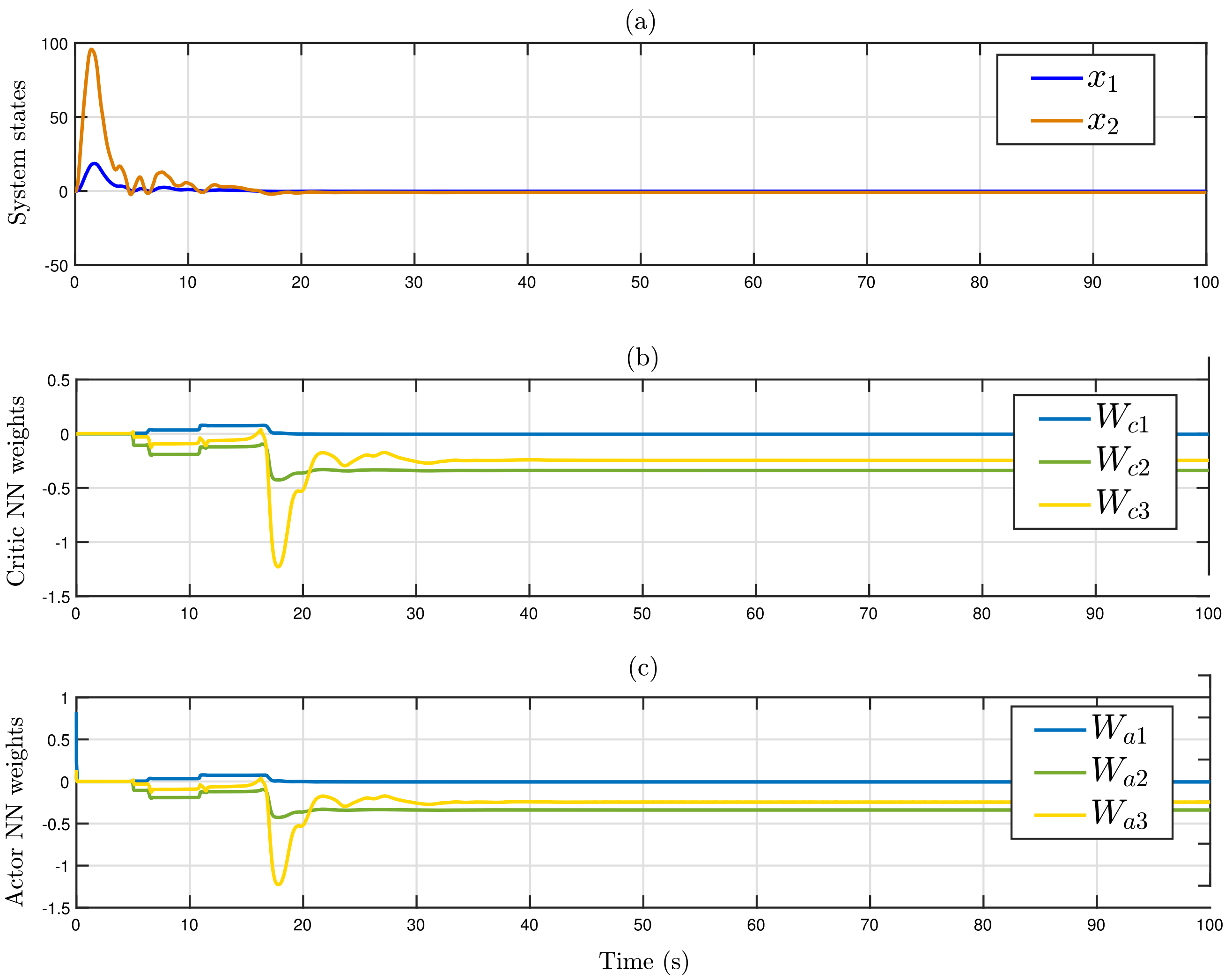

Fig. 3 **a** Trajectories of system states under (16) with reference signal $r = 0$. Before 80 s, states constantly fluctuates due to the presence of the excitation signal. **b** CNN weights. Driven by update law (15), the weights of the CNN eventually converge to within a bounded range of ideal weights. **c** ANN weights. Driven by update law (17), the weights of the ANN eventually converge to within a bounded range of ideal weights

$$\begin{aligned} &\tilde{W}_a^{\mathsf{T}} F_2 \hat{W}_a - \tilde{W}_a^{\mathsf{T}} F_2 \hat{W}_c \\ &= \tilde{W}_a^{\mathsf{T}} F_2 W - \tilde{W}_a^{\mathsf{T}} F_2 \tilde{W}_a - \tilde{W}_a^{\mathsf{T}} F_2 W + \tilde{W}_a^{\mathsf{T}} F_2 \tilde{W}_c. \end{aligned} \tag{23}$$

Then $\dot{V}$ can be obtained as below by adding (6), (15), (17) and (21)

$$\begin{aligned} \dot{V} = &-Q - \frac{1}{4} W_c^{\mathsf{T}} \bar{D}_1 W_c \\ &+ \tilde{W}_c^{\mathsf{T}} \frac{\sigma_2}{(\sigma_2^{\mathsf{T}} \sigma_2 + 1)^2} [-\sigma_2^{\mathsf{T}} \tilde{W}_c + \upsilon_H] + \upsilon_H + \mu_1 \\ &+ \frac{1}{4} \tilde{W}_a^{\mathsf{T}} \bar{D}_1 \hat{W}_a \frac{\bar{\sigma}^{\mathsf{T}}}{m_s} \tilde{W}_c + \frac{1}{2} W^{\mathsf{T}} \bar{D}_1 \tilde{W}_a + \frac{1}{4} \tilde{W}_a^{\mathsf{T}} \bar{D}_1 W \frac{\bar{\sigma}_2^{\mathsf{T}}}{m_s} \tilde{W}_a \\ &- \frac{1}{4} \tilde{W}_a^{\mathsf{T}} \bar{D}_1 W \frac{\bar{\sigma}_2^{T}}{m_s} W + \tilde{W}_a^{\mathsf{T}} F_2 W \\ &- \tilde{W}_a^{T} F_2 \tilde{W}_a - \tilde{W}_a^{\mathsf{T}} F_2 W + \tilde{W}_a^{\mathsf{T}} F_2 \tilde{W}_c, \end{aligned} \tag{24}$$

where $\bar{\sigma} = \frac{\sigma_2}{\sigma_2^{\mathsf{T}} \sigma_2 + 1}$, $m_s = \sigma_2^{\mathsf{T}} \sigma_2 + 1$.

It is obvious that under Assumption 2,

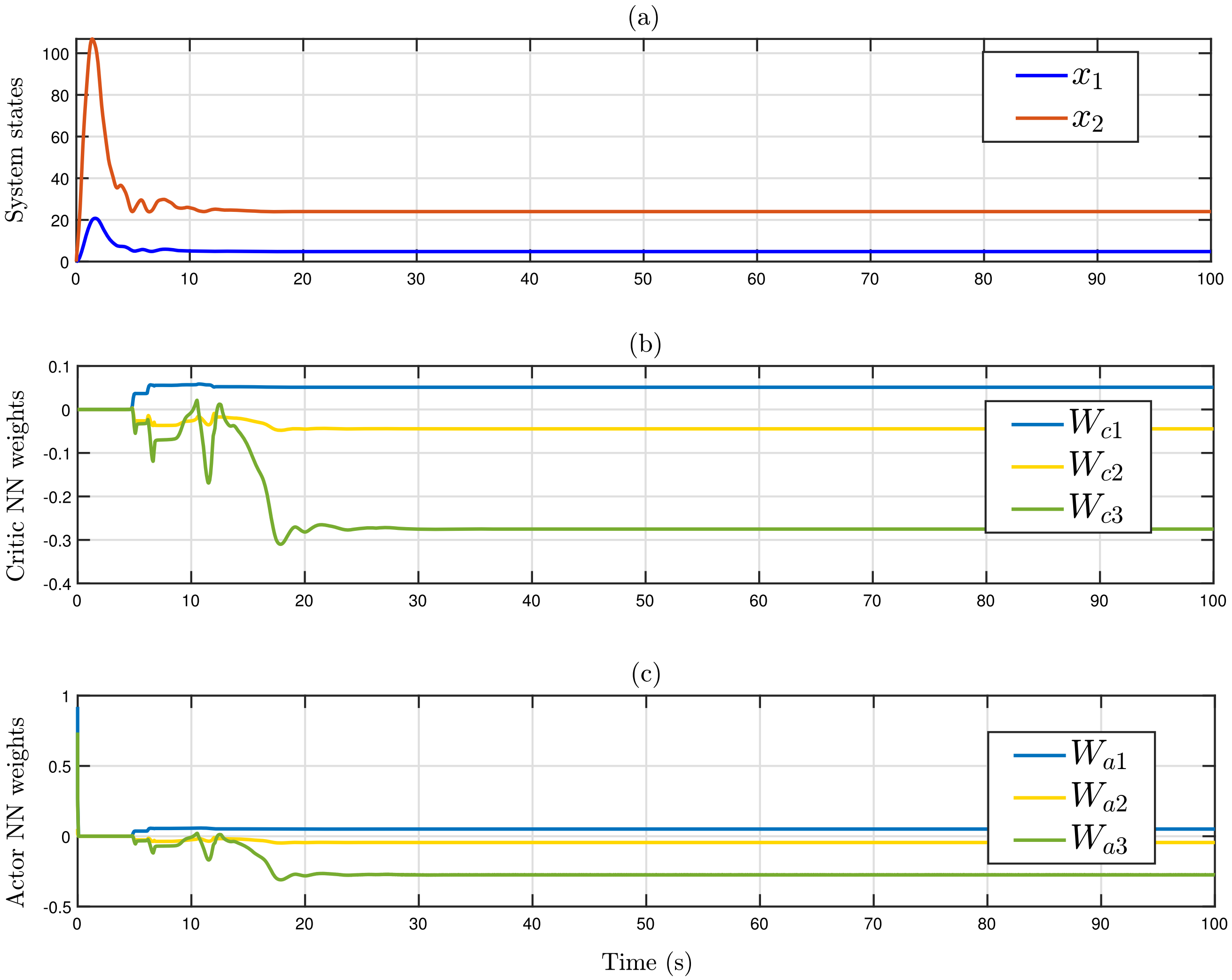

Fig. 4 **a** Trajectories of system states under (16) with reference signal $r = 5$. Before 80 s, states constantly fluctuates due to the presence of the excitation signal. After the excitation signal is removed, state x_1 converges to 5 and x_2 reaches the equilibrium point. **b** CNN weights. Driven by update law (15), the weights of the CNN eventually converge to within a bounded range of ideal weights. **c** ANN weights. Driven by update law (17), the weights of the ANN eventually converge to within a bounded range of ideal weights

$$\mu_1 < b_{\epsilon_x} \|b_f\| \|x\| + \frac{b_{\epsilon_x} b_{\phi_x} b_g^2 \sigma_{\min}(R)(\|W\| + \|\tilde{W}_a\|)}{2}. \tag{25}$$

As given in [18], υ_H converges to 0 as the neurons increase. Hence, N_0 can be selected such that $\sup_{x\in\Omega} \|\upsilon_H\| < \upsilon$. Assuming $N > N_0$, if we define $\tilde{Z} = [Z, \quad \tilde{W}_c, \quad \tilde{W}_a, \quad e]^{\mathsf{T}}$, then we have

$$\dot{V} < -\|\tilde{Z}\|^2 \sigma_{\min}(M) + \|p\| \|\tilde{z}\| + c + \upsilon, \tag{26}$$

where $c = \frac{1}{4}\|W\|^2 \|\bar{D_1}\| + \upsilon + \frac{1}{2}\|W\| b_{\epsilon_x} b_{\phi_x} b_g^2 \sigma_{\min}(R)$,

$$M = \begin{bmatrix} qI & 0 & 0 & 0 \\ 0 & I & (-\frac{F_2}{2} - \frac{\bar{D_1}W}{8m_s})^T & 0 \\ 0 & (-\frac{F_2}{2} - \frac{\bar{D_1}W}{8m_s}) & F_2 - \frac{\bar{D_1}Wm^T + mW^T\bar{D_1}}{8} & 0 \\ 0 & 0 & 0 & \frac{1}{2} \end{bmatrix},$$

$$p = \begin{bmatrix} b_{\varrho_x} b_f \\ \frac{\upsilon}{m_s} \\ (\frac{\bar{D_1} - \frac{\bar{D_1}Wm^T}{4}}{2})W + \frac{b_{\epsilon_x} b_{\phi_x} b_g^2 \sigma_{\min}(R)}{2} \\ -\frac{1}{2}e + f(e) + D \end{bmatrix}. \tag{27}$$

Let the parameters be chosen such that $M > 0$. If $\|\tilde{Z}\| > \sqrt{\frac{p^2}{4\sigma_{\min}(M)} + \frac{c+v}{\sigma_{\min}(M)}} + \frac{\|p\|}{2\sigma_{\min}(M)}$, then, $\dot{V}$ is negative. Thence, the state and the weight error are UUB. □

4 Examples

In this section, a linear system is presented firstly to show that the designed update law guarantees the convergence of the weights to their ideal values. Secondly, a nonlinear system example is employed to highlight the effectiveness of the proposed method.

4.1 Linear system example

Consider a linear system, $\dot{x_1} = -x_1 - 2x_2 + u$, $\dot{x_2} = x_1 - 4x_2 - 3u$, where x_1 and x_2 are system states and u is control input. Choose the cost function as $J = \int_0^\infty (x^T Qx + u^T Ru)\mathrm{d}t$, where $Q = diag(1 \quad 1)$ and $R = 1$.

Clearly, the optimal controller based on linear quadratic regulate theory can be easily found. Hence, the ideal NN wights can be also deduced as $W = [0.3199 \quad -0.1162 \quad 0.1292]$. For this system, the NN-based optimal control is implemented as (16) and the NN tuning law are selected as (15) and (17). In the process of NN convergence, in order to ensure PE condition, we add noise signal $0.5(sin(t)^2 * cos(t) + sin(2t)^2 * cos(0.1t)$ $+sin(-1.2t)^2 * cos(0.5t) + sin(t)^5)$ to the control input here. The reference signal is set as $r = 0$. The simulation results are shown in Fig. 2. The values converge to the optimal values after 50 s, i. e., $\hat{W}_c = [0.3199 \quad -0.1162 \quad 0.1292]$. Also, $\hat{W}_a = [0.3199 \quad -0.1162 \quad 0.1292]$ after 50 s. The optimal controller approximated by NNs is given as

$$\hat{u} = -\frac{R^{-1}}{2}\begin{bmatrix} 1 \\ -3 \end{bmatrix}^T \begin{bmatrix} 2x_1 & 0 \\ x_2 & x_1 \\ 0 & 2x_2 \end{bmatrix}^T \begin{bmatrix} 0.3199 \\ -0.1162 \\ 0.1292 \end{bmatrix}. \tag{28}$$

The excitation signal is introduced to satisfy the PE condition, with the result that sufficiently rich data is generated to train the neural network and ensure its convergence. After 80 s, the neural network has converged. After convergence, the exploration signal is removed, and the value of the state of the system remains near 0 after removal.

4.2 Nonlinear system example

Firstly, we consider a reference signal $r = 0$. In this case, the tracking problem is actually a stabilization problem. The exploration signal is chosen as $200e^{(-0.23t)} * (sin(t)^2 * cos(t) + sin(2t)^2 * cos(0.1t) + sin(-1.2t)^2 * cos(0.5t) + sin(t)^5 + sin(1.12t)^2 + cos(2.4t) * sin(2.4t)^3)$ and the corresponding results are depicted in Fig. 3.

Then, we set $r = 5$ and exploration signal as $exp(-0.35t)$ $*200 * (sin(t)^2 * cos(t) + sin(2t)^2 * cos(0.1t) + sin(-1.2t)^2 * cos(0.5t) + sin(t)^5 + sin(1.12t)^2 + cos(2.4t) * sin(2.4t)^3)$. The results are depicted in Fig. 4.

In our simulation, the sampling time is relatively small at 0.001 s. Therefore, it is reasonable to increase the exponential term in the excitation signal. This approach offers several advantages, including reducing the overall training time and minimizing computational resource wastage. However, in practical systems, hardware limitations often prevent maintaining a very small sampling time. In such cases, as highlighted in [2, 3], it becomes crucial to ensure that the excitation signal does not decay too rapidly. This ensures an ample amount of data is available for training the neural network.

5 Conclusion

This paper focused on the design of robust optimal controllers for high-order nonlinear systems in the presence of mismatched disturbances. The proposed approach involves the design of disturbance observers that ensure fixed-time convergence. Subsequently, the original system is transformed into a filtered error nonlinear system. To address the challenges associated with solving Hamilton–Jacobi–Bellman (HJB) equations, the reinforcement learning method has been introduced. Two neural networks have been designed to approximate the cost function and the optimal control, respectively. By integrating these components, a robust optimal controller is finally obtained. The effectiveness of the proposed method has been validated through two illustrative examples.

Data availability Data sharing not applicable as no new data were generated in this study.

Declarations

Conflict of interest All authors declare that there are no conflicts of interest in this paper.

References

1. Tang L, Gao Y, Liu YJ (2014) Adaptive near optimal neural control for a class of discrete-time chaotic system. Neural Comput Appl 25:1111–1117
2. Na J, Lv Y, Zhang K, Zhao J (2020) Adaptive identifier-critic-based optimal tracking control for nonlinear systems with

experimental validation. IEEE Trans Syst Man Cybern Syst 52(1):459–472

3. Fan ZX, Li S, Liu R (2022) ADP-based optimal control for dystems with mismatched disturbances: a PMSM application. IEEE Trans Circ Syst II Express Briefs 70(6):2057–2061
4. Fan ZX, Adhikary AC, Li S, Liu R (2020) Anti-disturbance inverse optimal control for systems with disturbances. Optim Control Appl Methods 44(3):1321–1340
5. Chen J, Li K, Li K, Yu PS (2021) Dynamic bicycle dispatching of dockless public bicycle-sharing systems using multi-objective reinforcement learning. ACM Trans Cyber-Phys Syst 5(4):1–24
6. Lewis FL, Vrabie DL, Syrmos VL (2012) Optimal control. Wiley
7. Werbos PJ (1992) Approximate dynamic programming for real-time control and neural modeling. handbook of intelligent control neural fuzzy and adaptive approaches, 1992
8. Wei Q, Zhu L, Song R, Zhang P, Liu D, Xiao J (2022) Model-free adaptive optimal control for unknown nonlinear multiplayer nonzero-sum game. IEEE Trans Neural Netw Learn Syst 33(2):879–892
9. Gao W, Jiang ZP (2016) Adaptive dynamic programming and adaptive optimal output regulation of linear systems. IEEE Trans Autom Control 61(12):4164–4169
10. Gao W, Jiang ZP, Lewis FL, Wang Y (2018) Leader-to-formation stability of multiagent systems: an adaptive optimal control approach. IEEE Trans Autom Control 63(10):3581–3587
11. Krstic M, Tsiotras P (1999) Inverse optimal stabilization of a rigid spacecraft. IEEE Trans Autom Control 44(5):1042–1049
12. Fan ZX, Adhikary AC, Li S, Liu R (2022) Disturbance observer based inverse optimal control for a class of nonlinear systems. Neurocomputing 500:821–831
13. Ming X, Balakrishnan SN (2005) A new method for suboptimal control of a class of non-linear systems. Optim Control Appl Methods 26(2):55–83
14. Do TD, Choi HH, Jung WJ (2015) θ-D approximation technique for nonlinear optimal speed control design of surface-mounted PMSM drives. IEEE/ASME Trans Mechatron 20(4):1822–1831
15. Zhang H, Cui L, Zhang X, Luo Y (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
16. Qin C, Zhang H, Luo Y (2014) Optimal tracking control of a class of nonlinear discrete-time switched systems using adaptive dynamic programming. Neural Comput Appl 24:531–538
17. Wang D, Liu D, Zhao D, Huang Y, Zhang D (2013) A neural-network-based iterative GDHP approach for solving a class of nonlinear optimal control problems with control constraints. Neural Comput Appl 22(2):219–227
18. Vamvoudakis KG, Lewis FL (2010) Online actor-critic algorithm to solve the continuous-time infinite horizon optimal control problem. Automatica 46(5):878–888
19. Yang W, Li K, Li K (2019) A pipeline computing method of SpTV for three-order tensors on CPU and GPU. ACM Trans Knowl Discov Data 13(6):1–27
20. Zhong K, Yang Z, Xiao G, Li X, Yang W, Li K (2022) An efficient parallel reinforcement learning approach to cross-layer defense mechanism in industrial control systems. IEEE Trans Parallel Distrib Syst 3(11):2979–2990
21. Liu C, Tang F, Hu Y, Li K, Tang Z, Li K (2021) Distributed task migration optimization in MEC by extending multi-agent deep reinforcement learning approach. IEEE Trans Parallel Distrib Syst 32(7):1603–1614
22. Jiang Y, Jiang ZP (2012) Computational adaptive optimal control for continuous-time linear systems with completely unknown dynamics. Automatica 48(10):2699–2704
23. Bian T, Jiang Y, Jiang ZP (2014) Adaptive dynamic programming and optimal control of nonlinear nonaffine systems. Automatica 50(10):2624–2632
24. Wang D (2020) Robust policy learning control of nonlinear plants with case studies for a power system application. IEEE Trans Industr Inf 16(3):1733–1741
25. Zhao J, Yang C, Gao W, Modares H, Chen X, Dai W (2023) Linear quadratic tracking control of unknown systems: a two-phase reinforcement learning method. Automatica 148:110761
26. Modares H, Lewis FL (2014) Optimal tracking control of nonlinear partially-unknown constrained-input systems using integral reinforcement learning. Automatica 50(7):1780–1792
27. Chen WH (2004) Disturbance observer based control for nonlinear systems. IEEE/ASME Trans Mechatron 9(4):706–710
28. Yu B, Du H, Ding L, Wu D, Li H (2022) Neural network-based robust finite-time attitude stabilization for rigid spacecraft under angular velocity constraint. Neural Comput Appl 34:5107–5117
29. Zhou K, Doyle J, Glover K (1995) Robust and optimal control. Prentice Hall, New Jersey
30. Utkin V (2003) Variable structure systems with sliding modes. IEEE Trans Autom Control 22(2):212–222
31. Levant A (2003) Higher-order sliding modes, differentiation and output-feedback control. Int J Control 76(9–10):924–941
32. Huang J (2004) Nonlinear output regulation- theory and applications. SIAM
33. Ohishi K, Nakao M, Ohnishi K et al (1987) Microprocessor-controlled DC motor for load-insensitive position servo system. IEEE Trans Industr Electron 34(1):44–49
34. Han J (2009) From PID to active disturbance rejection control. IEEE Trans Industr Electron 56(3):900–906
35. Li S, Yang J, Chen WH, Chen X (2014) Disturbance observer-based control: methods and applications. CRC Press, Inc., Boca Raton
36. Li S, Yang J, Chen WH, Chen X (2012) Generalized extended state observer based control for systems with mismatched uncertainties. IEEE Trans Industr Electron 59(12):4792–4802
37. Sun H, Guo L (2017) Neural network-based DOBC for a class of nonlinear systems with unmatched disturbances. IEEE Trans Neural Netw Learn Syst 28(2):482–489
38. Cui B, Zhang L, Xia Y, Zhang J (2022) Continuous distributed fixed-time attitude controller design for multiple spacecraft systems with a directed graph. IEEE Trans Circ Syst II- Express Briefs 69(11):478–4482
39. Li X, Ma L, Mei K, Ding S, Pan T (2023) Fixed-time adaptive fuzzy SOSM controller design with output constraint. Neural Comput Appl 35(13):9893–9905
40. Liu W, Chen M, Shi P (2022) Fixed-time disturbance observer-based control for quadcopter suspension transportation system. IEEE Trans Circ Syst I- Regul Pap 69(11):4632–4642

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.