AIMS Mathematics, 9(10): 27330–27360.
DOI: 10.3934/math.20241328
Received: 21 July 2024
Revised: 23 August 2024
Accepted: 27 August 2024
Published: 20 September 2024

https://www.aimspress.com/journal/Math

Research article

Reinforcement learning-based adaptive tracking control for flexible-joint robotic manipulators

Huihui Zhong[1], Weijian Wen[2,*], Jianjun Fan[1] and Weijun Yang[2]

[1] School of Automation, Guangdong University of Technology, Guangzhou 510006, China
[2] School of Intelligent manufacturing, Guangzhou City Polytechnic, Guangzhou 510405, China

* **Correspondence:** Email: wenweijian@gcp.edu.cn.

Abstract: In this paper, we investigated the optimal tracking control problem of flexible-joint robotic manipulators in order to achieve trajectory tracking, and at the same time reduced the energy consumption of the feedback controller. Technically, optimization strategies were well-integrated into backstepping recursive design so that a series of optimized controllers for each subsystem could be constructed to improve the closed-loop system performance, and, additionally, a reinforcement learning method strategy based on neural network actor-critic architecture was adopted to approximate unknown terms in control design, making that the Hamilton-Jacobi-Bellman equation solvable in the sense of optimal control. With our scheme, the closed-loop stability, the convergence of output tracking error can be proved rigorously. Besides theoretical analysis, the effectiveness of our scheme was also illustrated by simulation results.

Keywords: optimal control; reinforcement learning; neural networks; flexible-joint robotic manipulator; Lyapunov function
Mathematics Subject Classification: 68T40, 93C95, 93D05

1. Introduction

In recent decades, automation has flourished, leading to the widespread integration of robots across various sectors, including industrial production [1], healthcare [2], defense [3], aerospace engineering [4], and numerous other domains [5–7]. Robots used in industrial production are typically made of rigid materials, which results in high manufacturing costs and limited degrees of freedom. Furthermore, because of their relatively rigid structure, they are not well-suited for complex environments and may struggle to efficiently complete tasks in situations that involve interacting with unpredictable environments or objects. Therefore, the control problem of flexible-joint robotic manipulators with high adaptability and an extensive range of degrees of freedom has

received much attention, and various approaches have been developed(e.g., [8–12]), among which the backstepping-based strategy would be the commonly used only due to the advantages in handling nonlinearities [13–19].

The backstepping controller, which utilizes a sampled-data extended state observer (SD-ESO), was proposed in [17] as a methodology to optimize the transient response of a flexible-joint robotic manipulator. This methodology is devised to minimize estimation inaccuracies and other constraints, thereby enhancing the overall performance of the robotic system. In [18], an explicit state feedback controller has been designed to solve the problem of practical tracking control of a flexible-joint robotic manipulator in the presence of actuator saturation by cleverly combining an inverse stepping scheme, an adaptive technique and a method of constructing a command filter and an actuator saturation assist system. In the study presented in [19], an adaptive control scheme is introduced to ensure the convergence of tracking deviations in a flexible-joint robotic manipulator. The methodology employs a backstepping control strategy to ensure that the deviation converges within a specified timeframe to a predetermined range. While the tracking accuracy and convergence rate can be well improved with the existing backstepping-based control schemes such as those mentioned above, they overlook the energy consumption of the controller. Considering that flexible manipulators require more energy for deformation and adjustment compared to rigid manipulators, optimizing energy consumption becomes crucial to enhance system performance and reduce operational costs. Therefore, it is crucial to implement control methods to optimize energy consumption.

Bellman in [20] and Pontryagin in [21] proposed the optimal control. This control approach aims to find control strategies for dynamical systems and to optimize the structured cost metric, thus achieving a harmonious balance between the available resources and required performance. However, since the optimal control is typically determined by solving the Hamilton-Jacobi-Bellman (HJB) equation [22], its inherent nonlinearity and complexity make it challenging to solve directly using analytical methods. Fortunately, the adaptive dynamic programming (ADP) or reinforcement learning (RL) proposed by Werbos et al. [23–25] provides an efficient technique for learning solutions to the HJB equation. The fundamental concept underlying this methodology is to modify the action step-by-step through feedback from the environment. This is generally achieved through the interactive learning of two neural networks (NNs): the actor and the critic. The critic plays a pivotal role in evaluating the actor's actions and providing feedback that guides the actor's policy optimization and subsequent action execution. Therefore, the energy consumption problem of the flexible-joint robotic manipulator can be managed by incorporating optimal control based on RL into the backstepping control. It should be pointed out that, integrating optimized control into the backstepping control of a flexible-joint robotic manipulator remains challenging due to the complexity of system control and convergence analysis.

In this paper, we propose a trajectory tracking control approach for flexible-joint robotic manipulators. By integrating optimization techniques into the backstepping control framework, we formulate each controller as an optimal solution tailored for its respective subsystem. This approach enhances the overall control efficacy of the flexible-joint robotic manipulator system. Concurrently, we employ RL grounded in the NN-based actor-critic architecture to tackle the intricate challenge posed by the HJB equation. In summary, the contributions of this paper are as follows:

(1) By constructing the performance index function with an error term and controller input, the controller is designed to minimize energy consumption and achieve the desired trajectory tracking task of the flexible-joint robotic manipulator.

(2) In the optimal backstepping control of a flexible-joint robotic manipulator, RL based on a NN actor-critic architecture is utilized. In this setup, the critic evaluates performance and provides feedback to the actor, which then executes the actor. This simplifies the design of the controller for the higher-order nonlinear flexible-joint robotic manipulator model.

The rest of this paper is organized as follows. In Section 2, we formulate the control problem, and give some fundamentals for design and analysis. In Section 3, a complete procedure is presented to show how an optimized controller is constructed, and the closed-loop stability is established. In Section 4, simulation results are collected to illustrate the effectiveness of our scheme. The whole paper is concluded in Section 5.

2. Problem statement and preliminaries

2.1. Problem description

Disregarding the viscous damping effects, as referenced in [26], we obtain the dynamic equations for the single-link flexible-joint robotic manipulator depicted in Figure 1.

$$\begin{aligned} I\ddot{q}_1 + Mgl\sin(q_1) + k(q_1 - q_2) &= 0, \\ J\ddot{q}_2 + k(q_2 - q_1) &= u, \end{aligned} \tag{2.1}$$

where q_1 and q_2 are the angular positions of the link and motor shaft, and u is the torque generated by the driving motor. The inertia I and J, the link mass M, the gravity acceleration g, the position of the link's center of gravity l, and the coefficient of strength of the spring k can be obtained by the identification system, so all of them are regarded as known parameters.

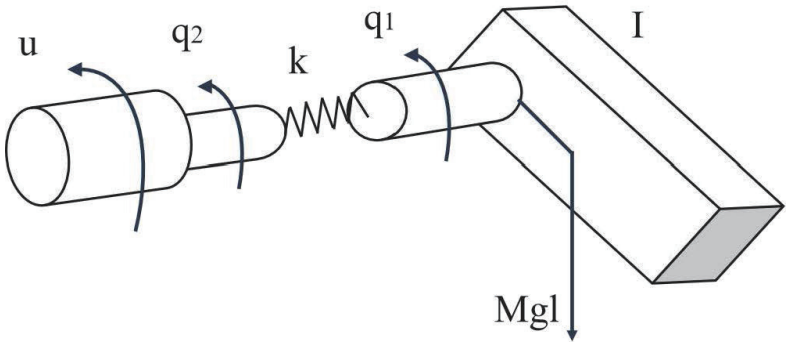

Figure 1. Schematic depiction of the single-link flexible manipulator's structural design.

By selecting the state variables, $x_1 = q_1$, $x_2 = \dot{q}_1$, $x_3 = q_2$, $x_4 = \dot{q}_2$, the dynamic equation of system (2.1) becomes

$$\begin{aligned} \dot{x}_1(t) &= x_2(t), \\ \dot{x}_2(t) &= -\frac{Mgl}{I}\sin(x_1(t)) - \frac{k}{I}(x_1(t) - x_3(t)), \\ \dot{x}_3(t) &= x_4(t), \end{aligned}$$

$$\dot{x}_4(t) = \frac{k}{J}(x_3(t) - x_1(t)) + \frac{1}{J}u(t). \tag{2.2}$$

System (2.2) is equivalent to the following nonlinear model

$$\begin{aligned} \dot{x}_1(t) &= x_2(t), \\ \dot{x}_2(t) &= f_2(\bar{x}_2(t)) + g_2 x_3(t), \\ \dot{x}_3(t) &= x_4(t), \\ \dot{x}_4(t) &= f_4(\bar{x}_4(t)) + g_4 u(t), \\ y(t) &= x_1(t), \end{aligned} \tag{2.3}$$

where $f_2(\bar{x}_2(t)) = -\frac{Mgd}{J_1}\sin(x_1(t)) - \frac{k}{I}x_1(t)$, $g_2 = \frac{k}{I}$, $f_4(\bar{x}_4(t)) = \frac{k}{J}(x_3(t) - x_1(t))$, $g_4 = \frac{1}{J}$. $y(t) \in R$ is the system output, $u(t) \in R$ is the control input, $f(\bar{x}_i(t)) \in R$ is a known and bounded continuous function, and $\dot{x}_i(t), i = 1, \ldots, 4$, are assumed to exhibit stabilizability properties within the subsets that include the origin, and to satisfy the Lipschitz continuous.

Remark 2.1. The assumption that $\dot{x}_i$ satisfies Lipschitz continuous is made here to ensure that the system evolves smoothly over time, preventing sudden changes that could lead to instability or suboptimal performance to facilitate optimal control. Moreover, the system's seamless progression is ensured to remain within a defined boundary, subject to the confinement imposed by the Lipschitz continuity condition. In other words, the velocity of variation exhibited by the system's state variables is confined to a bounded region, dictated by a Lipschitz constant.

Definition 2.1. (Stable and ultimately uniformly bounded (SGUUB) [27]). For a nonlinear system with the state vector $x(t) \in R^n$

$$\dot{x}(t) = f(x, t).$$

Its solution is said to be SGUUB if, for $x(0) \in \Omega_x$ where $\Omega_x \in R^n$ is a compact set, there exist two constants σ and $T(\sigma, x(0))$, such that $\|x(t)\| \leq \sigma$ is held for all $t > t_0 + T(\sigma, x(0))$.

The solution is characterized as SGUUB when, for any initial condition $x(0)$ within the compact subset $\Omega_x \in R^n$, there exist positive scalar constants σ and $T(\sigma, x(0))$ that satisfy the inequality $\|x(t)\| \leq \sigma$ for all time instants t exceeding the initial time t_0 by a duration greater than $T(\sigma, x(0))$.

Lemma 2.1. Given $G(t) \in R$ with $G(0)$ bounded, if $\dot{G}(t) \leq -aG(t) + c$ for $a, c > 0$, then $G(t) \leq e^{-at}G(0) + \frac{c}{a}(1 - e^{-at})$.

Control objectives: In developing a critic-actor RL-based optimal control strategy for the single-link manipulator system (2.3), our objective is to ensure the following:

P1) Within the closed-loop control framework, all error signals, designated as $z_i(t)$ for $i = 1, \cdots, 4$, and the weight estimation errors, expressed as $\tilde{W}_{ci}(t)$ and $\tilde{W}_{ai}(t)$ for $i = 1, \cdots, 4$, are assured to be SGUUB in a predictable and desirable fashion;

P2) The single-link manipulator joint angular position $q_1(t)$ exhibits the capability to follow the desired trajectory y_r in a predictable and desirable manner.

2.2. Basic knowledge of optimal control

To describe the optimal control strategy, consider the following nonlinear continuous-time dynamic system:

$$\dot{x}(t) = f(x) + g(x)u(x), \tag{2.4}$$

where $x(t) \in R^n$ represents the state variable, $f(x) \in R^n$ denotes a continuous function, $u(x) \in R^m$ signifies the input signal, and the term $g(x) \in R^{n\times m}$ is the continuous gain function. Assuming that the derivative $x(t)$ exhibits Lipschitz continuity within the set Ω encompassing the origin, it ensures the uniqueness of the solution for the nonlinear system (2.4) with bounded initial values. Furthermore, the stabilizability of the system (2.4) implies the availability of a continuous control function u that can asymptotically stabilize the system, as referenced in [28].

Define the performance index of the dynamic system (2.4) as follows

$$V(x) = \int_t^\infty r(x(\tau), u(x(\tau)))d\tau,$$

where $r(x, u) = x^T P_1 x + u^T P_2 u$ is the cost function, $P_1 = P_1^T \in R^{n\times n}$ and $P_2 = P_2^T \in R^{m\times m}$ are two positive semi-definite matrices, and P_2 signifies the impact of control efforts on the total cost.

Definition 2.2. The control strategy $u(x)$ is considered acceptable on Ω, denoted as $u(x) \in \Psi(\Omega)$, if $u(x)$ is continuous, $u(0) = 0$, u is stable on Ω, and $V(x)$ is finite.

When addressing the optimization of control strategies related to system (2.4), the primary objective is to determine a suitable control strategy, denoted as $u(x)$ and belonging to the set $\Psi(\Omega)$, that enables the minimization of the value function $V(x)$. Define the HJB function for system (2.4) as follows

$$\begin{aligned} H(x, u, V_x) &= r(x, u) + V_x^T(x)\dot{x}(t) \\ &= x^T P_1 x + u^T P_2 u + V_x^T(x)\big(f(x) + g(x)u(x)\big), \end{aligned}$$

where $V_x(x) = \partial V(x)/\partial x$ is the partial differentiation of the performance index function $V(x)$ with respect to the variable x.

To obtain optimal control, define the optimal function $V^*(x)$ for the dynamic system (2.4) mentioned above with the optimal input $u^*(x)$ as follows:

$$\begin{aligned} V^*(x) &= \min_{u\in\Psi(\Omega)} \left(\int_t^\infty r(x(\tau), u(x(\tau))d\tau \right) \\ &= \int_t^\infty r\big(x(\tau), u^*(x(\tau))\big)d\tau. \end{aligned}$$

The HJB function is then obtained as follows:

$$\begin{aligned} H(x, u^*, V_x^*) &= r(x, u^*) + V_x^{*T}(x)\dot{x}(t) \\ &= x^T P_1 x + u^{*T} P_2 u^* + V_x^{*T}(x)\big(f(x) + g(x)u^*\big) \\ &= 0, \end{aligned} \tag{2.5}$$

where $V_x^*(x) = \partial V^*(x)/\partial x$ denotes the partial derivative of the optimal performance index function $V^*(x)$ with respect to x.

Assuming that (2.5) has, and only has, a unique solution, by solving the equation $\partial H(x, u^*, V_x^*)/\partial u^* = 0$, the expression of $u^*(x)$ is derived as

$$u^*(x) = -\frac{1}{2}P_2^{-1}g^T(x)V_x^*(x). \tag{2.6}$$

Substituting (2.6) into (2.5) gives the following result as

$$\begin{aligned} H(x, u^*, V_x^*) &= x^T P_1 x + V_x^{*T} f(x) - \frac{1}{4}V_x^{*T}(x)g(x)P_2^{-1}g^T(x)V_x^*(x) \\ &= 0. \end{aligned} \tag{2.7}$$

The optimal control policy $u^*(x)$ in (2.5) is unknown because the term $V_x^*(x)$ is unknown, but it can be obtained by solving (2.7) to find the gradient term $V_x^*(x)$, and then substituting $V_x^*(x)$ into (2.6). However solving (2.7) is difficult or even impossible, especially for some high-order systems. To tackle such a problem, the prevalent approach in the extant literature involves employing the technique of RL with an actor-critic architecture: see in [29].

2.3. Neural networks and function approximation

Multiple use cases have formalized the strong function approximation and adaptive learning capabilities of NNs. Distinctly, for any given nonlinear and continuous function $F(z) : R^n \to R^m$ that is defined over a compact domain Ω, NNs of a specific configuration can serve as a proximate representation

$$F_{NN}(z) = W^T\Gamma(z),$$

where $W \in R^{p\times m}$ is the weight of the NN, $\Gamma(z) = [\gamma_1(z), \gamma_2(z), \ldots, \gamma_p(z)]^T \in R^p$ represents the Gaussian basis function vector, and p signifies the total number of neurons. Specifically, the expression for γ_i where $i = 1, \ldots, p$ is given as follows:

$$\gamma_i = exp[\frac{-(x - v_i)^T(x - v_i)}{\varphi_i^2}],$$

where $v_i = [v_{i1}, v_{i2}, \ldots, v_{in}]$ are centers of the respective field, and φ_i is the width of the Gaussian function.

In accordance with theoretical principles, there ought to exist an optimal weight matrix, denoted as W^*, which enables the accurate representation of $F(z)$ as follows

$$F(z) = W^{*T}\Gamma(z) + \varepsilon(z),$$

where $\varepsilon(z) \in R^m$ denotes the approximation error that when the number of neurons p is large enough to satisfy $\|\varepsilon(z)\| \le \delta$, δ is an extremely small positive constant, and W^* is the ideal weight used only for making stability analysis, denoted as

$$W^* \triangleq \arg\min_{W\in R^{p\times m}} \left\{ \sup_{z\in\Omega_z} \left\| F(z) - W^T\Gamma(z) \right\| \right\}.$$

3. Optimized backstepping design and stability analysis

Step 1: In this step, the tracking deviation vector is defined as $z_1(t) = x_1(t) - y_r(t)$. From (2.3), it can be deduced that its derivative is

$$\dot{z}_1(t) = x_2(t) - \dot{y}_r(t). \tag{3.1}$$

The optimal virtual control for the first step is denoted by $\alpha_1^*(z_1)$, with the optimal value function being defined accordingly,

$$\begin{aligned} V_1^*(z_1) &= \min_{\alpha_1 \in \Psi(\Omega_{z_1})} \left(\int_t^\infty r_1(z_1(\tau), \alpha_1(z_1(\tau)))d\tau \right) \\ &= \int_t^\infty r_1(z_1(\tau), \alpha_1^*(z_1(\tau)))d\tau, \end{aligned} \tag{3.2}$$

where $\alpha_1(z_1)$ is the virtual control, Ω_{z_1} is the admissible set of α_1^*, and $r_1 = z_1^2(t) + \alpha_1^2(z_1)$ is the cost function in the first step. The optimal performance index function $V_1^*(z_1)$ is divided into two components as shown below to facilitate the construction of optimal tracking control,

$$V_1^*(z_1) = \beta_1 z_1^2(t) + V_1^o(z_1), \tag{3.3}$$

where $\beta_1 > 0$ is a designable constant, and $V_1^o(z_1) = -\beta_1 z_1^2(t) + V_1^*(z_1)$. By viewing $x_2(t)$ as α_1^*, the HJB function can be obtained from tracking error (3.1) and the optimal function (3.3) as follows

$$\begin{aligned} H_1\left(z_1, \alpha_1^*, \frac{\partial V_1^*}{\partial z_1}\right) &= r_1 + \frac{\partial V_1^*(z_1)}{\partial z_1}\dot{z}_1(t) \\ &= z_1^2(t) + \alpha_1^{*2}(z_1) + \left(2\beta_1 z_1(t) + \frac{\partial V_1^o(z_1)}{\partial z_1}\right)(\alpha_1^*(z_1) - \dot{y}_r(t)) \\ &= 0. \end{aligned} \tag{3.4}$$

The optimal virtual control α_1^* can be derived by solving $\partial H_1/\partial \alpha_1^* = 0$ as

$$\alpha_1^*(z_1) = -\beta_1 z_1(t) - \frac{1}{2}\frac{\partial V_1^o(z_1)}{\partial z_1}. \tag{3.5}$$

Because solving $\partial V_1^o(z_1)/\partial z_1$ is complex, but the term is continuous for Ω_{z_1}, it can be approximated with an NN as

$$\frac{\partial V_1^o(z_1)}{\partial z_1} = W_1^{*T}\Gamma_1(z_1) + \varepsilon_1(z_1), \tag{3.6}$$

where $W_1^{*T} \in R^{m_1}$ represents the ideal weight in the NN, and the item $\Gamma_1(z_1) \in R^{m_1}$ signifies the basis function in the NN, and $\varepsilon_1(z_1) \in R$ is the bounded approximation error.

Remark 3.1. Note that both NNs and FLSs can be used to approximate uncertain functions: see [30–32] for examples. Nevertheless, compared with FLS, the NN approximator could have the following advantages: 1) NNs eliminate the need to formulate a rule base, as they can automatically

learn the input-output mapping relationship through training, making the process less complex, and 2) NNs can effectively handle anomalous samples through an adaptive mechanism.

With the aid of (3.6), it can be derived from (3.3) and (3.5) that

$$\frac{\partial V_1^*(z_1)}{\partial z_1} = 2\beta_1 z_1(t) + W_1^{*T}\Gamma_1(z_1) + \varepsilon_1(z_1), \tag{3.7}$$

$$\alpha_1^*(z_1) = -\beta_1 z_1(t) - \frac{1}{2}(W_1^{*T}\Gamma_1(z_1) + \varepsilon_1(z_1)). \tag{3.8}$$

Substituting (3.6) and (3.8) into (3.4), we can get the following expression:

$$\begin{aligned} H_1(z_1, \alpha_1^*, W_1^*) = & -(\beta_1^2 - 1)z_1^2(t) - 2\beta_1 \dot{y}_r(t) z_1(t) + W_1^{*T}\Gamma_1(z_1)(-\dot{y}_r(t) - \beta_1 z_1(t)) \\ & -\frac{1}{4}W_1^{*T}\Gamma_1(z_1)\Gamma_1^T(z_1)W_1^* + \epsilon_1(t) = 0, \end{aligned} \tag{3.9}$$

where $\epsilon_1(t) = \varepsilon_1(z_1)(-\dot{y}_r(t) + \alpha_1^*) + (1/4)\varepsilon_1^2(z_1)$ is bounded.

Due to the uncertainty surrounding the ideal weight W_1^*, the optimal virtual control in (3.8) remains undetermined. Therefore, to achieve the desired tracking control, we employ an RL algorithm based on an actor-critic framework. In this framework, we use the critic module to assess the effectiveness of the control, while the actor component formulates the virtual control signal

$$\frac{\partial \hat{V}_1^*(z_1)}{\partial z_1} = 2\beta_1 z_1(t) + \hat{W}_{c1}^T(t)\Gamma_1(z_1), \tag{3.10}$$

$$\hat{\alpha}_1(z_1) = -\beta_1 z_1(t) - \frac{1}{2}\hat{W}_{a1}^T(t)\Gamma_1(z_1), \tag{3.11}$$

where $\hat{V}_1^*$ is the estimation of V_1^*, $\hat{W}_{c1} \in R^{m1}$ represents the weight of critic NN, and $\hat{W}_{a1} \in R^{m1}$ is the actor NN weight.

Remark 3.2. It's worth noting that unlike the single NN approach for approximating unknown functions discussed in [31] and other works, this paper employs RL based on actor-critic NNs. In this framework, the critic evaluates performance and provides feedback to the participants, who then execute the suggested actions. Since the critic offers direct feedback on the policy, the actor can focus on optimizing the policy, resulting in a more stable and effective update. In contrast, a single NN typically updates its strategy based on direct returns to adjust the policy, which can result in greater variance and negatively impact the efficiency and stability of the learning process.

By incorporating Eqs (3.10) and (3.11) into the framework of (3.4), the HJB equation is derived as

$$\begin{aligned} H_1\left(z_1, \hat{\alpha}_1, \hat{W}_{c1}\right) = & \, z_1^2(t) + \left(-\beta_1 z_1(t) - \frac{1}{2}\hat{W}_{a1}^T(t)\Gamma_1(z_1)\right)^2 \\ & + (2\beta_1 z_1(t) + \hat{W}_{c1}^T(t)\Gamma_1(z_1))\left(-\beta_1 z_1(t) - \frac{1}{2}\hat{W}_{a1}^T(t)\Gamma_1(z_1) - \dot{y}_r(t)\right). \end{aligned} \tag{3.12}$$

Bellman residual error $e_1(t)$ can be derived from (3.9) and (3.12) as

$$\begin{aligned} e_1(t) &= H_1(z_1, \hat{\alpha}_1, \hat{W}_{c1}) - H_1(z_1, \alpha_1^*, W_1^*) \\ &= H_1(z_1, \hat{\alpha}_1, \hat{W}_{c1}). \end{aligned} \tag{3.13}$$

Define the positive definite function of the Bellman residual error (3.13) as

$$E_1(t) = \frac{1}{2}e_1^2(t). \tag{3.14}$$

To achieve the minimization of $E_1(t)$, the update law for the critic NN is derived by employing the method of gradient descent,

$$\begin{aligned}\dot{\hat{W}}_{c1}(t) =& -\frac{\mu_{c1}}{\|\omega_1\|^2+1}\frac{\partial E_1(t)}{\partial \hat{W}_{c1}} \\ =& -\frac{\mu_{c1}}{\|\omega_1\|^2+1}\omega_1(t)\Big(\omega_1^T(t)\hat{W}_{c1}(t) - (\beta_1^2-1)z_1^2(t) + 2\beta_1 z_1(-\dot{y}_r) \\ &+\frac{1}{4}\hat{W}_{a1}^T\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}\Big),\end{aligned} \tag{3.15}$$

where $\mu_{c1} > 0$ is the learning rate of critic NN and $\omega_1 = \Gamma_1(z_1)(-\beta_1 z_1(t) - (1/2)\hat{W}_{a1}^T\Gamma_1(z_1) - \dot{y}_r) \in R^{m_1}$.

Remark 3.3. The matrix $\omega_i(t)$ needs to satisfy the following equation for every t within the interval $[t, t+\bar{t}_i]$:

$$\Lambda_i I_{m_i} \le \omega_i(t)\omega_i^T(t) \le \eta_i I_{m_i}, i = 1, \cdots, 4, \tag{3.16}$$

where Λ_i, η_i, and $\bar{t}_i$ are all positive values, and $I_{m_i} \in R^{m_i \times m_i}$ is the identity matrix. Satisfying the aforementioned incentive persistence conditions enhances the robustness and adaptability of the system, which further ensures the stability and performance of the flexible-joint robotic manipulator system.

The actor NN weight is updated by the following law

$$\begin{aligned}\dot{\hat{W}}_{a1}(t) =& \frac{1}{2}\Gamma_1(z_1)z_1(t) - \mu_{a1}\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t) \\ &+\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\omega_1^T(t)\hat{W}_{c1}(t),\end{aligned} \tag{3.17}$$

where $\mu_{a1} > 0$ is the actor learning rate.

Designate the tracking discrepancy for the second step as $z_2(t) = x_2(t) - \hat{\alpha}_1(z_1)$. Replace $x_2(t)$ with $z_2(t) + \hat{\alpha}_1(z_1)$, then we can yield (3.1) as follows:

$$\dot{z}_1(t) = z_2(t) + \hat{\alpha}_1(z_1) - \dot{y}_r(t). \tag{3.18}$$

Taking into account the scalar quadratic Lyapunov function pertaining to the first step, its formulation is presented as follows:

$$L_1(t) = \frac{1}{2}z_1^2(t) + \frac{1}{2}\tilde{W}_{c1}^T(t)\tilde{W}_{c1}(t) + \frac{1}{2}\tilde{W}_{a1}^T(t)\tilde{W}_{a1}(t), \tag{3.19}$$

where $\tilde{W}_{c1}(t) = \hat{W}_{c1}(t) - W_1^*$ is the critic NN weight error, and $\tilde{W}_{a1}(t) = \hat{W}_{a1}(t) - W_1^*$ is the NN weight error of the actor. The derivative of (3.19) is

$$\dot{L}_1(t) = z_1(t)\dot{z}_1(t) + \tilde{W}_{c1}^T(t)\dot{\hat{W}}_{c1}(t) + \tilde{W}_{a1}^T(t)\dot{\hat{W}}_{a1}(t). \tag{3.20}$$

Then, recalling the tracking error (3.18), the updating law (3.15) (3.17), and the virtual control (3.11), we have

$$\begin{aligned}\dot{L}_1(t) = & z_1(t)(z_2(t) + \hat{a}_1(z_1) - \dot{y}_r(t)) \\ & -\frac{\mu_{c1}}{\|\omega_1\|^2+1}\tilde{W}_{c1}^T(t)\omega_1\Big(\omega_1^T\hat{W}_{c1}(t) - (\beta_1^2-1)z_1^2(t) - 2\beta_1 z_1(t)\dot{y}_r(t) \\ & +\frac{1}{4}\hat{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\Big) \\ & +\tilde{W}_{a1}^T(t)\Big(\frac{1}{2}\Gamma_1(z_1)z_1(t) - \mu_{a1}\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t) \\ & +\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\omega_1^T(t)\hat{W}_{c1}(t)\Big). \end{aligned} \tag{3.21}$$

By collating Eq (3.21), the following expression can be obtained:

$$\begin{aligned}\dot{L}_1(t) = & z_1(t)z_2(t) - \beta_1 z_1^2(t) - z_1(t)\dot{y}_r - \frac{1}{2}z_1(t)\hat{W}_{a1}^T(t)\Gamma_1(z_1) \\ & +\frac{1}{2}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)z_1(t) - \mu_{a1}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t) \\ & +\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\omega_1^T\hat{W}_{c1}(t) \\ & -\frac{\mu_{c1}}{\|\omega_1\|^2+1}\tilde{W}_{c1}^T(t)\omega_1\Big(\omega_1^T\hat{W}_{c1}(t) - (\beta_1^2-1)z_1^2(t) + 2\beta_1 z_1(t)(-\dot{y}_r) \\ & +\frac{1}{4}\hat{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\times\hat{W}_{a1}(t)\Big). \end{aligned} \tag{3.22}$$

The following results can be deduced because of the equation $\tilde{W}_{a1}(t) = \hat{W}_{a1}(t) - W_1^*$:

$$\tilde{W}_{a1}^T(t)\Gamma_1(z_1)z_1 - z_1\hat{W}_{a1}^T(t)\Gamma_1(z_1) = -z_1(t)W_1^{*T}\Gamma_1(z_1), \tag{3.23}$$

$$\begin{aligned}\mu_{a1}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t) = & \frac{\mu_{a1}}{2}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\tilde{W}_{a1}(t) \\ & +\frac{\mu_{a1}}{2}\hat{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t) \\ & -\frac{\mu_{a1}}{2}W_1^{*T}\Gamma_1(z_1)\Gamma_1^T(z_1)W_1^*. \end{aligned} \tag{3.24}$$

By inserting (3.23) and (3.24) into (3.22), $\dot{L}_1(t)$ is rewritten as

$$\begin{aligned}\dot{L}_1(t) = & z_1(t)z_2(t) - \beta_1 z_1^2(t) - z_1(t)\dot{y}_r - \frac{1}{2}z_1(t)W_1^{*T}\Gamma_1(z_1) \\ & -\frac{\mu_{a1}}{2}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\tilde{W}_{a1}(t) - \frac{\mu_{a1}}{2}\hat{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t) \\ & +\frac{\mu_{a1}}{2}W_1^{*T}\Gamma_1(z_1)\Gamma_1^T(z_1)W_1^* + \frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\omega_1^T\hat{W}_{c1}(t) \\ & -\frac{\mu_{c1}}{\|\omega_1\|^2+1}\tilde{W}_{c1}^T(t)\omega_1\Big(\omega_1^T\hat{W}_{c1}(t) - (\beta_1^2-1)z_1^2(t) + 2\beta_1 z_1(t)(-\dot{y}_r)\end{aligned}$$

$$+\frac{1}{4}\hat{W}_{a1}^{T}\Gamma_1(z_1)\Gamma_1^{T}(z_1)\hat{W}_{a1}\Big). \tag{3.25}$$

Utilizing Young's inequality $ab \le (a^2/2) + (b^2/2)$, the following results are derived

$$-z_1(t)\dot{y}_r(t) \le \frac{1}{2}z_1^2(t) + \frac{1}{2}\dot{y}_r^2(t), \tag{3.26}$$

$$z_1(t)z_2(t) \le z_1^2(t) + z_2^2(t), \tag{3.27}$$

$$-\frac{1}{2}z_1(t)W_1^{*T}\Gamma_1(z_1) \le \frac{1}{2}z_1^2(t) + \frac{1}{2}(W_1^{*T}\Gamma_1(z_1))^2. \tag{3.28}$$

By substituting (3.26), (3.27), and (3.28) into (3.25), we can get the following derivation:

$$\begin{aligned}\dot{L}_1(t) \le\ & z_2^2(t) - (\beta_1 - 2)z_1^2(t) + \frac{1}{2}\dot{y}_r^2 + \frac{\mu_{a1}+1}{2}(W_1^{*T}\Gamma_1(z_1))^2 \\ & -\frac{\mu_{a1}}{2}\tilde{W}_{a1}^{T}(t)\Gamma_1(z_1)\Gamma_1^{T}(z_1)\tilde{W}_{a1}(t) - \frac{\mu_{a1}}{2}\hat{W}_{a1}^{T}(t)\Gamma_1(z_1)\Gamma_1^{T}(z_1)\hat{W}_{a1}(t) \\ & +\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{a1}^{T}(t)\Gamma_1(z_1)\Gamma_1^{T}(z_1)\hat{W}_{a1}(t)\omega_1^{T}\hat{W}_{c1}(t) \\ & -\frac{\mu_{c1}}{\|\omega_1\|^2+1}\tilde{W}_{c1}^{T}(t)\omega_1\Big(\omega_1^{T}\hat{W}_{c1}(t) - (\beta_1^2-1)z_1^2(t) + 2\beta_1 z_1(-\dot{y}_r) \\ & +\frac{1}{4}\hat{W}_{a1}^{T}(t)\Gamma_1(z_1)\Gamma_1^{T}(z_1)\hat{W}_{a1}(t)\Big).\end{aligned} \tag{3.29}$$

There is the following fact:

$$\begin{aligned}& -(\beta_1^2-1)z_1^2 + 2\beta_1 z_1(-\dot{y}_r) \\ &= -W_1^{*T}\Gamma_1(z_1)(-\dot{y}_r(t) - \beta_1 z_1(t)) + \frac{1}{4}W_1^{*T}\Gamma_1(z_1)\Gamma_1^{T}(z_1)W_1^{*} - \epsilon_1(t) \\ &= -\omega_1^{T}W_1^{*} - \frac{1}{2}\hat{W}_{a1}^{T}(t)\Gamma_1(z_1)\Gamma_1^{T}(z_1)W_1^{*} + \frac{1}{4}W_1^{*T}\Gamma_1(z_1)\Gamma_1^{T}(z_1)W_1^{*} - \epsilon_1(t),\end{aligned} \tag{3.30}$$

then we can rewrite the inequality (3.29) as

$$\begin{aligned}\dot{L}_1(t) \le\ & z_2^2(t) - (\beta_1 - 2)z_1^2(t) + \frac{\mu_{a1}+1}{2}(W_1^{*T}\Gamma_1(z_1))^2 + \frac{1}{2}\dot{y}_r^2 \\ & -\frac{\mu_{a1}}{2}\tilde{W}_{a1}^{T}(t)\Gamma_1(z_1)\Gamma_1^{T}(z_1)\tilde{W}_{a1}(t) - \frac{\mu_{a1}}{2}\hat{W}_{a1}^{T}(t)\Gamma_1(z_1)\Gamma_1^{T}(z_1)\hat{W}_{a1}(t) \\ & +\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{a1}^{T}(t)\Gamma_1(z_1)\Gamma_1^{T}(z_1)\hat{W}_{a1}(t)\omega_1^{T}\hat{W}_{c1}(t) \\ & -\frac{\mu_{c1}}{\|\omega_1\|^2+1}\tilde{W}_{c1}^{T}(t)\omega_1\Big(\omega_1^{T}(t)\tilde{W}_{c1}(t) - \frac{1}{2}\hat{W}_{a1}^{T}(t)\Gamma_1(z_1)\Gamma_1^{T}(z_1)W_1^{*} \\ & +\frac{1}{4}W_1^{*T}\Gamma_1(z_1)\Gamma_1^{T}(z_1)W_1^{*} \\ & +\frac{1}{4}\hat{W}_{a1}^{T}(t)\Gamma_1(z_1)\Gamma_1^{T}(z_1)\hat{W}_{a1}(t) - \epsilon_1(t)\Big).\end{aligned} \tag{3.31}$$

Given the equation $\tilde{W}_{a1}(t) = \hat{W}_{a1}(t) - W_1^*$, it leads to the following equations:

$$\begin{aligned}
&-\frac{1}{2}\hat{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)W_1^* + \frac{1}{4}W_1^{*T}\Gamma_1(z_1)\Gamma_1^T(z_1)W_1^* + \frac{1}{4}\hat{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\\
&= \frac{1}{4}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t) - \frac{1}{4}W_1^{*T}\Gamma_1(z_1)\Gamma_1^T(z_1)\tilde{W}_{a1}(t). \qquad (3.32)
\end{aligned}$$

Pursuant to Young's inequality, the subsequent consequence can be deduced

$$\frac{\mu_{c1}}{\|\omega_1\|^2+1}\tilde{W}_{c1}^T(t)\omega_1(t)\epsilon_1(t) \le \frac{\mu_{c1}}{2(\|\omega_1\|^2+1)}\tilde{W}_{c1}^T(t)\omega_1(t)\omega_1^T(t)\tilde{W}_{c1}(t) + \frac{\mu_{c1}}{2}\epsilon_1^2(t). \qquad (3.33)$$

Adding (3.32) and (3.33) into (3.31) yields

$$\begin{aligned}
\dot{L}_1(t) \le\ & z_2^2(t) - (\beta_1 - 2)z_1^2(t) + +\frac{\mu_{a1}+1}{2}(W_1^{*T}\Gamma_1(z_1))^2 + \frac{1}{2}\dot{y}_r^2\\
&-\frac{\mu_{a1}}{2}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\tilde{W}_{a1}(t) - \frac{\mu_{a1}}{2}\hat{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\\
&-\frac{\mu_{c1}}{2(\|\omega_1\|^2+1)}\tilde{W}_{c1}^T(t)\omega_1\omega_1^T\tilde{W}_{c1}(t) + \frac{\mu_{c1}}{2}\epsilon_1^2(t)\\
&+\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\omega_1^T\hat{W}_{c1}(t)\\
&-\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{c1}^T(t)\omega_1\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\\
&+\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{c1}^T(t)\omega_1 W_1^{*T}\Gamma_1(z_1)\Gamma_1^T(z_1)\tilde{W}_{a1}. \qquad (3.34)
\end{aligned}$$

Substituting the following equation

$$\begin{aligned}
&\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\omega_1^T\hat{W}_{c1}(t)\\
&-\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{c1}^T(t)\omega_1\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\\
&= \frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)W_1^{*T}\omega_1\Gamma_1^T(z_1)\hat{W}_{a1}(t), \qquad (3.35)
\end{aligned}$$

into (3.34), we have

$$\begin{aligned}
\dot{L}_1(t) \le\ & z_2^2(t) - (\beta_1 - 2)z_1^2(t) + \frac{\mu_{a1}+1}{2}(W_1^{*T}\Gamma_1(z_1))^2 + \frac{1}{2}\dot{y}_r^2\\
&-\frac{\mu_{a1}}{2}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\tilde{W}_{a1}(t) - \frac{\mu_{a1}}{2}\hat{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\\
&-\frac{\mu_{c1}}{2(\|\omega_1\|^2+1)}\tilde{W}_{c1}^T(t)\omega_1\omega_1^T\tilde{W}_{c1}(t) + \frac{\mu_{c1}}{2}\epsilon_1^2(t)\\
&+\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)W_1^{*T}\omega_1\Gamma_1^T(z_1)\hat{W}_{a1}(t)\\
&+\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{c1}^T(t)\omega_1 W_1^{*T}\Gamma_1(z_1)\Gamma_1^T(z_1)\tilde{W}_{a1}(t). \qquad (3.36)
\end{aligned}$$

Employing the principles of Young's inequality in conjunction with Cauchy's inequality, a series of inequalities can be formulated as follows:

$$\begin{aligned}
&\frac{\mu_{c1}}{4(\|\omega_1\|^2+1)}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)W_1^{*T}\omega_1(t)\Gamma_1^T(z_1)\hat{W}_{a1}(t)\\
\leq\ &\frac{1}{32}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)W_1^{*T}\omega_1\omega_1^TW_1^*\Gamma_1^T(z_1)\tilde{W}_{a1}(t)\\
&+\frac{\mu_{c1}^2}{2}\hat{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t), \qquad (3.37)\\
&\frac{\mu_{c1}^2}{4(\|\omega_1\|^2+1)}\tilde{W}_{c1}^T(t)\omega_1(t)W_1^{*T}\Gamma_1(z_1)\Gamma_1^T(z_1)\tilde{W}_{a1}(t)\\
\leq\ &\frac{1}{32(\|\omega_1\|^2+1)}\tilde{W}_{c1}^T(t)\Gamma_1(z_1)W_1^{*T}\omega_1\omega_1^TW_1^*\Gamma_1^T(z_1)\tilde{W}_{c1}(t)\\
&+\frac{\mu_{c1}^2}{2}\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\tilde{W}_{a1}(t). \qquad (3.38)
\end{aligned}$$

By incorporating the aforementioned inequalities into (3.36), we have made the necessary substitution,

$$\begin{aligned}
\dot{L}_1(t)\ \leq\ & z_2^2(t)-(\beta_1-2)z_1^2(t)\\
&-\Big(\frac{\mu_{a1}}{2}-\frac{\mu_{c1}^2}{2}-\frac{1}{32}W_1^{*T}\omega_1\omega_1^TW_1^*\Big)\tilde{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\tilde{W}_{a1}(t)\\
&-\frac{1}{\|\omega_1\|^2+1}\Big(\frac{\mu_{c1}}{2}-\frac{1}{32}W_1^{*T}\Gamma_1(z_1)\Gamma_1^T(z_1)W_1^*\Big)\tilde{W}_{c1}^T(t)\omega_1\omega_1^T\tilde{W}_{c1}(t)\\
&-\Big(\frac{\mu_{a1}}{2}-\frac{\mu_{c1}^2}{2}\Big)\hat{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t)+\frac{1}{2}\dot{y}_r^2(t)+\frac{\mu_{a1}+1}{2}\left(W_1^{*T}\Gamma_1(z_1)\right)^2+\frac{\mu_{c1}}{2}\epsilon_1^2(t). \qquad (3.39)
\end{aligned}$$

Rewrite (3.39) as follows:

$$\begin{aligned}
\dot{L}_1(t)\ \leq\ & -\xi_1^T(t)A_1(t)\xi_1(t)+C_1(t)+z_2^2(t)\\
&-\Big(\frac{\mu_{a1}}{2}-\frac{\mu_{c1}^2}{2}\Big)\hat{W}_{a1}^T(t)\Gamma_1(z_1)\Gamma_1^T(z_1)\hat{W}_{a1}(t), \qquad (3.40)
\end{aligned}$$

where $\xi_1(t)=[z_1(t),\tilde{W}_{a1}^T(t),\tilde{W}_{c1}^T(t)]^T$, $C_1(t)=\frac{1}{2}\dot{y}_r^2(t)+\frac{\mu_{a1}+1}{2}(W_1^{*T}\Gamma_1(z_1))^2+\frac{\mu_{c1}}{2}\epsilon_1^2(t)$.

In accordance with the persistence of excitation (PE) assumption, the positivity of the definite matrix $A_1(t)$ can be ensured through the deliberate design of the parameters β_1, μ_{c1} and μ_{a1} in such a way that they satisfy the specified set of inequalities

$$\beta_1>2,\quad \mu_{c1}>\frac{1}{16}\lambda_1,\quad \mu_{a1}>\mu_{c1}^2+\frac{\eta_1}{16}W_1^{*T}W_1^*, \qquad (3.41)$$

where λ_1 is the maximal eigenvalue of $\Lambda_1=W_1^{*T}\Gamma_1(z_1)\Gamma_1^T(z_1)W_1^*$. Then, (3.40) becomes

$$\dot{L}_1(t)<z_2^2(t)-a_1\|\xi_1(t)\|^2+c_1, \qquad (3.42)$$

where a_1 is the lower bound on the minimum eigenvalue of $A_1(t)$ and c_1 is the maximum value of $C_1(t)$.

Step 2: According to the tracking error $z_2(t) = x_2(t) - \hat{\alpha}_1(z_1)$ in the second step, it yields that

$$\dot{z}_2(t) = f_2(\bar{x}_2) + g_2 x_3(t) - \dot{\hat{\alpha}}_1(z_1). \tag{3.43}$$

The optimal value function $V_2^*(z_2)$ in second step can be defined with the dynamic error $z_2(t)$ and the optimal virtual control α_2^* as

$$\begin{aligned} V_2^*(z_2) &= \min_{\alpha_2 \in \Psi(\Omega_{z_2})} \Big(\int_t^\infty r_2(z_2(\tau), \alpha_2(z_2(\tau)) d\tau \Big) \\ &= \int_t^\infty r_2(z_2(\tau), \alpha_2^*(z_2(\tau))) d\tau, \end{aligned} \tag{3.44}$$

where $r_2 = z_2^2(t) + \alpha_2^2(z_2)$ is the cost function, and $\alpha_2(z_2)$ represents the virtual control. $\Psi(\Omega_{z_2})$ is the set of admissible control policies over Ω_{z_2}, where Ω_{z_2} denotes a compact set that includes the origin of the system. To minimize the tracking error $z_2(t)$, we can rewrite the optimal value function V_2^* as

$$V_2^*(z_2) = \beta_2 z_2^2(t) + V_2^o(z_2), \tag{3.45}$$

where β_2 is a positive designable constant and $V_2^o(z_2) = -\beta_2 z_2^2(t) + V_2^*(z_2)$ is a scalar-valued function. According to both (3.43) and (3.45), the HJB equation of the second step is

$$\begin{aligned} H_2\Big(z_2, \alpha_2^*, \frac{\partial V_2^*}{\partial z_2}\Big) &= z_2^2(t) + \alpha_2^{*2}(z_2) + \Big(2\beta_2 z_2(t) + \frac{\partial V_2^o(z_2)}{\partial z_2}\Big)(f_2(\bar{x}_2) + g_2\alpha_2^*(z_2) - \dot{\hat{\alpha}}_1(z_1)) \\ &= 0. \end{aligned} \tag{3.46}$$

Assuming that there is a solution and that it is unique, then by solving $\partial H_2/\partial \alpha_2^* = 0$, the optimal virtual control α_2^* is

$$\alpha_2^*(z_2) = g_2\Big(-\beta_2 z_2(t) - \frac{1}{2}\frac{\partial V_2^o(z_2)}{\partial z_2}\Big). \tag{3.47}$$

Utilizing an NN approximator to estimate $\partial V_2^o(z_2)/\partial z_2$ yields that

$$\frac{\partial V_2^o(z_2)}{\partial z_2} = W_2^{*T}\Gamma_2(z_2) + \varepsilon_2(z_2), \tag{3.48}$$

where $W_2^{*T} \in R^{m_2}$ signifies the ideal weight in the NN, and the item $\Gamma_2(z_2) \in R^{m_2}$ represents the basis function, and $\varepsilon_2(z_2)$ denotes the approximation error that is bounded. The gradient term $\partial V_2^*(z_2)/\partial z_2$ and the optimal virtual control $\alpha_2^*(z_2)$ become

$$\frac{\partial V_2^*(z_2)}{\partial z_2} = 2\beta_2 z_2(t) + W_2^{*T}\Gamma_2(z_2) + \varepsilon_2(z_2), \tag{3.49}$$

$$\alpha_2^*(z_2) = g_2\Big(-\beta_2 z_2(t) - \frac{1}{2}(W_2^{*T}\Gamma_2(z_2) + \varepsilon_2(z_2))\Big). \tag{3.50}$$

The optimal virtual control (3.50) cannot be used directly because the ideal weight vector W_2^{*T} is unknown. To achieve an effective and optimized control strategy, we implement an RL based on actor-critic NNs for deriving practical optimization

$$\frac{\partial \hat{V}_2^*}{\partial z_2} = 2\beta_2 z_2(t) + \hat{W}_{c2}^T(t)\Gamma_2(z_2), \tag{3.51}$$

$$\hat{\alpha}_2(z_2) = g_2(-\beta_2 z_2(t) - \frac{1}{2}\hat{W}_{a2}^T(t)\Gamma_2(z_2)), \tag{3.52}$$

where $\hat{V}_2^*$ is the estimation of V_2^*, $\hat{W}_{c2} \in R^{m2}$ represents the weight of critic NN, and $\hat{W}_{a2} \in R^{m2}$ denotes the actor NN weight. Upon inserting Eqs (3.51) and (3.52) into (3.46), we obtain the HJB equation

$$\begin{aligned} H_2(z_2, \hat{\alpha}_2, \hat{W}_{c2}) = {} & z_2^2(t) + \left(-\beta_2 g_2 z_2(t) - \frac{g_2}{2}\hat{W}_{a2}^T(t)\Gamma_2(z_2)\right)^2 \\ & + (2\beta_2 z_2(t) + \hat{W}_{c2}^T(t)\Gamma_2(z_2))\Big(f_2(\bar{x}_2) - \beta_2 g_2^2 z_2(t) - \frac{g_2^2}{2}\hat{W}_{a2}^T(t)\Gamma_2(z_2) - \dot{\hat{\alpha}}_1(z_1)\Big). \end{aligned} \tag{3.53}$$

Remark 3.4. To ensure the boundedness of the HJB function H_2, here we prove the boundedness of $\dot{\hat{\alpha}}_1(z_1)$.

The expression for $\dot{\alpha_1}(z_1)$ is as follows:

$$\dot{\hat{\alpha}}_1(z_1) = -\beta_1(\dot{x}_1(t) - \dot{y}_r(t)) - \frac{1}{2}\left(\dot{\hat{W}}_{a1}^T\Gamma_1(z_1) + \hat{W}_{a1}^T\dot{\Gamma}_1(z_1)\right).$$

Because the term $\dot{x}_1$ satisfies Lipschitz continuity, it is bounded. Obviously, $\dot{y}_r(t)$ and $\dot{\hat{W}}_{a1}^T\Gamma_1(z_1) + \hat{W}_{a1}^T\dot{\Gamma}_1(z_1)$ are also bounded. Consequently, the function $\dot{\alpha_1}(z_1)$, which consists of these bounded terms, is also bounded. Furthermore, $\dot{\alpha_i}(z_i), i = 1, \ldots, 3$ is bounded at each step, although this will not be shown hereafter.

To optimize the function $E_2(t) = e_2^2(t)/2$, we employ the gradient descent methodology. Then we can derive the subsequent update law for the critic NN weight $\hat{W}_{c2}(t)$,

$$\begin{aligned} \dot{\hat{W}}_{c2}(t) = {} & -\frac{\mu_{c2}}{\|\omega_2\|^2 + 1}\omega_2(t)\Big(\omega_2^T(t)\hat{W}_{c2}(t) - (\beta_2^2 g_2^2 - 1)z_2^2(t) \\ & + 2\beta_2 z_2(f_2(\bar{x}_2) - \dot{\hat{\alpha}}_1(z_1)) + \frac{g_2^2}{4}\hat{W}_{a2}^T\Gamma_2(z_2)\Gamma_2^T(z_2)\hat{W}_{a2}\Big), \end{aligned} \tag{3.54}$$

where $\mu_{c2} > 0$ is the learning rate and $\omega_2 = \Gamma_2(z_2)(f_2(\bar{x}_2) - \beta_2 g_2^2 z_2(t) - (g_2^2/2)\hat{W}_{a2}^T\Gamma_2(z_2) - \dot{\hat{\alpha}}_1(z_1)) \in R^{m_2}$. The renewal law of actor NN weight $\hat{W}_{a2}(t)$ is designed as

$$\begin{aligned} \dot{\hat{W}}_{a2}(t) = {} & \frac{g_2^2}{2}\Gamma_2(z_2)z_2(t) - \mu_{a2}\Gamma_2(z_2)\Gamma_2^T(z_2)\hat{W}_{a2}(t) \\ & + \frac{\mu_{c2}g_2^2}{4(\|\omega_2\|^2 + 1)}\Gamma_2(z_2)\Gamma_2^T(z_2)\hat{W}_{a2}(t)\omega_2^T(t)\hat{W}_{c2}(t), \end{aligned} \tag{3.55}$$

where $\mu_{a2} > 0$ is the learning rate of the actor NN.

By introducing the error variable in the third step as $z_3(t) = x_3(t) - \hat{\alpha_2}(z_2)$, we can rewrite (3.43) as

$$\dot{z}_2(t) = f_2(\bar{x}_2) + g_2(z_3(t) + \hat{\alpha}_2(z_2)) - \dot{\hat{\alpha}}_1(z_1). \tag{3.56}$$

Design the Lyapunov function as

$$L_2(t) = L_1(t) + \frac{1}{2}z_2^2(t) + \frac{1}{2}\tilde{W}_{c2}^T(t)\tilde{W}_{c2}(t) + \frac{1}{2}\tilde{W}_{a2}^T(t)\tilde{W}_{a2}(t), \tag{3.57}$$

where $\tilde{W}_{c2}(t) = \hat{W}_{c2}(t) - W_2^*$ and $\tilde{W}_{a2}(t) = \hat{W}_{a2}(t) - W_2^*$. Its derivative is as follows:

$$\dot{L}_2(t) = \dot{L}_1(t) + z_2(t)\dot{z}_2(t) + \tilde{W}_{c2}^T(t)\dot{\hat{W}}_{c2}(t) + \tilde{W}_{a2}^T(t)\dot{\hat{W}}_{a2}(t). \tag{3.58}$$

Inserting (3.52), (3.54), (3.55), and (3.56) into (3.58), we have

$$\begin{aligned}\dot{L}_2(t) = \ & \dot{L}_1(t) + g_2 z_2(t) z_3(t) + f_2(\bar{x}_2) z_2(t) - \beta_2 g_2^2 z_2^2(t) - z_2(t)\dot{\hat{\alpha}}_1(z_1) \\ & + \frac{\mu_{c2}}{4(\|\omega_2\|^2+1)} \tilde{W}_{a2}^T(t)\Gamma_2(z_2)\Gamma_2^T(z_2)\hat{W}_{a2}(t)\omega_2^T\hat{W}_{c2}(t) \\ & - \frac{g_2^2}{2} z_2(t)\hat{W}_{a2}^T(t)\Gamma_2(z_2) + \frac{g_2^2}{2}\tilde{W}_{a2}^T(t)\Gamma_2(z_2)z_2(t) - \mu_{a2}\tilde{W}_{a2}^T(t)\Gamma_2(z_2)\Gamma_2^T(z_2)\hat{W}_{a2} \\ & - \frac{\mu_{c2}}{\|\omega_2\|^2+1}\tilde{W}_{c2}^T(t)\omega_2\Big(\omega_2^T\hat{W}_{c2}(t) - (\beta_2^2 g_2^2 - 1)z_2^2(t) + 2\beta_2 z_2(t)(f_2(\bar{x}_2) - \dot{\hat{\alpha}}_1) \\ & + \frac{g_2^2}{4}\hat{W}_{a2}^T(t)\Gamma_2(z_2)\Gamma_2^T(z_2)\hat{W}_{a2}(t)\Big).\end{aligned} \tag{3.59}$$

Analogous to the first step, we can obtain the inequality shown as follows:

$$\begin{aligned}\dot{L}_2(t) \le \ & \dot{L}_1(t) + z_3^2(t) - (\beta_2 g_2^2 - g_2^2 - 1)z_2^2(t) \\ & - \left(\frac{\mu_{a2}}{2} - \frac{\mu_{c2}^2 g_2^4}{2} - \frac{1}{32}W_2^{*T}\omega_2\omega_2^T W_2^*\right)\tilde{W}_{a2}^T(t)\Gamma_2(z_2)\Gamma_2^T(z_2)\tilde{W}_{a2}(t) \\ & - \frac{1}{\|\omega_2\|^2+1}\left(\frac{\mu_{c2}}{2} - \frac{1}{32}W_2^{*T}\Gamma_2(z_2)\Gamma_2^T(z_2)W_2^*\right)\tilde{W}_{c2}^T(t)\omega_2\omega_2^T\tilde{W}_{c2}(t) \\ & - \left(\frac{\mu_{a2}}{2} - \frac{\mu_{c2}^2 g_2^4}{2}\right)\hat{W}_{a2}^T(t)\Gamma_2(z_2)\Gamma_2^T(z_2)\hat{W}_{a2}(t) + \frac{1}{2}f_2^2(\bar{x}_2) + \frac{1}{2}\dot{\hat{\alpha}}_1^2 \\ & + \frac{\mu_{a2}+g_2^2}{2}\left(W_2^{*T}\Gamma_2(z_2)\right)^2 + \frac{\mu_{c2}}{2}\epsilon_2^2(t).\end{aligned} \tag{3.60}$$

Rewrite (3.60) as follows:

$$\begin{aligned}\dot{L}_2(t) \le \ & (-a_1 \| \xi_1(t)\|^2 + c_1) - \xi_2^T(t)A_2(t)\xi_2(t) + C_2(t) + z_3^2(t) \\ & - \left(\frac{\mu_{a2}}{2} - \frac{\mu_{c2}^2 g_2^4}{2}\right)\hat{W}_{a2}^T(t)\Gamma_2(z_2)\Gamma_2^T(z_2)\hat{W}_{a2}(t),\end{aligned} \tag{3.61}$$

with the matrix $\xi_2(t) = [z_2(t), \tilde{W}_{a2}^T(t), \tilde{W}_{c2}^T(t)]^T$ and the term $C_2(t) = \frac{1}{2}f_2^2(\bar{x}_2) + \frac{1}{2}\dot{\hat{\alpha}}_1^2 + \frac{\mu_{c2}}{2}\epsilon_2^2(t) + \frac{\mu_{a2}+g_2^2}{2}\left(W_2^{*T}\Gamma_2(z_2)\right)^2$.

In order to satisfy that the matrix $A_2(t)$ is positive definite, the parameters are designed as follows:

$$\beta_2 > \frac{1}{g_2^2} + 1, \quad \mu_{c2} > \frac{1}{16}\lambda_2, \quad \mu_{a2} > \mu_{c2}^2 g_2^4 + \frac{\zeta_2}{16}W_2^{*T}W_2^*, \tag{3.62}$$

where λ_2 is the maximal eigenvalue of matrix $\Lambda_2 = W_2^{*T}\Gamma_2(z_2)\Gamma_2^T(z_2)W_2^*$. Consequently, we have

$$\dot{L}_2(t) < z_3^2(t) - a_1\|\xi_1(t)\|^2 + c_1 - a_2\|\xi_2(t)\|^2 + c_2, \tag{3.63}$$

where a_2 is the minimum eigenvalue of $A_2(t)$ and c_2 is the maximum value of $C_2(t)$.

Step 3: Define the tracking error between $x_3(t)$ and $\hat{\alpha_2}(z_2)$ for the third step as $z_3(t) = x_3(t) - \hat{\alpha_2}(z_2)$. Its time derivative along the pure-feedback system (2.3) is

$$\dot{z}_3(t) = x_4(t) - \dot{\hat{\alpha}}_2(z_2). \tag{3.64}$$

In the process, we first define the virtual control term $\alpha_3(z_3)$ and further introduce its optimal counterpart, denoted as $\alpha_3^*(z_3)$. Describe the performance index function $V_3^*(z_3)$ as

$$\begin{aligned} V_3^*(z_3) &= \min_{\alpha_3 \in \Psi(\Omega_{z_3})} \left(\int_t^\infty r_3(z_3(\tau), \alpha_2(z_3(\tau))) d\tau \right) \\ &= \int_t^\infty r_3(z_3(\tau), \alpha_3^*(z_3(\tau))) d\tau, \end{aligned} \tag{3.65}$$

where $r_3 = z_3^2(t) + \alpha_3^2(z_3)$ is the cost function, and the set Ω_{z_3} represents a compact domain that encompasses the origin of the system. Rewrite the optimal value function V_3^* as

$$V_3^*(z_3) = \beta_3 z_3^2(t) + V_3^o(z_3), \tag{3.66}$$

where β_3 is a positive designable constant and $V_3^o(z_3) = -\beta_3 z_3^2(t) + V_3^*(z_3)$ is a scalar-valued function. Then, we can derive the HJB equation as follows:

$$\begin{aligned} H_3\left(z_3, \alpha_3^*, \frac{\partial V_3^*}{\partial z_3}\right) &= \alpha_3^{*2}(z_3) + \left(2\beta_3 z_3(t) + \frac{\partial V_3^o(z_3)}{\partial z_3}\right)(\alpha_3^*(z_3) - \dot{\hat{\alpha}}_2(z_2)) \\ &= 0. \end{aligned} \tag{3.67}$$

By solving $\partial H_3 / \partial \alpha_3^* = 0$, the optimal virtual control α_3^* is

$$\alpha_3^*(z_3) = -\beta_3 z_3(t) - \frac{1}{2} \frac{\partial V_3^o(z_3)}{\partial z_3}. \tag{3.68}$$

By applying NN, the part $\partial V_3^o(z_3)/\partial z_3$ can be approximated as

$$\frac{\partial V_3^o(z_3)}{\partial z_3} = W_3^{*T} \Gamma_3(z_3) + \varepsilon_3(z_3), \tag{3.69}$$

where $W_3^{*T} \in R^{m_3}$ represents the ideal weight, $\Gamma_3(z_3) \in R^{m_3}$ denotes the basis function in the NN, and $\varepsilon_3(z_3)$ signifies the bounded approximation error. With (3.69), the gradient term $\partial V_3^*(z_3)/\partial z_3$ and the optimal virtual control $\alpha_3^*(z_3)$ are obtained:

$$\frac{\partial V_3^*(z_3)}{\partial z_3} = 2\beta_3 z_3(t) + W_3^{*T} \Gamma_3(z_3) + \varepsilon_3(z_3), \tag{3.70}$$

$$\alpha_3^*(z_3) = -\beta_3 z_3(t) - \frac{1}{2}(W_3^{*T} \Gamma_3(z_3) + \varepsilon_3(z_3)). \tag{3.71}$$

Since W_3^* is not directly available, an RL based on the actor-critic architecture is employed as

$$\frac{\partial \hat{V}_3^*}{\partial z_3} = 2\beta_3 z_3(t) + \hat{W}_{c3}^T(t) \Gamma_3(z_3), \tag{3.72}$$

$$\hat{\alpha}_3(z_3) = -\beta_3 z_3(t) - \frac{1}{2}\hat{W}_{a3}^T(t)\Gamma_3(z_3), \tag{3.73}$$

where $\hat{V}_3^*$ is the estimation of V_3^*, $\hat{W}_{c3}$ is the weight of critic NN, and $\hat{W}_{a3}$ is the weight of actor NN. Substituting (3.72) and (3.73) into (3.67), we can rewrite the HJB equation as

$$\begin{aligned} H_3\Big(z_3, \hat{\alpha}_3, \hat{W}_{c3}\Big) = {} & z_3^2(t) + \left(-\beta_3 z_3(t) - \frac{1}{2}\hat{W}_{a3}^T(t)\Gamma_3(z_3)\right)^2 \\ & + (2\beta_3 z_3(t) + \hat{W}_{c3}^T(t)\Gamma_3(z_3))\Big(- \beta_3 z_3(t) - \frac{1}{2}\hat{W}_{a3}^T(t)\Gamma_3(z_3) - \dot{\hat{\alpha}}_2(z_2)\Big). \end{aligned} \tag{3.74}$$

To minimize $E_3(t) = e_3^2(t)/2$, design the following updating laws for the weights in the critic and actor NNs

$$\begin{aligned} \dot{\hat{W}}_{c3}(t) = {} & -\frac{\mu_{c3}}{\|\omega_3\|^2 + 1}\omega_3(t)\Big(\omega_3^T(t)\hat{W}_{c3}(t) - (\beta_3^2 - 1)z_3^2(t) - 2\beta_3 z_3\dot{\hat{\alpha}}_2(z_2) \\ & + \frac{1}{4}\hat{W}_{a3}^T\Gamma_3(z_3)\Gamma_3^T(z_3)\hat{W}_{a3}\Big), \end{aligned} \tag{3.75}$$

$$\begin{aligned} \dot{\hat{W}}_{a3}(t) = {} & \frac{1}{2}\Gamma_3(z_3)z_3(t) - \mu_{a3}\Gamma_3(z_3)\Gamma_3^T(z_3)\hat{W}_{a3}(t) \\ & + \frac{\mu_{c3}}{4(\|\omega_3\|^2 + 1)}\Gamma_3(z_3)\Gamma_3^T(z_3)\hat{W}_{a3}(t)\omega_3^T(t)\hat{W}_{c3}(t), \end{aligned} \tag{3.76}$$

where $\mu_{a3} > 0$ and $\mu_{c3} > 0$ represent the designable learning rates of the actor NN and critic NN, respectively, and $\omega_3 = \Gamma_3(z_3)(- \beta_3 z_3(t) - (1/2)\hat{W}_{a3}^T\Gamma_3(z_3) - \dot{\hat{\alpha}}_2(z_2)) \in R^{m_3}$.

The tracking error in the step 4 is written as $z_4(t) = x_4(t) - \hat{\alpha_3}(z_3)$, then (3.64) is replaced as

$$\dot{z}_3(t) = z_4(t) + \hat{\alpha}_3(z_3) - \dot{\hat{\alpha}}_2(z_2). \tag{3.77}$$

The Lyapunov function can be formulated as described below:

$$L_3(t) = \sum_{k=1}^{2} L_k(t) + \frac{1}{2}z_3^2(t) + \frac{1}{2}\tilde{W}_{c3}^T(t)\tilde{W}_{c3}(t) + \frac{1}{2}\tilde{W}_{a3}^T(t)\tilde{W}_{a3}(t), \tag{3.78}$$

where $\tilde{W}_{c3}(t) = \hat{W}_{c3}(t) - W_3^*$ represents the estimation error of the critic NN, while $\tilde{W}_{a3}(t) = \hat{W}_{a3}(t) - W_3^*$ is the actor NN estimation error. The derivative of the Lyapunov quadratic scalar function (3.78) is

$$\dot{L}_3(t) = \sum_{k=1}^{2} \dot{L}_k(t) + z_3(t)\dot{z}_3(t) + \tilde{W}_{c3}^T(t)\dot{\hat{W}}_{c3}(t) + \tilde{W}_{a3}^T(t)\dot{\hat{W}}_{a3}(t). \tag{3.79}$$

The equation along with (3.73), (3.75), (3.76), and (3.77) is

$$\begin{aligned} \dot{L}_3(t) = {} & \sum_{k=1}^{2} \dot{L}_k(t) + z_3(t)z_4(t) - \beta_3 z_3^2(t) - z_3(t)\dot{\hat{\alpha}}_2(z_2) - \frac{1}{2}z_3(t)\hat{W}_{a3}(t)\Gamma_3(z_3) \\ & + \frac{1}{2}\tilde{W}_{a3}^T(t)\Gamma_3(z_3)z_3(t) - \mu_{a3}\tilde{W}_{a3}^T(t)\Gamma_3(z_3)\Gamma_3^T(z_3)\hat{W}_{a3}(t) \end{aligned}$$

$$+\frac{\mu_{c3}}{4(\|\omega_3\|^2+1)}\tilde{W}_{a3}^T(t)\Gamma_3(z_3)\Gamma_3^T(z_3)\hat{W}_{a3}(t)\omega_3^T(t)\hat{W}_{c3}(t)$$
$$-\frac{\mu_{c3}}{\|\omega_3\|^2+1}\tilde{W}_{c3}^T(t)\omega_3\Big(\omega_3^T\hat{W}_{c3}(t)-(\beta_3^2-1)z_3^2(t)-2\beta_3 z_3(t)\dot{\hat{\alpha}}_2$$
$$+\frac{1}{4}\hat{W}_{a3}^T(t)\Gamma_3(z_3)\Gamma_3^T(z_3)\hat{W}_{a3}(t)\Big). \tag{3.80}$$

Applying the control (3.73), (3.75), and (3.76), similar with step 1, we have the result

$$\begin{aligned}\dot{L}_3(t)\le&\sum_{k=1}^{2}\dot{L}_k(t)+z_4^2(t)-(\beta_3-2)z_3^2(t)\\&-\Big(\frac{\mu_{a3}}{2}-\frac{\mu_{c3}^2}{2}-\frac{1}{32}W_3^{*T}\omega_3\omega_3^TW_3^*\Big)\tilde{W}_{a3}^T(t)\Gamma_3(z_3)\Gamma_3^T(z_3)\tilde{W}_{a3}(t)\\&-\frac{1}{\|\omega_3\|^2+1}\Big(\frac{\mu_{c3}}{2}-\frac{1}{32}W_3^{*T}(z_3)\Gamma_3^T(z_3)W_3^*\Big)\tilde{W}_{c3}^T(t)\omega_3\omega_3^T\tilde{W}_{c3}(t)\\&-\Big(\frac{\mu_{a3}}{2}-\frac{\mu_{c3}^2}{2}\Big)\hat{W}_{a3}^T(t)\Gamma_3(z_3)\Gamma_3^T(z_3)\hat{W}_{a3}(t)+\frac{1}{2}\dot{\hat{\alpha}}_2^2\\&+\frac{\mu_{a3}+1}{2}(W_3^{*T}\Gamma_3(z_3))^2+\frac{\mu_{c3}}{2}\epsilon_3^2(t).\end{aligned} \tag{3.81}$$

Rewrite (3.81) as follows:

$$\begin{aligned}\dot{L}_3(t)\le&\sum_{k=1}^{2}\Big(-a_k\|\xi_k(t)\|^2+c_k\Big)-\xi_3^T(t)A_3(t)\xi_3(t)+C_3(t)+z_4^2(t)\\&-\Big(\frac{\mu_{a3}}{2}-\frac{\mu_{c3}^2}{2}\Big)\hat{W}_{a3}^T(t)\Gamma_3(z_3)\Gamma_3^T(z_3)\hat{W}_{a3}(t),\end{aligned} \tag{3.82}$$

where $\xi_3(t)=[z_3(t),\tilde{W}_{a3}^T(t),\tilde{W}_{c3}^T(t)]^T$, $C_3(t)=\frac{1}{2}\dot{\hat{\alpha}}_2^2+\frac{\mu_{a3}+1}{2}(W_3^{*T}\Gamma_3(z_3))^2+\frac{\mu_{c3}}{2}\epsilon_3^2(t)$.

Select parameters within the following intervals:

$$\beta_3>2,\quad \mu_{c3}>\frac{1}{16}\lambda_3,\quad \mu_{a3}>\mu_{c3}^2+\frac{\zeta_3}{16}W_3^{*T}W_3^*, \tag{3.83}$$

where λ_3 is the maximal eigenvalue of matrix $\Lambda_3=W_3^{*T}\Gamma_3(z_3)\Gamma_3^T(z_3)W_3^*$. We have

$$\dot{L}_3(t)<z_4^2(t)+\sum_{k=1}^{3}(-a_k\|\xi_k(t)\|^2+c_k), \tag{3.84}$$

where a_3 is the lower bound on the minimum eigenvalue of $A_3(t)$ and c_3 is the maximum value of $C_3(t)$.

Step 4: The actual input u is obtained in the final step. The tracking error is $z_4(t)=x_4(t)-\hat{\alpha}_3(z_3)$, then

$$\dot{z}_4(t)=f_4(\bar{x}_4)+g_4u-\dot{\hat{\alpha}}_3(z_3). \tag{3.85}$$

The performance index function in the final step is described as

$$V_4^*(z_4) = \min_{u\in\Psi(\Omega_{z_4})}\Big(\int_t^\infty r_4(z_4(\tau),u(z_4(\tau))d\tau\Big) = \int_t^\infty r_4(z_4(\tau),u^*(z_4(\tau)))d\tau, \tag{3.86}$$

where u^* is the optimal actual input and $r_4 = z_4^2(t) + u^2(z_4)$ represents the cost function.

Without prejudice to generality, the actual controller $u(z_4)$ can be obtained as follows:

$$u(z_4) = g_4(-\beta_4 z_4(t) - \frac{1}{2}\hat{W}_{a4}^T(t)\Gamma_4(z_4)), \tag{3.87}$$

where $\hat{W}_{a4}$ is the weight of actor NN. With the critic and actor updating law

$$\dot{\hat{W}}_{c4}(t) = -\frac{\mu_{c4}}{\|\omega_2\|^2+1}\omega_4(t)\Big(\omega_4^T(t)\hat{W}_{c4}(t) - (\beta_4^2 g_4^2 - 1)z_4^2(t) + 2\beta_4 z_4(f_4(\bar{x}_4) - \dot{\hat{\alpha}}_3(z_3)) + \frac{g_4^2}{4}\hat{W}_{a4}^T\Gamma_4(z_4)\Gamma_4^T(z_4)\hat{W}_{a4}\Big), \tag{3.88}$$

$$\dot{\hat{W}}_{a4}(t) = \frac{g_4^2}{2}\Gamma_4(z_4)z_4(t) - \mu_{a4}\Gamma_4(z_4)\Gamma_4^T(z_4)\hat{W}_{a4}(t) + \frac{\mu_{c4}g_4^2}{4(\|\omega_4\|^2+1)}\Gamma_4(z_4)\Gamma_4^T(z_4)\hat{W}_{a4}(t)\omega_4^T(t)\hat{W}_{c4}(t), \tag{3.89}$$

where $\mu_{c4} > 0$ and $\mu_{a4} > 0$ are the critic and actor learning rate, separately, and $\omega_4 = \Gamma_4(z_4)(f_4(\bar{x}_4) - \beta_4 g_4^2 z_4(t) - (g_4^2/2)\hat{W}_{a4}^T\Gamma_4(z_4) - \dot{\hat{\alpha}}_3(z_3)) \in R^{m_4}$.

In the final step, the Lyapunov quadratic scalar function is chosen as

$$L_4(t) = \sum_{k=1}^3 L_k(t) + \frac{1}{2}\tilde{W}_{c4}^T(t)\tilde{W}_{c4}(t) + \frac{1}{2}z_4^2(t) + \frac{1}{2}\tilde{W}_{a4}^T(t)\tilde{W}_{a4}(t), \tag{3.90}$$

where $\tilde{W}_{c4}(t) = \hat{W}_{c4}(t) - W_4^*$ is the critic NN estimation error, and $\tilde{W}_{a4}(t) = \hat{W}_{a4}(t) - W_4^*$ is estimation error of the actor NN. The derivative of (3.90) is

$$\dot{L}_4(t) = \sum_{k=1}^3 \dot{L}_k(t) + z_4(t)\dot{z}_4(t) + \tilde{W}_{a4}^T(t)\dot{\hat{W}}_{a4}(t) + \tilde{W}_{c4}^T(t)\dot{\hat{W}}_{c4}(t). \tag{3.91}$$

According to (3.87), (3.88), and (3.89), we have

$$\dot{L}_4(t) = \sum_{k=1}^3 \dot{L}_k(t) + f_4(\bar{x}_4)z_4(t) - \beta_4 g_4^2 z_4^2(t) - z_4(t)\dot{\hat{\alpha}}_3 - \frac{g_4^2}{2}z_4(t)\hat{W}_{a4}(t)\Gamma_4(z_4) + \frac{g_4^2}{2}\tilde{W}_{a4}^T(t)\Gamma_4(z_4)z_4(t) - \mu_{a4}\tilde{W}_{a4}^T(t)\Gamma_4(z_4)\Gamma_4^T(z_4)\hat{W}_{a4}(t) + \frac{\mu_{c4}}{4(\|\omega_4\|^2+1)}\tilde{W}_{a4}^T(t)\Gamma_4(z_4)\Gamma_4^T(z_4)\hat{W}_{a4}(t)\omega_4^T(t)\hat{W}_{c4}(t)$$

$$-\frac{\mu_{c4}}{\|\omega_4\|^2+1}\tilde{W}_{c4}^T(t)\omega_4\Big(\omega_4^T\hat{W}_{c4}(t)-(\beta_4^2g_4^2-1)z_4^2(t)+2\beta_4z_4(t)(f_4(\bar{x}_4)-\dot{\hat{\alpha}}_3)$$
$$+\frac{g_4^2}{4}\hat{W}_{a4}^T(t)\Gamma_4(z_4)\Gamma_4^T(z_4)\hat{W}_{a4}(t)\Big). \tag{3.92}$$

Similar to the first step, we can also deduce the following result

$$\begin{aligned}\dot{L}_4(t)\leq&\sum_{k=1}^{3}\dot{L}_k(t)-(\beta_4g_4^2-g_4^2-1)z_4^2\\&-\Big(\frac{\mu_{a4}}{2}-\frac{\mu_{c4}^2g_4^4}{2}-\frac{1}{32}W_4^{*T}\omega_4\omega_4^TW_4^*\Big)\tilde{W}_{a4}^T(t)\Gamma_4(z_4)\Gamma_4^T(z_4)\tilde{W}_{a4}(t)\\&-\frac{1}{\|\omega_4\|^2+1}\Big(\frac{\mu_{c4}}{2}-\frac{1}{32}W_4^{*T}\Gamma_4(z_4)\Gamma_4^T(z_4)W_4^*\Big)\tilde{W}_{c4}^T(t)\omega_4\omega_4^T\tilde{W}_{c4}(t)\\&-\Big(\frac{\mu_{a4}}{2}-\frac{\mu_{c4}^2g_4^4}{2}\Big)\hat{W}_{a4}^T(t)\Gamma_4(z_4)\Gamma_4^T(z_4)\hat{W}_{a4}(t)+\frac{1}{2}f_4^2(\bar{x}_4)+\frac{1}{4}\dot{\hat{\alpha}}_3^2(t)\\&+\frac{\mu_{a4}+g_4^2}{2}(W_4^{*T}\Gamma_4(z_4))^2+\frac{\mu_{c4}}{2}\epsilon_4^2(t).\end{aligned} \tag{3.93}$$

Rewrite (3.93) as follows:

$$\begin{aligned}\dot{L}_4(t)\leq&\sum_{k=1}^{3}\Big(-a_k\|\xi_k(t)\|^2+c_k\Big)-\xi_4^T(t)A_4(t)\xi_4(t)+C_4(t)\\&-\Big(\frac{\mu_{a4}}{2}-\frac{\mu_{c4}^2g_4^4}{2}\Big)\hat{W}_{a4}^T(t)\Gamma_4(z_4)\Gamma_4^T(z_4)\hat{W}_{a4}(t),\end{aligned} \tag{3.94}$$

with the matrix $\xi_4(t)=[z_4(t),\tilde{W}_{a4}^T(t),\tilde{W}_{c4}^T(t)]^T$, and the term $C_4(t)=\frac{1}{2}f_4^2(\bar{x}_4)+\frac{1}{2}\dot{\hat{\alpha}}_3^2+\frac{\mu_{a4}+g_4^2}{2}(W_4^{*T}\Gamma_4(z_4))^2+\frac{\mu_{c4}}{2}\epsilon_4^2(t)$.

To ensure system stability, the design parameters β_4, μ_{c4}, and μ_{a4} must satisfy

$$\beta_4>\frac{1}{g_4^2}+1,\quad \mu_{c4}>\frac{1}{16}\lambda_4,\quad \mu_{a4}>\mu_{c4}^2g_4^4+\frac{\zeta_4}{16}W_4^{*T}W_4^*, \tag{3.95}$$

where λ_4 is the maximal eigenvalue of matrix $\Lambda_4=W_4^{*T}\Gamma_4(z_4)\Gamma_4^T(z_4)W_4^*$.

The selection of a_4 as the infimum over $t\geq 0$ of the minimum eigenvalue of $A_4(t)$ and c4 as the supremum over $t\geq 0$ of $C_4(t)$ allows Eq (3.94) to be reformulated as follows:

$$\dot{L}(t)<\sum_{k=1}^{4}(-a_k\|\xi_k(t)\|^2+c_k). \tag{3.96}$$

Based on the above derivation, we can achieve the objectives:

1) Within the closed-loop control framework, all error signals, designated as $z_i(t)$ for $i=1,\cdots,4$ and the weight estimation errors, expressed as $\tilde{W}_{ci}(t)$ and $\tilde{W}_{ai}(t)$ for $i=1,\cdots,4$, are assured to be SGUUB in anpredictable and desirable fashion;

2) The single-link manipulator joint angular position $q_1(t)$ exhibits the capability to follow the desired trajectory y_r in a predictable and desirable manner.

Prove as follows:

1) The inequality (3.96) can be

$$\dot{L}(t) < -aL(t) + c,$$

where a is the minimal of $a_k, k = 1, 2, \cdots, 4$ and c is the sum of $c_k, k = 1, 2, \cdots, 4$. According to Lemma 2.1, we can clearly get the following result:

$$L(t) < e^{-at}L(0) + \frac{c}{a}(1 - e^{-at}),$$

which can prove that that control objective 1 is valid.

2) Define $L_z(t) = (1/2)\sum_{k=1}^{4} z_k^2(t)$. According to the Eqs (3.18), (3.56), (3.77), and (3.85), we have

$$\begin{aligned}\dot{L}_z(t) = {} & z_1(t)(\hat{\alpha}_1(z_1) + z_2(t) - \dot{y}_r(t)) + z_2(t)\Big(f_2(\bar{x}_2) + g_2(\hat{\alpha}_2(z_2) + z_3(t)) - \dot{\hat{\alpha}}_1(z_1)\Big) \\ & + z_3(t)(z_4(t) + \hat{\alpha}_3(z_3) - \dot{\hat{\alpha}}_2(z_2)) + z_4(t)(f_4(\bar{x}_4) + g_4 u(t) - \dot{\hat{\alpha}}_3(z_3)). \end{aligned} \tag{3.97}$$

Substituting (3.11), (3.52), (3.73), and (3.87) into (3.97), we have the following result:

$$\begin{aligned}\dot{L}_z(t) = {} & -\beta_1 z_1(t)^2 + z_1(t)z_2(t) - z_1(t)\dot{y}_r - \frac{1}{2}z_1(t)\hat{W}_{a1}^T\Gamma_1 \\ & -g_2^2\beta_2 z_2(t)^2 + g_2 z_2(t)z_3(t) - z_2(t)\dot{\hat{\alpha}}_1 - \frac{g_2^2}{2}z_2(t)\hat{W}_{a2}^T\Gamma_2 + z_2(t)f_2(\bar{x}_2) \\ & -\beta_3 z_3(t)^2 + z_3(t)z_4(t) - z_1(t)\dot{\hat{\alpha}}_2 - \frac{1}{2}z_3(t)\hat{W}_{a3}^T\Gamma_3 \\ & -g_4^2\beta_4 z_4(t)^2 - z_4(t)\dot{\hat{\alpha}}_3 - \frac{g_4^2}{2}z_4(t)\hat{W}_{a4}^T\Gamma_4 + z_4(t)f_4(\bar{x}_4). \end{aligned} \tag{3.98}$$

Using Young's inequality, it is clear that we can get the following result:

$$\begin{aligned}\dot{L}_z(t) \leq {} & -(\beta_1 - 2)z_1^2(t) - (\beta_2 g_2^2 - g_2^2 - 1)z_2^2(t) \\ & -(\beta_3 - 2)z_3^2(t) - (\beta_4 g_4^2 - g_4^2 - 1)z_4^2(t) + D(t), \end{aligned} \tag{3.99}$$

where $D(t) = (1/2)f_2^2(\bar{x}_2) + (1/2)f_4^2(\bar{x}_4) + (1/2)\sum_{k=1}^{3}\dot{\hat{\alpha}}_k^2 + (1/2)\dot{\hat{y}}_r^2(t) + (1/2)(\hat{W}_{a1}^T(t)\Gamma_1(z_1))^2 + (1/2)(\hat{W}_{a3}^T(t)\Gamma_3(z_3))^2 + (g_2^2/2)(\hat{W}_{a2}^T(t)\Gamma_2(z_2))^2 + (g_4^2/2)(\hat{W}_{a4}^T(t)\Gamma_4(z_4))^2$ is bounded. A constant ρ exists, bounding $|D(t)|$. Hence, the above result can be described as

$$\dot{L}_z(t) < -\beta L_z(t) + \rho,$$

where β is the minimal of $\{\beta_1 - 2, \beta_2 g_2^2 - g_2^2 - 1, \beta_3 - 2, \beta_4 g_4^2 - g_4^2 - 1\}$. Obviously, we can get the following result:

$$L_z(t) < e^{-\beta t}L_z(0) + \frac{\rho}{\beta}(1 - e^{-\beta t}).$$

It implies that increasing β sufficiently ensures desired tracking accuracy and control performance.

Ultimately, according to (3.11), (3.52), (3.73), and (3.87), we design an adaptive tracking control strategy for the flexible-joint manipulator. The details of this control method are illustrated in Figure 2.

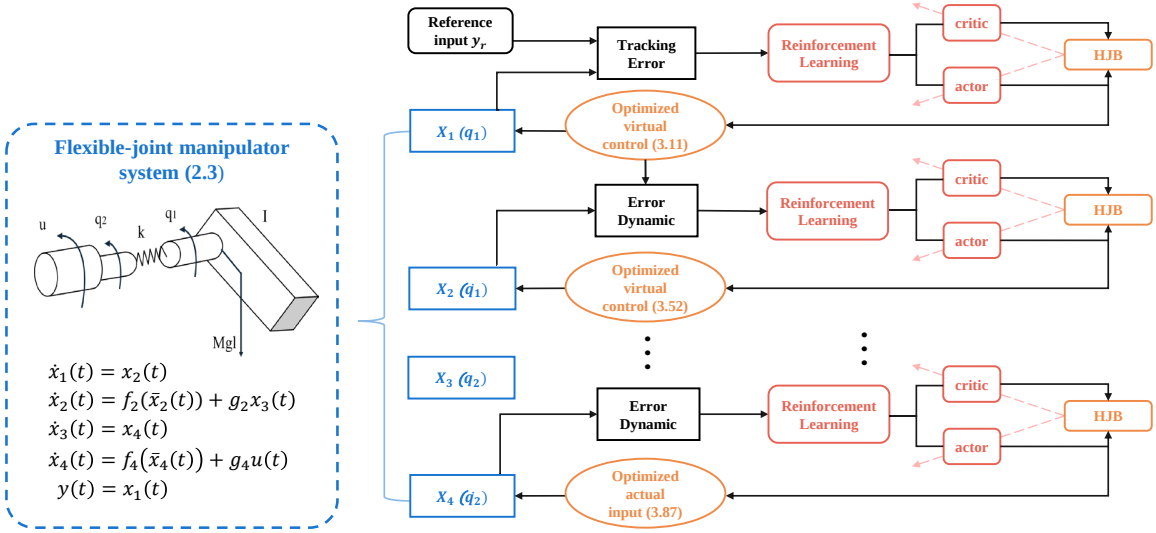

Figure 2. The control block diagram (the dotted line indicates back propagation and training the NNs).

4. Simulation

To enhance the validation of the method's effectiveness in controlling a flexible-joint robotic manipulator, numerical simulations were conducted. Table 1 provides the key parameters relevant to the single-link manipulator. The initial conditions are set to $q_1(0) = 8\text{deg}$, $\dot{q}_1(0) = 0\text{deg/s}$, $q_2(0) = 10\text{deg}$, and $\dot{q}_2(0) = 0\text{deg/s}$, and we choose the desired trajectory as $y_r(t) = 28\sin(3t/4)$, shown in Figure 3.

To achieve the tracking objectives, the design of the virtual controller for the first three steps and the input signal for the final step correspond to (3.11), (3.52), (3.73), and (3.87), respectively, where the designable parameters are set as $[\beta_1, \beta_2, \beta_3, \beta_4] = [6.00, 2.04, 11.00, 2.01]$. The NN at each step has 36 neurons with centers uniformly distributed in the range $[-6, 6]$, and the widths $\varphi_i, i = 1, \cdots, 4$ of the Gaussian functions of the basis functions Γ_i are all chosen to be 2. The update rate of the critic weights at each step corresponds to (3.15), (3.54), (3.75), and (3.88), respectively, and the designable parameters learning rate and initial weights are $[\mu_{c1}, \mu_{c2}, \mu_{c3}, \mu_{c4}] = [0.4, 0.4, 0.4, 0.4]$, $W_c i(0) = [0.5]_{36\times 1}, i = 1, \cdots, 4$. The update rate of the actor weights at each step corresponds to (3.17), (3.55), (3.76), and (3.89), respectively, where the designable parameters learning rate and initial weights are $[\mu_{a1}, \mu_{a2}, \mu_{a3}, \mu_{a4}] = [300, 300, 300, 300]$, $W_a i(0) = [0.4]_{36\times 1}, i = 1, \cdots, 4$.

Table 1. Parameters of the single-link manipulator.

Parameters	Description	Values	Unit
I	the mass inertia	20	$kg \cdot m^2$
J	the actuator inertia	0.1	$kg \cdot m^2$
M	the link mass	0.1	kg
g	gravity acceleration	9.8	m/s^2
l	the link's center of gravity position	0.1	m
k	the joint flexible	100	$N \cdot m/rad$

Simulation result: The individual figures depict the results of the simulation process. The actual output $y(t)$ and the expected trajectory $y_r(t)$ are demonstrated in Figure 3, which it is clear to see that the actual output is able to better align with the desired output. Figure 4 shows the states $x_i, i = 1, \cdots, 4$. The weight's norm of critic NN $W_c i(t), i = 1, \cdots, 4$ is presented in Figure 5 and the weight's norm of actor NN $W_a i(t), i = 1, \cdots, 4$ is presented in Figure 6, which it is clear that all weights are bounded and converge to a certain value. The input $u(t)$ is illustrated in Figure 7, which observes that the input converges to the range of $[-5, 5]$. In addition, Figures 8 and 9 illustrate the tracking error $z_1(t)$ as k varies within the range of $[100, 200]$ and i varies within the range of $[15, 30]$, demonstrating the robustness of this control method. In conclusion, it is observed that our control strategy enables the actual output $y(t)$ to track well on the expected trajectory $y_r(t)$ while optimizing the controller energy consumption. In order to better demonstrate the optimization of the energy consumption in this control scheme for a flexible robotic manipulator, we conduct a comparative experiment with the control scheme referenced in [19]. As illustrated in Figures 10 and 11, under conditions of similar tracking effectiveness, the control energy consumption of our scheme is significantly improved compared to that of the scheme in [19].

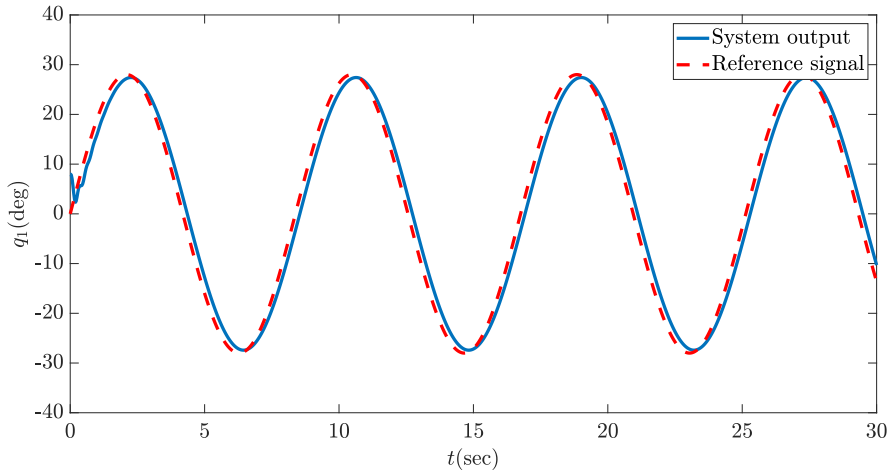

Figure 3. Tracking performance.

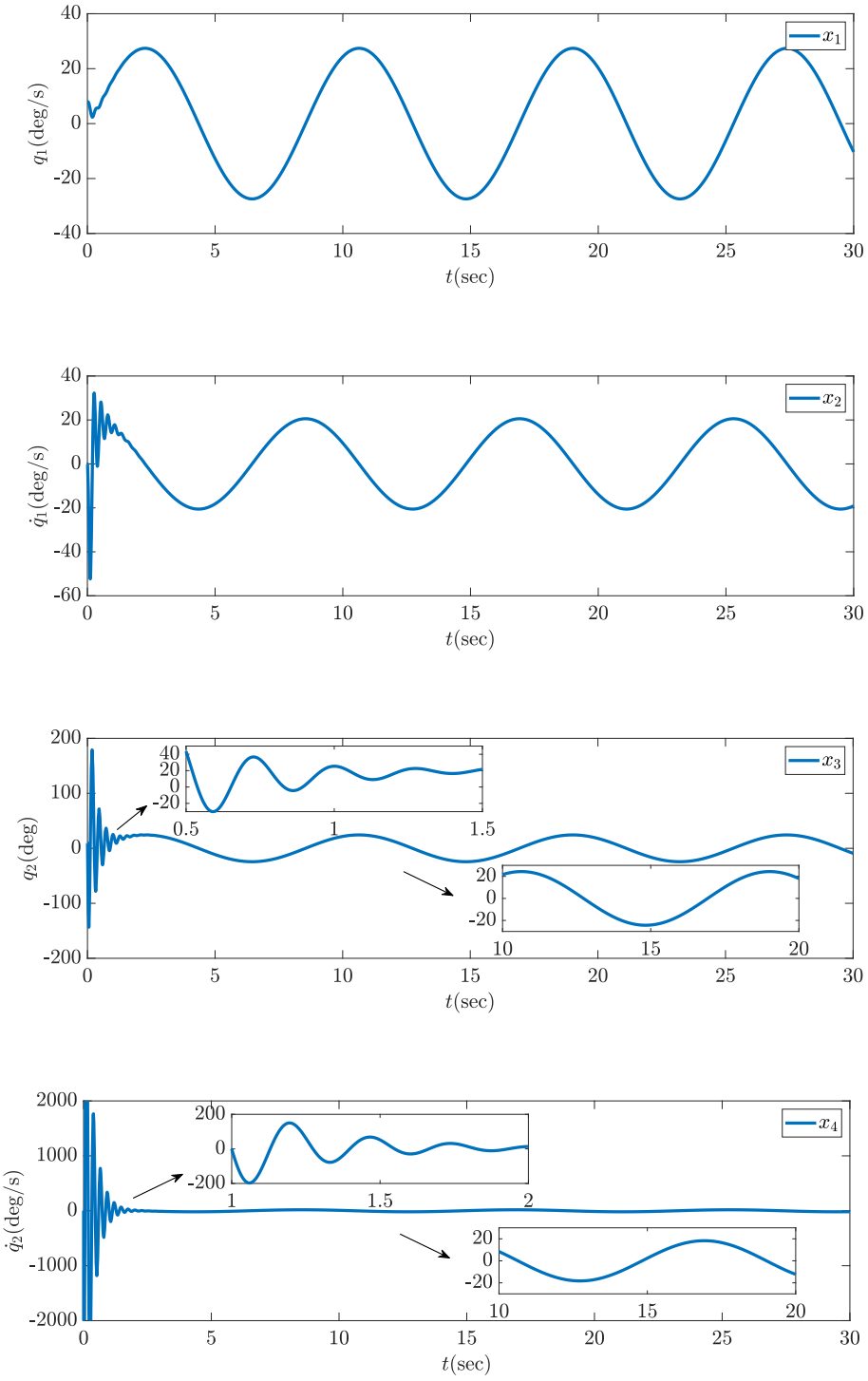

Figure 4. The trajectories of $x_i, i = 2, 3, 4$.

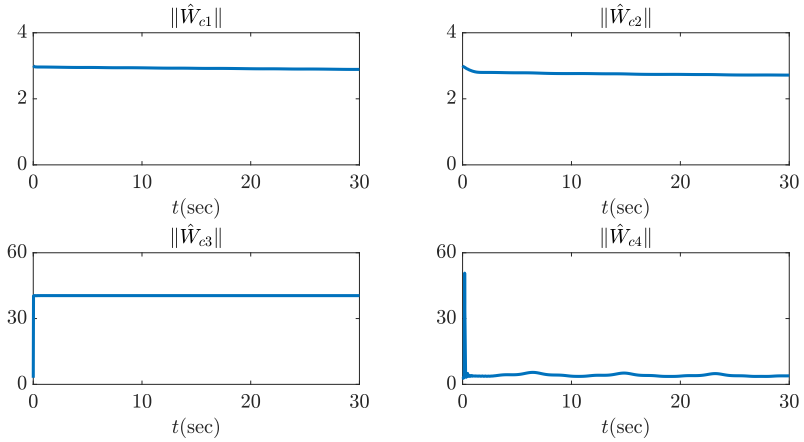

Figure 5. The weight's norm of critic NN in each step.

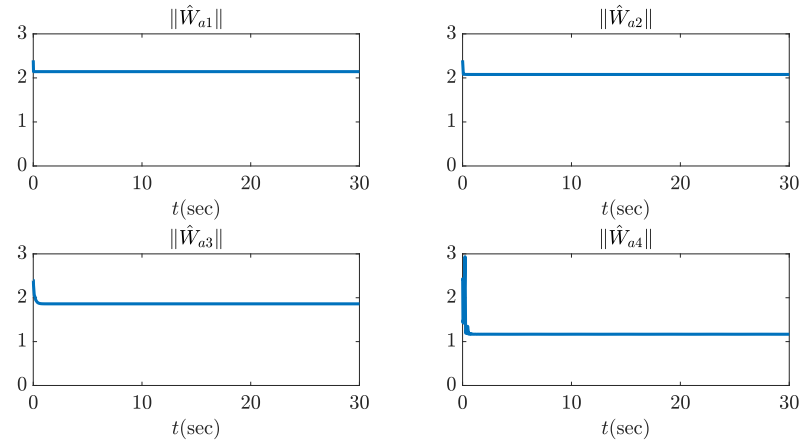

Figure 6. The weight's norm of actor NN in each step.

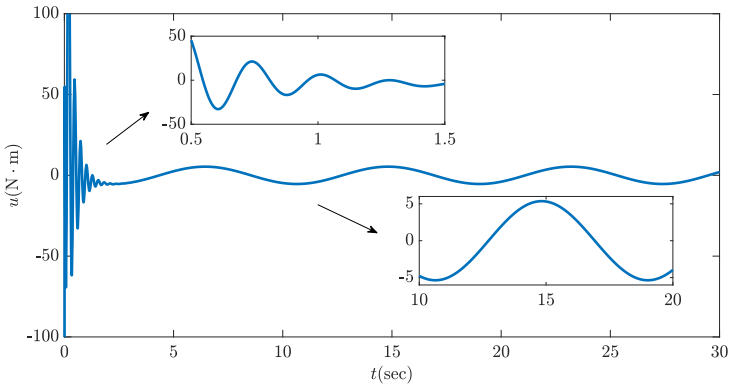

Figure 7. The control input $u(t)$.

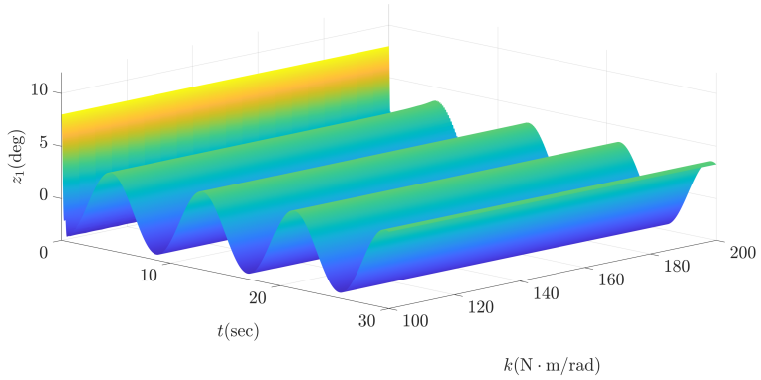

Figure 8. Tracking error $z_1(t)$ in the case of the joint flexible k.

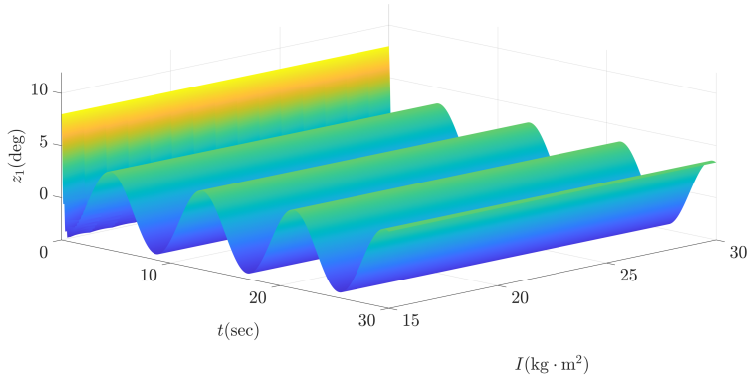

Figure 9. Tracking error $z_1(t)$ in the case of the mass inertia I.

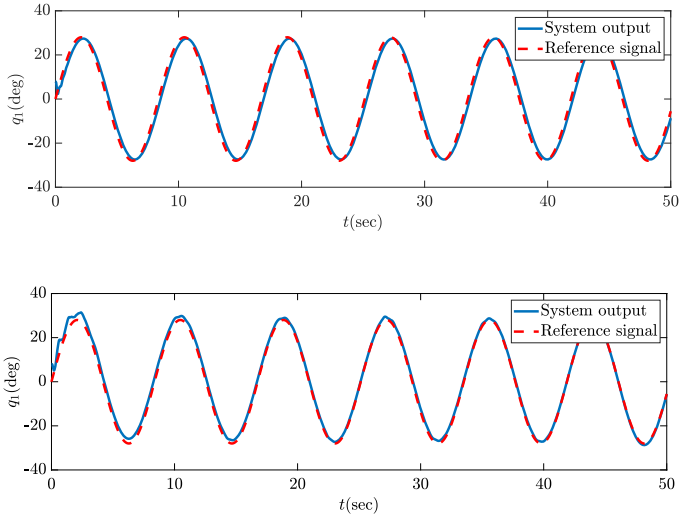

Figure 10. Tracking performance (this paper on the top and [19] on the bottom).

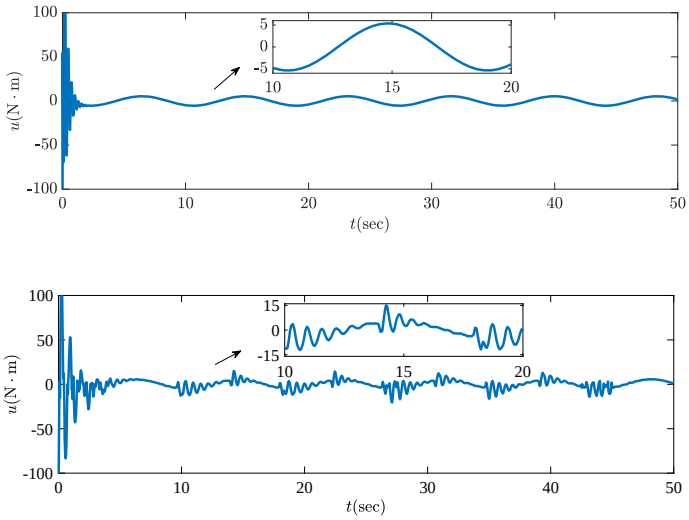

Figure 11. Control input (this paper on the top and [19] on the bottom).

5. Conclusions

In this paper, an optimal backstepping control scheme is proposed for trajectory tracking of a flexible manipulator by integrating optimal control into backstepping control. In this control scheme, each virtual controller, as well as the actual controller, is designed as an optimized solution in the corresponding inverse step. This approach achieves performance optimization for the entire flexible robotic manipulator system. RL is built on a critic-actor architecture, where the critic assesses performance then provides feedback to the actor. The actor then controls the system, and the two NNs collaborate to learn. Since the RL update law is derived from the negative gradient of a simple function, we simplify the design of the controller compared to existing optimal control methods for flexible robotic manipulators. Finally, the effectiveness of the control method for solving the trajectory tracking problem of flexible robotic manipulators is demonstrated through both theoretical analysis and simulation studies.

Author contributions

Huihui Zhong: Methodology, Validation, Writing-original draft; Weijian Wen: Formal analysis, Supervision; Jianjun Fan: Conceptualization, Investigation; Weijun Yang: Writing – Review and Editing, Visualization. All authors have read and approved the final version of the manuscript for publication.

Acknowledgments

This work was supported by the Special projects in key fields of colleges and universities in Guangdong Province, China (No.2024ZDZX1070, No.2024ZDZX3094), and the Guangdong University research and innovation team project (No.2024KCXTD075).

Conflict of interest

All authors declare no conflicts of interest in this paper.

References

1. Z. Li, S. Li, X. Luo, An overview of calibration technology of industrial robots, *IEEE-CAA J. Automatica Sin.*, **8** (2021), 23–36. https://doi.org/10.1109/JAS.2020.1003381
2. M. Kyrarini, F. Lygerakis, A. Rajavenkatanarayanan, C. Sevastopoulos, H. R. Nambiappan, K. K. Chaitanya, et al., A survey of robots in healthcare, *Technologies*, **9** (2021), 8. https://doi.org/10.3390/technologies9010008
3. M. Payal, P. Dixit, T. V. M. Sairam, N. Goyal, Robotics, AI, and the IoT in defense systems, In: *AI and IoT-based intelligent automation in robotics*, Wiley, 2021. https://doi.org/10.1002/9781119711230.ch7
4. Q. Qi, G. Qin, Z. Yang, G. Chen, J. Xu, Z. Lv, et al., Design and motion control of a tendon-driven continuum robot for aerospace applications, *P. I. Mech. Eng. G J. Aer.*, 2024. https://doi.org/10.1177/09544100241263004
5. M. Sostero, Automation and robots in services: Review of data and taxonomy, In: *JRC working papers series on labour, education and technology*, Joint Research Centre, 2020.
6. Q. Yang, X. Du, Z. Wang, Z. Meng, Z. Ma, Q. Zhang, A review of core agricultural robot technologies for crop productions, *Comput. Electron. Agr.*, **206** (2023), 107701. https://doi.org/10.1016/j.compag.2023.107701
7. I. Arocena, A. Huegun-Burgos, I. Rekalde-Rodriguez, Robotics and education: A systematic review, *TEM J.*, **11** (2022), 379–387. https://doi.org/10.18421/TEM111-48
8. C. E. Boudjedir, M. Bouri, D. Boukhetala, An enhanced adaptive time delay control-based integral sliding mode for trajectory tracking of robot manipulators, *IEEE Trans. Control Syst. Technol.*, **31** (2023), 1042–1050. http://dx.doi.org/10.1109/TCST.2022.3208491
9. P. Li, D. Liu, S. Baldi, Adaptive integral sliding mode control in the presence of state-dependent uncertainty, *IEEE-ASME Trans. Mechatron.*, **27** (2022), 3885–3895. http://dx.doi.org/10.1109/TMECH.2022.3145910
10. J. Park, W. Kwon, P. Park, An improved adaptive sliding mode control based on time-delay control for robot manipulators, *IEEE Trans. Ind. Electron.*, **70** (2023), 10363–10373. http://dx.doi.org/10.1109/TIE.2022.3222616
11. H. Ma, H. Ren, Q. Zhou, H. Li, Z. Wang, Observer-based neural control of N-link flexible-joint robots, *IEEE Trans. Neural Netw. Learn. Syst.*, **35** (2024), 5295–5305. https://doi.org/10.1109/TNNLS.2022.3203074
12. Y. Xie, Q. Ma, J. Gu, G. Zhou, Event-triggered fixed-time practical tracking control for flexible-joint robot, *IEEE Trans. Fuzzy Syst.*, **31** (2023), 67–76. https://doi.org/10.1109/TFUZZ.2022.3181463

13. M. M. Arefi, N. Vafamand, B. Homayoun, M. Davoodi, Command filtered backstepping control of constrained flexible joint robotic manipulator, *IET Control Theory Appl.*, **17** (2023), 2506–2518. https://doi.org/10.1049/cth2.12528

14. X. Cheng, Y. J. Zhang, H. S. Liu, D. Wollherr, M. Buss, Adaptive neural backstepping control for flexible-joint robot manipulator with bounded torque inputs, *Neurocomputing*, **458** (2021), 70–86. https://doi.org/10.1016/j.neucom.2021.06.013

15. Y. Zhang, M. Zhang, F. Du, Robust finite-time command-filtered backstepping control for flexible-joint robots with only position measurements, *IEEE Trans. Syst. Man Cybern. Syst.*, **54** (2024), 1263–1275. https://doi.org/10.1109/TSMC.2023.3324761

16. R. Datouo, J. J. B. M. Ahanda, A. Melingui, F. Biya-Motto, B. E. Zobo, Adaptive fuzzy finite-time command-filtered backstepping control of flexible-joint robots, *Robotica*, **39** (2021), 1081–1100. https://doi.org/10.1017/S0263574720000910

17. U. K. Sahu, B. Subudhi, D. Patra, Sampled-data extended state observer-based backstepping control of two-link flexible manipulator, *Trans. Inst. Meas. Control*, **41** (2019), 3581–3599. https://doi.org/10.1177/0142331219832954

18. J. Li, L. Zhu, Practical tracking control under actuator saturation for a class of flexible-joint robotic manipulators driven by DC motors, *Nonlinear Dyn.*, **109** (2022), 2745–2758. https://doi.org/10.1007/s11071-022-07602-4

19. G. Lai, S. Zou, H. Xiao, L. Wang, Z. Liu, K. Chen, Fixed-time adaptive fuzzy control with prescribed tracking performances for flexible-joint manipulators, *J. Franklin Inst.*, **361** (2024), 106809. https://doi.org/10.1016/j.jfranklin.2024.106809

20. R. Bellman, Dynamic programming, *Science*, **153** (1966), 34–37. https://doi.org/10.1126/science.153.3731.34

21. L. S. Pontryagin, *Mathematical theory of optimal processes*, London: Routledge, 2017. https://doi.org/10.1201/9780203749319

22. Y. Yang, H. Modares, K. G. Vamvoudakis, W. He, C. Z. Xu, D. C. Wunsch, Hamiltonian-driven adaptive dynamic programming with approximation errors, *IEEE Trans. Cybern.*, **52** (2022), 13762–13773. https://doi.org/10.1109/TCYB.2021.3108034

23. P. J. Werbos, Neural networks for control and system identification, In: *Proceedings of the 28th IEEE conference on decision and control*, **1** (1989), 260–265. https://doi.org/10.1109/CDC.1989.70114

24. W. T. Miller, R. S. Sutton, P. J. Webros, A menu of designs for reinforcement learning over time, In: *Neural networks for control*, MIT Press, 1995, 67–95.

25. P. J. Webros, Approximate dynamic programming for real-time control and neural modeling, In: *Handbook of intelligent control: Neural fuzzy and adaptive approaches*, New York: Van Nostrand Reinhold, 1992.

26. G. Lai, Y. Zhang, Z. Liu, J. Wang, K. Chen, C. L. P. Chen, Direct adaptive fuzzy control scheme with guaranteed tracking performances for uncertain canonical nonlinear systems, *IEEE Trans. Fuzzy Syst.*, **30** (2022), 818–829. https://doi.org/10.1109/TFUZZ.2021.3049902

27. Y. Wang, Y. Chang, A. F. Alkhateeb, N. D. Alotaibi, Adaptive fuzzy output-feedback tracking control for switched nonstrict-feedback nonlinear systems with prescribed performance, *Circuits Syst. Signal Process.*, **40** (2021), 88–113. https://doi.org/10.1007/s00034-020-01466-y

28. D. Wang, M. Ha, M. Zhao, The intelligent critic framework for advanced optimal control, *Artif. Intell. Rev.*, **55** (2022), 1–22. https://doi.org/10.1007/s10462-021-10118-9

29. D. Li, J. Dong, Fractional-order systems optimal control via actor-critic reinforcement learning and its validation for chaotic MFET, *IEEE Trans. Autom. Sci. Eng.*, 2024, 1–10. https://doi.org/10.1109/TASE.2024.3361213

30. D. Cui, C. K. Ahn, Y. Sun, Z. Xiang, Mode-dependent state observer-based prescribed performance control of switched systems, *IEEE Trans. Circuits Syst. II-Express Briefs*, **71** (2024), 3810–3814. https://doi.org/10.1109/TCSII.2024.3370865

31. H. Jiang, W. Su, B. Niu, H. Wang, J. Zhang, Adaptive neural consensus tracking control of distributed nonlinear multiagent systems with unmodeled dynamics, *Int. J. Robust Nonlinear Control*, **32** (2022), 8999–9016. https://doi.org/10.1002/rnc.6313

32. G. Lai, Y. Zhang, Z. Liu, C. L. P. Chen, Indirect adaptive fuzzy control design with guaranteed tracking error performance for uncertain canonical nonlinear systems, *IEEE Trans. Fuzzy Syst.*, **27** (2019), 1139–1150. https://doi.org/10.1109/TFUZZ.2018.2870574

AIMS Press

© 2024 the Author(s), licensee AIMS Press. This is an open access article distributed under the terms of the Creative Commons Attribution License (https://creativecommons.org/licenses/by/4.0)