



Adaptive practical predefined-time neural tracking control for multi-joint uncertain robotic manipulators with input saturation

Huayang Sai^{1,2} · Zhenbang Xu¹ · Enyang Zhang^{1,2}

Received: 13 December 2022 / Accepted: 28 June 2023 / Published online: 21 July 2023
© The Author(s), under exclusive licence to Springer-Verlag London Ltd., part of Springer Nature 2023

Abstract

Predefined-time stability contributes to constraining the tracking time of robotic manipulators, but the stringent judgment conditions restrict its practical application. This paper studies an adaptive practical predefined-time neural control scheme for uncertain multi-joint robotic manipulators with input saturation. First, a practical predefined-time stability criterion is established to ensure the closed-loop stability of uncertain systems. Then, the unknown robotic dynamic model can be approximated by radial basis function neural networks. Meanwhile, the input saturation of the robotic manipulator is compensated by introducing an adaptive term. Based on the constructed stability criterion, the proposed controller is proved to guarantee that the tracking error of the system converges to a small neighborhood of the origin within a predefined time and is independent of the initial state. Finally, the effectiveness of the proposed control scheme is emphasized by numerical simulations and experiments of a two-joint and a nine-joint robotic manipulator, respectively.

Keywords Predefined-time control · Multi-joint robotic manipulator · Adaptive control · Input saturation · Neural network

1 Introduction

Achieving fast, high-precision tracking control of robotic manipulators has always been a fundamental research goal for mechanical robots. With the rapid development of the computational power, the dynamic model of a robotic manipulator can be approximated by relying on fuzzy/neural adaptive technologies [1–3], which allows researchers to focus more on the study of special properties in robotic manipulator tracking control, such as the convergence time of tracking errors.

In the last decade and more, as one of the key performance indicators for each control system that requires a timely response, the research on the settling time of tracking errors has never stopped. Since Haimo et al. [4] proposed a finite-time controller to replace the traditional method of asymptotic regulation, a large number of finite-time control schemes have emerged to ensure that the tracking error converges to the origin in a finite time [5]. Among them, terminal sliding mode control (TSMC) [6] is a typical finite-time control method, and many relevant control algorithms have been derived from the TSMC, such as the fast TSMC [7] and the nonsingular fast TSMC [8]. However, with the increasing requirements of robotic manipulators in tracking control, the problem of the settling time of finite-time control depending on the initial state of the system has concerned scholars. This problem was not solved until Polyakov [9] proposed a fixed-time stability concept and rigorously proved that the upper bound for the settling time of the tracking error in this framework depends only on the control parameters and is irrelevant to initial states. Subsequently, Zuo [10, 11] presented several significant lemmas in the fixed-time theory, which contributed to the development of fixed-time control and its application in robotic manipulators and

✉ Zhenbang Xu
xuzhenbang@ciomp.ac.cn

Huayang Sai
saihuayang18@mails.ucas.ac.cn

Enyang Zhang
zhangenyanguaa@126.com

¹ CAS Key Laboratory of On-Orbit Manufacturing and Integration for Space Optics System, Changchun Institute of Optics, Fine Mechanics and Physics, Chinese Academy of Sciences, Changchun 130033, China

² University of Chinese Academy of Sciences, Beijing 100049, China

multi-agent systems [12–16]. However, the fixed-time stability criterion is stringent in applications, especially considering controlled objects with unknown system parameters. Therefore, Ba et al. [17] and Wang et al. [18] proposed a practical fixed-time stability criterion to relax the requirement of proving fixed-time stability. The practical fixed-time stability criterion allows the fuzzy logic systems and the neural network (NN) to approximate unknown parameters of the controlled object while ensuring the practical fixed-time stability of the closed-loop system [19, 20].

Despite the previous progress in fixed-time control studies, the relationship between the tuning gains and the upper bound for the settling time is not explicit; hence, finding the control parameters to guarantee a desired maximum settling time is challenging. To address this problem, the concept of predefined-time stability was creatively proposed in [21]. In contrast to the case of finite-time and fixed-time stability, a system that satisfies predefined-time stability allows explicitly setting upper bounds on the convergence time in the design of the controller. As a result, the predefined-time stability concept has the potential to guarantee a strict hierarchy of tasks in trajectory tracking for robotic manipulators and avoid conflicts between decoupled task constraints. As a promising and valuable concept, the notion of predefined-time stability has gained extensive interest in the tracking control of robotic manipulators [22–24]. In particular, for highly flexible and redundant robotic manipulators capable of performing complex tasks in enclosed and unstructured workspaces containing obstacles [25, 26], it is crucial to ensure trajectory tracking errors converge in a pre-specified time for the successful execution of the task. It is worth noting that the dynamic parameters are required to be known a priori in the predefined-time controller, but building an accurate dynamic model of a multi-joint manipulator is not easy. Therefore, most predefined-time control schemes only consider the simulation of two-linked robotic manipulators [22, 23, 27], which is still a substantial challenge for the practical application of multi-joint manipulators. Although the approximation of the dynamic model of a robotic manipulator using NNs has been proposed early in [28], the proof of predefined-time stability requires a more stringent stability criterion compared to fixed-time stability [21, 29, 30], which prevents the combination of predefined-time control with these intelligent control techniques. In recent work, a practical predefined-time criterion was described in [31], but its settling-time bound violated the principle of being determined by only a control parameter. Moreover, a definition of practical prescribed-time stability was given in [32] relying on a segmentation function with exponential, whose judgment condition is complex and has the risk of

exponential explosions. Therefore, up to now, establishing a suitable practical predefined-time stability criterion to deal with unknown systems is still a challenging issue.

To guarantee fast convergence of the tracking error, predefined-time controllers always require large initial control torques [33], which tend to lead to actuator saturation in the tracking control of robotic manipulators. In the framework of finite-time or fixed-time stability, some anti-windup solutions for robotic systems have been proposed in [34–36]. These approaches to deal with the input saturation include ignoring control saturation and adding compensation terms to minimize the adverse effects of input saturation on the system [34], and correcting inconsistencies between the input and controller states due to input saturation [35, 36]. However, to the best of our knowledge, there are no results about the input saturation in predefined-time control schemes due to the difficulty in mathematical analysis despite its importance in robotic manipulators.

Driven by the practical requirements of tracking control for multi-joint robotic manipulators with time constraints and inspired by previous discussions, we investigate a novel adaptive practical predefined-time neural tracking control scheme for uncertain multi-joint robotic manipulators with considering the input saturation. The contributions of this paper can be summarized as follows:

1. Based on the predefined-time convergent algorithm in [29], a novel practical predefined-time stability criterion is constructed, which relaxes the stringent proof conditions for the existence of predefined-time stability, thus paving the way for the application of adaptive techniques and intelligent control algorithms to the predefined-time control theory.

2. Most predefined-time controllers require a large initial control torque, ignoring the actuator saturation in the tracking control of the robotic manipulator [22, 30, 37]. This study constructs an adaptive auxiliary term through the relationship between the control input and the controller state, thus ensuring the practical predefined-time stability of the system with the actuator saturation.

3. Combined with the NN technique, the prior requirement of dynamic parameters is avoided so that the proposed control scheme has more robust applicability in multi-joint robotic manipulators. Most of the existing predefined-time controllers have only been simulated or experimentally investigated on two- or three-joint robotic manipulators [22, 23, 27, 38]. This paper presents the first simulation and experimental study of predefined-time control algorithms on a robotic manipulator with nine joints, and the tracking accuracy of the manipulator end-effector is also investigated and analyzed.

The rest of the paper is organized as follows. Section 2 outlines the problem formulation and provides some

preliminaries. The main work is presented in Sect. 3. To illustrate the effectiveness of the theoretical results, Sects. 4 and 5, respectively, introduce two simulation examples and an experiment conducted on a two-joint and a nine-joint robotic manipulator. In Sect. 6, a comparison between the control scheme in this paper and those in previous studies is presented. Finally, Sect. 7 provides the conclusions drawn from this study.

2 Problem formulation and preliminaries

2.1 Preliminaries

Consider an autonomous dynamical system

$$\dot{x} = f(x; \rho), x(0) = x_0, \tag{1}$$

where $x \in \mathbb{R}^n$ denotes the system state, $\rho \in \mathbb{R}^b$ stands for the system parameters. The function $f : \mathbb{R}^n \rightarrow \mathbb{R}^n$ is non-linear and continuous, and the initial condition of this system is $x_0 = x(0)$.

Definition 1 The origin of system (1) is said to be practically predefined-time stable, if for any initial state x_0 , there is a constant $\delta > 0$ and a predefined settling time T_c , such that $\|x(t)\| \leq \delta$, for all $t > T_c$, and T_c is an arbitrarily selected control constant $T_c := T_c(\rho) > 0$.

Remark 1 As described in [17, 18, 39], the concept of “practically“ in Definition 1 implies that the tracking error converges to the set of residuals of the origin. That is, the origin of (1) is said to be predefined-time stable if any solution $x(t, x_0)$ of system (1) reaches the origin within a predefined time T_c [21].

Definition 2 [40] With a constant $a > 0$, the gamma function is defined as

$$\Gamma(a) = \int_0^\infty t^{a-1} \exp(-t) dt. \tag{2}$$

Lemma 1 [29] Consider a continuous radially unbounded function $V : \mathbb{R}^n \rightarrow \mathbb{R}$ satisfies

$$\dot{V}(x) \leq -\frac{\gamma}{T_c} (\alpha V^p(x) + \beta V^q(x))^k, \tag{3}$$

for $x \in \mathbb{R}^n \setminus \{0\}$ and constants satisfy $\alpha, \beta, p, q, k > 0, kp < 1, kq > 1$. Let ρ be the parameter vector $\rho = [\alpha \ \beta \ p \ q \ k]^T \in \mathbb{R}^5$ of (1). Then, the origin $x = 0$ of system (1) is predefined-time stable with the settling time T_c , and γ is a function of ρ that can be written as

$$\gamma(\rho) = \frac{\Gamma(m_p)\Gamma(m_q)}{\alpha^k \Gamma(k)(q-p)} \left(\frac{\alpha}{\beta}\right)^{m_p}, \tag{4}$$

where $m_p = \frac{1-kp}{q-p}$ and $m_q = \frac{kq-1}{q-p}$ are calculated positive parameters.

Lemma 2 Consider a continuous radially unbounded function $V : \mathbb{R}^n \rightarrow \mathbb{R}$ satisfies

$$\dot{V}(x) \leq -\frac{\bar{\gamma}}{T_c} \bar{\alpha} V^p(x) - \frac{\bar{\gamma}}{T_c} \bar{\beta} V^q(x) + \eta, \tag{5}$$

for $x \in \mathbb{R}^n \setminus \{0\}$ and constants satisfy $\bar{\alpha}, \bar{\beta} > 0, 0 < \phi, p < 1, q > 1, 0 \leq \eta < \infty$. Let ρ be the parameter vector $\rho = [\bar{\alpha} \ \bar{\beta} \ p \ q \ \phi]^T \in \mathbb{R}^5$ of (1), and $\bar{\gamma}$ can be calculated by

$$\bar{\gamma}(\rho) = \frac{\Gamma(m_p)\Gamma(m_q)}{\bar{\alpha}\phi(q-p)} \left(\frac{\bar{\alpha}}{\bar{\beta}}\right)^{m_p}, \tag{6}$$

where $m_p = \frac{1-p}{q-p}$ and $m_q = \frac{q-1}{q-p}$ are calculated positive parameters. Then, the trajectory of system (1) is practically predefined-time stable with the settling time T_c , and the residual set of the solution of system (1) is given by

$$x \in \left\{ V(x) < \min \left\{ \left(\frac{\eta}{1-\phi} \frac{T_c}{\bar{\alpha}\bar{\gamma}} \right)^{\frac{1}{p}}, \left(\frac{\eta}{1-\phi} \frac{T_c}{\bar{\beta}\bar{\gamma}} \right)^{\frac{1}{q}} \right\} \right\}. \tag{7}$$

Proof Considering a constant $0 < \phi < 1$, (5) can be written as

$$\dot{V} \leq -\phi \frac{\bar{\gamma}}{T_c} \bar{\alpha} V^p(x) - (1-\phi) \frac{\bar{\gamma}}{T_c} \bar{\alpha} V^p(x) - \frac{\bar{\gamma}}{T_c} \bar{\beta} V^q(x) + \eta, \tag{8}$$

and

$$\dot{V} \leq -\frac{\bar{\gamma}}{T_c} \bar{\alpha} V^p(x) - \phi \frac{\bar{\gamma}}{T_c} \bar{\beta} V^q(x) - (1-\phi) \frac{\bar{\gamma}}{T_c} \bar{\beta} V^q(x) + \eta. \tag{9}$$

Case 1. If $V \geq \left(\frac{\eta T_c}{\bar{\alpha}\bar{\gamma}(1-\phi)}\right)^{\frac{1}{p}}$ and considering $\bar{\gamma} > 0, 0 < \phi < 1$, it has $\eta - (1-\phi) \frac{\bar{\gamma}}{T_c} \bar{\alpha} V^p \leq 0$. Then, (8) can yield

$$\begin{aligned} \dot{V} &\leq -\frac{\bar{\gamma}}{T_c} \bar{\alpha} \phi V^p(x) - \frac{\bar{\gamma}}{T_c} \bar{\beta} V^q(x) \leq \\ &\quad -\frac{\bar{\gamma}}{T_c} \bar{\alpha} \phi V^p(x) - \frac{\bar{\gamma}}{T_c} \bar{\beta} \phi V^q(x). \end{aligned} \tag{10}$$

Let $\alpha = \bar{\alpha}\phi, \beta = \bar{\beta}\phi$, and (10) can be written as

$$\dot{V} \leq -\frac{\bar{\gamma}}{T_c} \alpha V^p(x) - \frac{\bar{\gamma}}{T_c} \beta V^q(x). \tag{11}$$

If $k = 1, 0 < p < 1, q > 1$ in (4), we can have $\gamma(\rho) = \bar{\gamma}(\rho)$ with $\Gamma(1) = 1$. Therefore, (11) can be written as (3) with $0 < p < 1, q > 1$ in (5). Then, according to Lemma 1, it shows that the solution of system (1) is practically

predefined-time stable with the settling time T_c and converges to the compact set

$$x \in \left\{ V(x) < \left(\frac{\eta T_c}{1 - \phi \bar{\alpha} \bar{\gamma}} \right)^{\frac{1}{p}} \right\}. \tag{12}$$

Case 2. If $V \geq \left(\frac{\eta T_c}{\bar{\beta} \bar{\gamma} (1 - \phi)} \right)^{\frac{1}{q}}$ and considering $\bar{\gamma} > 0, 0 < \phi < 1$, it has $\eta - (1 - \phi) \frac{\bar{\gamma}}{T_c} \bar{\beta} V^q \leq 0$. Then, (9) can yield

$$\begin{aligned} \dot{V} &\leq -\frac{\bar{\gamma}}{T_c} \bar{\alpha} V^p(x) - \frac{\bar{\gamma}}{T_c} \bar{\beta} \phi V^q(x) \leq \\ &-\frac{\bar{\gamma}}{T_c} \bar{\alpha} \phi V^p(x) - \frac{\bar{\gamma}}{T_c} \bar{\beta} \phi V^q(x). \end{aligned} \tag{13}$$

Similarly, let $\alpha = \bar{\alpha} \phi, \beta = \bar{\beta} \phi$ and combine with $\Gamma(1) = 1$, (13) can also be written as (11). Then, it shows that the solution of system (1) is practically predefined-time stable with the settling time T_c and converge to the compact set

$$x \in \left\{ V(x) < \left(\frac{\eta T_c}{1 - \phi \bar{\beta} \bar{\gamma}} \right)^{\frac{1}{q}} \right\}. \tag{14}$$

According to case 1 and case 2, Lemma 2 can be proved. \square

Remark 2 If $\eta = 0$, Lemma 2 can be considered as a special case of $k = 1$ in Lemma 1. It is obtained from (7) that the solution of the system (1) reaches the origin within a predefined time if $\eta = 0$.

Lemma 3 [41] For any $\vartheta \in \mathbb{R}$ and $\varsigma > 0$, it has

$$0 \leq |\vartheta| - \frac{\vartheta^2}{\sqrt{\vartheta^2 + \varsigma^2}} < \varsigma. \tag{15}$$

Lemma 4 [11] For any $\xi_1, \xi_2, \dots, \xi_n \geq 0$, it has

$$\sum_{i=1}^n \xi_i^r \geq \left(\sum_{i=1}^n \xi_i \right)^r, \text{ for } 0 < r \leq 1, \tag{16}$$

$$\sum_{i=1}^n \xi_i^r \geq n^{1-r} \left(\sum_{i=1}^n \xi_i \right)^r, \text{ for } 1 < r < \infty. \tag{17}$$

Lemma 5 [42] For any real variables x, y and the constant $l_i (i = 1, 2, 3)$, it has

$$|x|^{l_1} |y|^{l_2} \leq \frac{l_1}{l_1 + l_2} l_3 |x|^{l_1+l_2} + \frac{l_2}{l_1 + l_2} l_3^{-\frac{l_1}{l_2}} |y|^{l_1+l_2}. \tag{18}$$

2.2 Dynamic model

Consider a general n -degree of freedom (DOF) rigid robotic manipulator, whose dynamic model can be written as [43]

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + \tau_d = \tau, \tag{19}$$

where q, \dot{q}, \ddot{q} denote the position, velocity, and acceleration vector of the robotic manipulator, respectively. $M(q) \in \mathbb{R}^{n \times n}$ is the symmetric and positive-definite matrix, $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$ is the centrifugal-Coriolis matrix, and $G(q) \in \mathbb{R}^n$ is the Cartesian gravitational. $\tau \in \mathbb{R}^n$ is the vector of control inputs, and $\tau_d \in \mathbb{R}^n$ is a vector of unknown but bounded disturbances. For ease of presentation, M, C, G are used in this paper to denote $M(q), C(q, \dot{q}), G(q)$, respectively.

Assumption 1 The vector of external disturbances τ_d is unknown but bounded and satisfies $|\tau_{di}| \leq \bar{d}_i (i = 1, 2, \dots, n)$, where \bar{d}_i is an unknown positive constant.

Assumption 2 The desired trajectory $q_d(t)$ and its derivatives $q_d^{(i)}(t) (i = 1, 2)$ are known and bounded.

The external disturbance in robotic systems is primarily caused by frictional forces, which are usually bounded in real-world systems. As such, it is reasonable to assume that the external disturbance term $|\tau_{di}|$ is also bounded. Additionally, a known and bounded desired trajectory and velocity serve as the foundation for tracking control, as an unreasonable tracking trajectory would result from a lack of such information. These assumptions are commonly employed in dynamic control of robotic manipulators, as seen in previous works such as [19, 44, 45].

Considering the actuator saturation caused by the large torque of the predefined-time controller, the designed control input $\tau(t) \in \mathbb{R}^n$ is affected by the saturation non-linearity and can be expressed as

$$u(\tau_i(t)) = \begin{cases} \text{sign}(\tau_i(t))\tau_{\max i}, & |\tau_i(t)| \geq \tau_{\max i} \\ \tau_i(t), & |\tau_i(t)| < \tau_{\max i} \end{cases} \tag{20}$$

where $\tau_{\max i}$ represents the maximum control torque allowed for joint i .

2.3 Neural network

Applying the powerful concept of predefined-time stability requires an exact knowledge of the controlled system. However, dynamic parameters in (19) are often challenging to measure accurately, especially for multi-joint robotic systems. To address this concern, a brief review of the radial basis function NN approximation of the unknown continuous function is considered.

The following radial basis function NN is utilized to approximate the unknown continuous function: $h_{nn} = W^T \Psi(Z)$, where $Z \in \Omega_Z \subset \mathbb{R}^l$ is the input vector; $W = [w_1, w_2, \dots, w_b]^T \in \mathbb{R}^b$ is the weight vector with the node number $b > 1$. $\Psi(Z) = [\psi_1(Z), \psi_2(Z), \dots, \psi_b(Z)]^T \in \mathbb{R}^b$ is the basis function vector and $\psi_i(Z)$ is chosen as $\psi_i(Z) = \exp(-(Z - \theta_i)^T(Z - \theta_i)/\varpi_i^2), i = 1, 2, \dots, b$, where $\theta_i = [\theta_{i1}, \dots, \theta_{il}]^T$ is the center of the receptive field and ϖ_i is the width of the Gaussian function.

Lemma 6 [46] *For an arbitrary continuous function $h(Z)$ defined on a compact set $\Omega_Z \subset \mathbb{R}^l$, there exists NN such that*

$$h(Z) = W^{*T} \Psi(Z) + \Delta(Z), \forall Z \in \Omega_Z \subset \mathbb{R}^l, \tag{21}$$

where W^* is the ideal weight vector and $\Delta(Z)$ is the approximation error. The ideal weight vector W^* is an artificial quantity required for analytical purposes which is defined as $W^* := \arg \min_{W \in \mathbb{R}^b} \left\{ \sup_{Z \in \Omega_Z} |h(Z) - W^T \Psi(Z)| \right\}$, such that $|\Delta(Z)| \leq \delta$, and $\delta > 0$.

This paper aims to establish a control signal τ for each joint of an uncertain multi-joint robotic system under actuator saturation so that the tracking error of the robotic joints can converge to a small neighborhood of the origin within a predefined time.

3 Controller design and stability analysis

In the design of the predefined-time controller, the dynamic parameters of the robotic manipulator are unknown, and the actuator saturation is considered. Then, it is rigorously demonstrated that the proposed controller can be incorporated within the framework of the practical predefined-time stability.

3.1 Predefined-time controller design

Before the controller design, we consider one of the joints of the robotic manipulator. According to (19), for a robotic manipulator with n joints, the control torque of the i th joint can be expressed as

$$\begin{aligned} \tau_i = & M_{i1}\ddot{q}_1 + \dots + M_{ii}\ddot{q}_i + \dots + M_{in}\ddot{q}_n \\ & + C_{i1}\dot{q}_1 + \dots + C_{ii}\dot{q}_i + \dots \\ & + C_{in}\dot{q}_n + G_i + \tau_{di}, \end{aligned} \tag{22}$$

where M_{i1}, M_{ii}, M_{in} and C_{i1}, C_{ii}, C_{in} denote the first, i th and n th elements of the i th row of matrices M and C ; G_i and τ_{di}

denote the i th elements of vectors G and τ_d ; $\ddot{q}_i, \dot{q}_i, \tau_i$ denote the acceleration, velocity and torque of joint i .

Then, the acceleration of the joint i can be obtained from (22) as

$$\begin{aligned} \ddot{q}_i = & -M_{ii}^{-1}(M_{i1}\ddot{q}_1 \\ & + \dots + M_{ii-1}\ddot{q}_{i-1} + M_{ii+1}\ddot{q}_{i+1} + \dots + M_{in}\ddot{q}_n \\ & + C_{i1}\dot{q}_1 + \dots + C_{ii}\dot{q}_i + \dots \\ & + C_{in}\dot{q}_n + G_i) + M_{ii}^{-1}\tau_i - M_{ii}^{-1}\tau_{di}. \end{aligned} \tag{23}$$

For convenience, and considering input saturation, the inverse dynamic equation for the i th joint of the robotic manipulator can be expressed as

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f(x) + b(x)u(\tau) + d \end{cases} \tag{24}$$

where $x_1, x_2, \tau \in \mathbb{R}$ denote q_i, \dot{q}_i, τ_i ; $f(x) = -M_{ii}^{-1}(M_{i1}\ddot{q}_1 + \dots + M_{ii-1}\ddot{q}_{i-1} + M_{ii+1}\ddot{q}_{i+1} + \dots + M_{in}\ddot{q}_n + C_{i1}\dot{q}_1 + \dots + C_{ii}\dot{q}_i + \dots + C_{in}\dot{q}_n + G_i), b(x) = M_{ii}^{-1}$, and $d = -M_{ii}^{-1}\tau_{di}$. Combining with Assumption 1, we reasonably assume that d satisfies $|d| \leq \bar{d}$, where \bar{d} is an unknown positive constant. Then, the controller for the i th joint will be developed in (25-37).

First, define the position tracking error as

$$s_1 = x_1 - y_d, \tag{25}$$

where y_d denotes the desired trajectory of joint i . Then, an intermediate variable ξ_1 is designed as

$$\xi_1 = -\frac{s_1 \bar{\xi}_1^2}{\sqrt{s_1^2 \bar{\xi}_1^2 + l_1^2}}, \tag{26}$$

where l_1 is a small constant satisfying $l_1 > 0$. $\bar{\xi}_1$ is an intermediate variable, which can be expressed as

$$\bar{\xi}_1 = \frac{\gamma}{T_c} k_1 |s_1|^{\frac{1}{2}} \text{sign}(s_1) + \frac{\gamma}{T_c} c_1 s_1^3, \tag{27}$$

where k_1 and c_1 are two positive gain constants.

Combined with Lemma 2, the predefined-time parameter γ is defined as

$$\gamma = \frac{\Gamma(m_p)\Gamma(m_q)}{\alpha\phi(q-p)} \left(\frac{\alpha}{\beta}\right)^{m_p}, \tag{28}$$

where p, q, ϕ are defined positive constants, which satisfy $0 < p, \phi < 1, q > 1$. Meanwhile, m_p, m_q, α, β can be derived from $m_p = \frac{1-p}{q-p}, m_q = \frac{q-1}{q-p}, \alpha = \min\{2^{\frac{3}{2}}k_{\min}, r_1\}$,

$$\beta = \min\left\{2c_{\min}, 2r_2 - \frac{9}{2}\rho^{\frac{4}{3}}r_2\right\}, \tag{29}$$

where

$0 < \rho < (\frac{2}{3})^{\frac{3}{2}}$, $r_1, r_2 > 0$, and k_{\min} and c_{\min} will be defined later.

Then, the radial basis function is used to approximate the following function

$$F(Z) = f(x) + [b(x) - 1]u(\tau) - \dot{\xi}_1 = W^{*T}\Psi(Z) + \Delta(Z), \tag{29}$$

where $Z = [x'_1, x'_2, y'_d, \dot{y}'_d, \ddot{y}'_d]^T$. Since the dynamic model of the robotic manipulator depends on the states of all joints, $x'_1 \in \mathbb{R}^n$ denotes a row vector with respect to all joint positions, i.e., $x'_1 = [q_1, q_2, \dots, q_n]$. Similarly, $x'_2, y'_d, \dot{y}'_d, \ddot{y}'_d \in \mathbb{R}^n$ denote the vectors of all joint velocities, desired positions, desired velocities and desired accelerations, respectively. According to the description in Sect. 2.3, we have $|\Delta(Z)| \leq \delta$, and $\delta > 0$. Additionally, since the ideal weight W^* is unknown, σ is utilized to denote $\sigma = \|W^*\|^2$. Then, using $\hat{\sigma}$ to represent the estimate of σ , and it is constructed by the following adaptive law

$$\dot{\hat{\sigma}} = \lambda \frac{s_2^2 \Psi^T \Psi}{2a^2} - \frac{\gamma}{T_c} r_1 \hat{\sigma} - \frac{\gamma}{\lambda T_c} r_2 \hat{\sigma}^3, \tag{30}$$

where $\lambda > 0, a > 0$, r_1, r_2 are positive parameters defined in (28), and s_2 denotes an error variable that will be given in (35).

Furthermore, considering the input saturation, the saturated control torque can be approximated by

$$u(\tau) = g(\tau) + \zeta(\tau), \tag{31}$$

where $g(\tau)$ denotes a smooth function used to approximate the saturated torque, and $\zeta(\tau)$ is the bounded approximation error. Referring to [47], $g(\tau)$ can be chosen as

$$g(\tau) = \tau_{\max} \times \tanh\left(\frac{\tau}{\tau_{\max}}\right) = \tau_{\max} \frac{e^{\tau/\tau_{\max}} - e^{-\tau/\tau_{\max}}}{e^{\tau/\tau_{\max}} + e^{-\tau/\tau_{\max}}}, \tag{32}$$

and the bounded approximation error $\zeta(\tau)$ satisfying

$$|\zeta(\tau)| = |u(\tau) - g(\tau)| \leq \tau_{\max}(1 - \tanh(1)) = \bar{\Delta}. \tag{33}$$

To compensate for the saturated controller, an auxiliary variable η is calculated by the following adaptive law

$$\dot{\eta} = -\eta + (g(\tau) - \tau). \tag{34}$$

Then, based on the designed intermediate variable ξ_1 and auxiliary variable η , the error variable s_2 can be defined as

$$s_2 = x_2 - \xi_1 - \eta. \tag{35}$$

Finally, the input torque is calculated by

$$\tau = -\frac{s_2 \bar{\xi}_2^2}{\sqrt{s_2^2 \bar{\xi}_2^2 + l_2^2}}, \tag{36}$$

where $l_2 > 0$. The intermediate variable $\bar{\xi}_2$ is expressed as

$$\begin{aligned} \bar{\xi}_2 = & \frac{\gamma}{T_c} k_2 |s_2|^{\frac{1}{2}} \text{sign}(s_2) \\ & + \frac{\gamma}{T_c} c_2 s_2^3 + \frac{3}{2} s_2 + s_1 + \eta + \frac{s_1(\eta - \dot{y}_d)}{s_2} \\ & + \frac{\hat{\sigma} s_2 \Psi^T \Psi}{2a^2}, \end{aligned} \tag{37}$$

where k_2 and c_2 are positive gain constants. Then, with k_1, c_1 defined in (27), k_{\min} and c_{\min} in (28) can be denoted as $k_{\min} = \min\{k_1, k_2\}$ and $c_{\min} = \min\{c_1, c_2\}$.

Based on the design of the controller mentioned above, the block diagram on the implementation of the proposed control scheme can be shown in Fig. 1. Different from the existing robotic controllers, the proposed control scheme can provide independent control torque for each joint of the robotic manipulator, which avoids complex matrix operations. Meanwhile, for multi-joint robotic manipulators, only several joints of the manipulator are required to be controlled in some tasks; thus, the proposed control scheme contributes to reducing the computational burden of the controller.

3.2 Stability analysis

The conclusion of the proposed controller regarding convergence analysis is summarized in Theorem 1.

Theorem 1 *For any joint of a multi-joint uncertain robotic manipulator, the position tracking error of the joint can converge to a small neighborhood near the origin within the predefined time T_c based on the control scheme in (25–37).*

Proof Consider a Lyapunov function candidate as follows:

$$V = \frac{1}{2} s_1^2 + \frac{1}{2} s_2^2 + \frac{1}{2\lambda} \tilde{\sigma}^2, \tag{38}$$

where $\tilde{\sigma} = \sigma - \hat{\sigma}$ denotes the estimation error of σ . Take the derivative of V with respect to time yields

$$\dot{V} = s_1 \dot{s}_1 + s_2 \dot{s}_2 - \frac{1}{\lambda} \tilde{\sigma} \dot{\tilde{\sigma}}. \tag{39}$$

According to (24), (25) and (35), we have

$$\dot{s}_1 = \dot{x}_1 - \dot{y}_d = s_2 + \xi_1 + \eta - \dot{y}_d. \tag{40}$$

Then, combined with (34), \dot{s}_2 can be written as

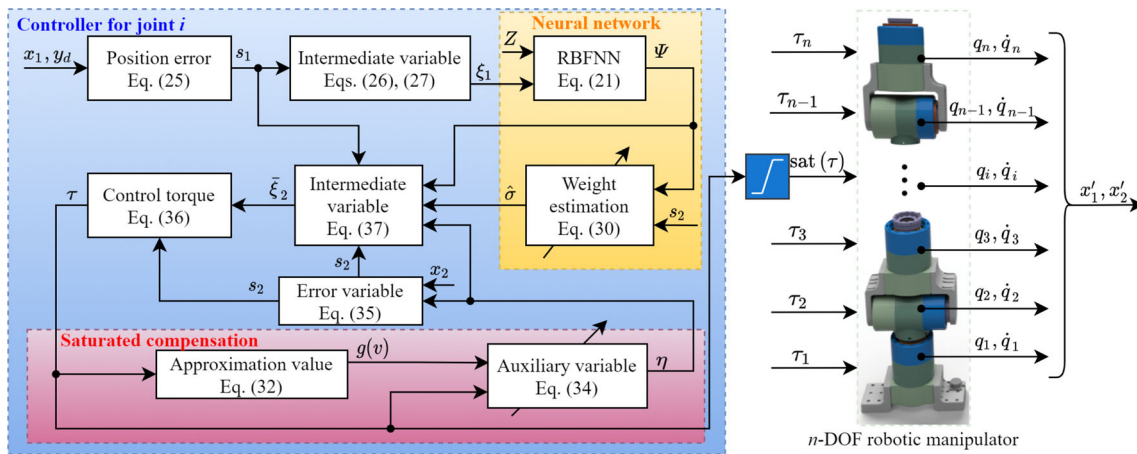


Fig. 1 Control flowchart of the proposed control scheme

$$\begin{aligned} \dot{s}_2 &= \dot{x}_2 - \dot{\xi}_1 - \dot{\eta} = f(x) + b(x)u(\tau) \\ &+ d - \dot{\xi}_1 + \eta - g(\tau) + \tau. \end{aligned} \tag{41}$$

By substituting (29), (40) and (41) into (38), one has

$$\begin{aligned} \dot{V} &= s_1(s_2 + \xi_1 + \eta - \dot{y}_d) - \frac{1}{\lambda} \tilde{\sigma} \dot{\hat{\sigma}} \\ &+ s_2 \left(f(x) + b(x)u(\tau) + d - \dot{\xi}_1 + \eta - g(\tau) + \tau \right) \\ &= s_1 s_2 + s_1 \xi_1 + s_1 \eta - s_1 \dot{y}_d + s_2 F(Z) \\ &+ s_2 (u(\tau) + d + \eta - g(\tau) + \tau) - \frac{1}{\lambda} \tilde{\sigma} \dot{\hat{\sigma}}. \end{aligned} \tag{42}$$

Based on (26), (27) and combined with Lemma 3, it has

$$\begin{aligned} s_1 \xi_1 &= -\frac{s_1^2 \bar{\xi}_1^2}{\sqrt{s_1^2 \bar{\xi}_1^2 + l_1^2}} \leq l_1 - s_1 \bar{\xi}_1 = l_1 - \\ &s_1 \left(\frac{\gamma}{T_c} k_1 |s_1|^{\frac{1}{2}} \text{sign}(s_1) + \frac{\gamma}{T_c} c_1 s_1^3 \right) \\ &= l_1 - \frac{\gamma}{T_c} k_1 |s_1|^{\frac{3}{2}} - \frac{\gamma}{T_c} c_1 s_1^4. \end{aligned} \tag{43}$$

Similarly, based on (36) and Lemma 3, it has

$$s_2 \tau = -\frac{s_2^2 \bar{\xi}_2^2}{\sqrt{s_2^2 \bar{\xi}_2^2 + l_2^2}} \leq l_2 - s_2 \bar{\xi}_2. \tag{44}$$

With the aid of Young’s inequality, and considering (29), $|A(Z)| \leq \delta, \sigma = \|W^*\|^2, a > 0, s_2 F(Z)$ can yield

$$\begin{aligned} s_2 F(Z) &= s_2 W^{*T} \Psi + s_2 A(Z) \leq \frac{s_2 W^{*T} \Psi}{a} a + |s_2| \delta \\ &\leq \frac{\sigma s_2^2 \Psi^T \Psi}{2a^2} + \frac{1}{2} a^2 + \frac{1}{2} s_2^2 + \frac{1}{2} \delta^2. \end{aligned} \tag{45}$$

Next, with (43) and (45), (42) can lead to

$$\begin{aligned} \dot{V} &\leq s_1 s_2 + l_1 - \frac{\gamma}{T_c} k_1 |s_1|^{\frac{3}{2}} - \frac{\gamma}{T_c} c_1 s_1^4 + s_1 \eta - s_1 \dot{y}_d + \frac{\sigma s_2^2 \Psi^T \Psi}{2a^2} \\ &+ \frac{1}{2} a^2 + \frac{1}{2} s_2^2 \\ &+ \frac{1}{2} \delta^2 + s_2 (u(\tau) + d + \eta - g(\tau) + \tau) - \frac{1}{\lambda} \tilde{\sigma} \dot{\hat{\sigma}}. \end{aligned} \tag{46}$$

Combining (31), (44), and considering that $|d| \leq \bar{d}, |\zeta(\tau)| \leq \bar{A}$, we have

$$\begin{aligned} s_2 (u(\tau) + d + \eta - g(\tau) + \tau) &= s_2 (\zeta(\tau) + d + \eta + \tau) \\ &\leq s_2^2 + \frac{1}{2} (\bar{A}^2 + \bar{d}^2) + s_2 \tau + s_2 \eta \\ &\leq s_2^2 + \frac{1}{2} (\bar{A}^2 + \bar{d}^2) + l_2 - s_2 \bar{\xi}_2 + s_2 \eta. \end{aligned} \tag{47}$$

Then, substituting (37) and (47) into (46) leads to

$$\begin{aligned} \dot{V} \leq & s_1 s_2 + l_1 - \frac{\gamma}{T_c} k_1 |s_1|^{\frac{3}{2}} \\ & - \frac{\gamma}{T_c} c_1 s_1^4 + s_1 \eta - s_1 \dot{y}_d + \frac{1}{2} s_2^2 + \frac{1}{2} \delta^2 + \frac{\sigma s_2^2 \Psi^T \Psi}{2a^2} \\ & + \frac{1}{2} a^2 + \frac{1}{2} (\bar{d}^2 + d^2) + s_2^2 + l_2 - \frac{\gamma}{T_c} k_2 |s_2|^{\frac{3}{2}} \\ & - \frac{\gamma}{T_c} c_2 s_2^4 - \frac{3}{2} s_2^2 - s_1 s_2 \\ & - s_2 \eta - s_1 (\eta - \dot{y}_d) - \frac{\hat{\sigma} s_2^2 \Psi^T \Psi}{2a^2} + s_2 \eta \\ & - \frac{1}{\lambda} \tilde{\sigma} \dot{\hat{\sigma}} \\ = & - \frac{\gamma}{T_c} k_1 |s_1|^{\frac{3}{2}} - \frac{\gamma}{T_c} c_1 s_1^4 - \frac{\gamma}{T_c} k_2 |s_2|^{\frac{3}{2}} - \frac{\gamma}{T_c} c_2 s_2^4 \\ & + \frac{\tilde{\sigma} s_2^2 \Psi^T \Psi}{2a^2} - \frac{1}{\lambda} \tilde{\sigma} \dot{\hat{\sigma}} + A_1, \end{aligned} \tag{48}$$

where $A_1 = l_1 + l_2 + \frac{1}{2} (a^2 + \bar{d}^2 + d^2 + \delta^2)$.

The next step is to simplify the expression of \dot{V} in (48). With the adaptive law in (30), one has

$$\begin{aligned} \frac{1}{\lambda} \tilde{\sigma} \dot{\hat{\sigma}} = & \frac{\tilde{\sigma} s_2^2 \Psi^T \Psi}{2a^2} - \frac{\gamma}{T_c} \frac{r_1}{\lambda} \tilde{\sigma} \hat{\sigma} \\ & - \frac{\gamma}{T_c} \frac{r_2}{\lambda^2} \tilde{\sigma} \hat{\sigma}^3. \end{aligned} \tag{49}$$

Substituting (49) into (48) can lead to

$$\begin{aligned} \dot{V} \leq & \frac{\gamma}{T_c} (-k_1 |s_1|^{\frac{3}{2}} - c_1 s_1^4 - k_2 |s_2|^{\frac{3}{2}} - c_2 s_2^4 + \frac{r_1}{\lambda} \tilde{\sigma} \hat{\sigma} \\ & + \frac{r_2}{\lambda^2} \tilde{\sigma} \hat{\sigma}^3) + A_1. \end{aligned} \tag{50}$$

Considering $k_{\min} = \min\{k_1, k_2\}$, $c_{\min} = \min\{c_1, c_2\}$, (50) can be written as

$$\begin{aligned} \dot{V} \leq & -2^{\frac{3}{4}} \frac{\gamma}{T_c} k_{\min} \left\{ \left(\frac{1}{2} s_1^2\right)^{\frac{3}{4}} + \left(\frac{1}{2} s_2^2\right)^{\frac{3}{4}} \right\} \\ & - \frac{4\gamma}{T_c} c_{\min} \left\{ \left(\frac{1}{2} s_1^2\right)^2 \right. \\ & \left. + \left(\frac{1}{2} s_2^2\right)^2 \right\} \\ & + \frac{\gamma}{T_c} \frac{r_1}{\lambda} \tilde{\sigma} \hat{\sigma} \\ & + \frac{\gamma}{T_c} \frac{r_2}{\lambda^2} \tilde{\sigma} \hat{\sigma}^3 + A_1. \end{aligned} \tag{51}$$

Based on $\tilde{\sigma} = \sigma - \hat{\sigma}$, we have

$$\frac{\gamma}{T_c} \frac{r_1}{\lambda} \tilde{\sigma} \hat{\sigma} \leq -\frac{\gamma}{2T_c} \frac{r_1}{\lambda} \tilde{\sigma}^2 + \frac{\gamma}{2T_c} \frac{r_1}{\lambda} \sigma^2, \tag{52}$$

$$\begin{aligned} \frac{\gamma}{T_c} \frac{r_2}{\lambda^2} \tilde{\sigma} \hat{\sigma}^3 \leq & \frac{\gamma}{T_c} \frac{r_2}{\lambda^2} \tilde{\sigma} \\ & (\sigma^3 - 3\sigma^2 \tilde{\sigma} + 3\sigma \tilde{\sigma}^2 - \tilde{\sigma}^3). \end{aligned} \tag{53}$$

Then, substituting (52) and (53) into (51) yields

$$\begin{aligned} \dot{V} \leq & -2^{\frac{3}{4}} \frac{\gamma}{T_c} k_{\min} \left\{ \left(\frac{1}{2} s_1^2\right)^{\frac{3}{4}} + \left(\frac{1}{2} s_2^2\right)^{\frac{3}{4}} \right\} \\ & - \frac{4\gamma}{T_c} c_{\min} \left\{ \left(\frac{1}{2} s_1^2\right)^2 + \left(\frac{1}{2} s_2^2\right)^2 \right\} \\ & - \frac{\gamma}{2T_c} \frac{r_1}{\lambda} \tilde{\sigma}^2 + \frac{\gamma}{2T_c} \frac{r_1}{\lambda} \sigma^2 + \frac{\gamma}{T_c} \frac{r_2}{\lambda^2} \tilde{\sigma} \sigma^3 \\ & - \frac{3\gamma}{T_c} \frac{r_2}{\lambda^2} \tilde{\sigma}^2 \sigma^2 + \frac{3\gamma}{T_c} \frac{r_2}{\lambda^2} \tilde{\sigma}^3 \sigma \\ & - \frac{\gamma}{T_c} \frac{r_2}{\lambda^2} \tilde{\sigma}^4 - \frac{\gamma}{T_c} r_1 \left(\frac{1}{2\lambda} \tilde{\sigma}^2\right)^{\frac{3}{4}} \\ & + \frac{\gamma}{T_c} r_1 \left(\frac{1}{2\lambda} \tilde{\sigma}^2\right)^{\frac{3}{4}} + A_1. \end{aligned} \tag{54}$$

Using Lemma 5, the following inequality can be derived as

$$\frac{\gamma}{T_c} r_1 \left(\frac{1}{2\lambda} \tilde{\sigma}^2\right)^{\frac{3}{4}} \leq \frac{\gamma}{2T_c} \frac{r_1}{\lambda} \tilde{\sigma}^2 + \frac{\gamma}{4T_c} r_1 \left(\frac{3}{4}\right)^3, \tag{55}$$

$$\frac{\gamma}{T_c} \frac{r_2}{\lambda^2} \tilde{\sigma} \sigma^3 \leq \frac{3\gamma}{T_c} \frac{r_2}{\lambda^2} \tilde{\sigma}^2 \sigma^2 + \frac{\gamma}{12T_c} \frac{r_2}{\lambda^2} \sigma^4, \tag{56}$$

$$\frac{3\gamma}{T_c} \frac{r_2}{\lambda^2} \tilde{\sigma} \sigma^3 \leq \frac{9\gamma}{4T_c} \frac{\rho^{\frac{4}{3}} r_2}{\lambda^2} \tilde{\sigma}^4 + \frac{3\gamma}{4T_c} \frac{r_2}{\lambda^2} \rho^4 \sigma^4, \tag{57}$$

where ρ is a defined constant satisfying $0 < \rho < \left(\frac{2}{3}\right)^{\frac{3}{2}}$.

Combining with (55)–(57), (54) can be further written as

$$\begin{aligned} \dot{V} \leq & -2^{\frac{3}{4}} \frac{\gamma}{T_c} k_{\min} \left\{ \left(\frac{1}{2} s_1^2\right)^{\frac{3}{4}} + \left(\frac{1}{2} s_2^2\right)^{\frac{3}{4}} \right\} \\ & - \frac{\gamma}{T_c} r_1 \left(\frac{1}{2\lambda} \tilde{\sigma}^2\right)^{\frac{3}{4}} \\ & - \frac{4\gamma}{T_c} c_{\min} \left\{ \left(\frac{1}{2} s_1^2\right)^2 + \left(\frac{1}{2} s_2^2\right)^2 \right\} \\ & - \frac{\gamma}{T_c} (4r_2 - 9\rho^{\frac{4}{3}} r_2) \left(\frac{1}{2\lambda} \tilde{\sigma}^2\right)^2 \\ & + \frac{\gamma}{2T_c} \frac{r_1}{\lambda} \sigma^2 + \frac{\gamma}{4T_c} r_1 \left(\frac{3}{4}\right)^3 + \frac{\gamma}{12T_c} \frac{r_2}{\lambda^2} \sigma^4 \\ & + \frac{3\gamma}{4T_c} \frac{r_2}{\lambda^2} \rho^4 \sigma^4 + A_1. \end{aligned} \tag{58}$$

Let

$$A_2 = \frac{\gamma}{2T_c} \frac{r_1}{\lambda} \sigma^2 + \frac{\gamma}{4T_c} r_1 \left(\frac{3}{4}\right)^3 + \frac{\gamma}{12T_c} \frac{r_2}{\lambda^2} \sigma^4 + \frac{3\gamma}{4T_c} \frac{r_2}{\lambda^2 \rho^4} \sigma^4 + A_1$$

and considering $\alpha = \min\{2^{\frac{3}{4}}k_{\min}, r_1\}$, $\beta = \min\{2c_{\min}, 2r_2 - \frac{9}{2}\rho^{\frac{4}{3}}r_2\}$ in (28), (58) can be simplified as

$$\dot{V} \leq -\frac{\gamma}{T_c} \alpha \left[\left(\frac{1}{2}s_1^2 + \frac{1}{2}s_2^2\right)^{\frac{3}{4}} + \left(\frac{1}{2\lambda}\tilde{\sigma}^2\right)^{\frac{3}{4}} \right] - \frac{2\gamma}{T_c} \beta \left[\left(\frac{1}{2}s_1^2 + \frac{1}{2}s_2^2\right)^2 + \left(\frac{1}{2\lambda}\tilde{\sigma}^2\right)^2 \right] + A_2. \tag{59}$$

With Lemma 4, the Lyapunov function V satisfies the following inequalities

$$V^{\frac{3}{4}} \leq \left(\frac{1}{2}s_1^2 + \frac{1}{2}s_2^2\right)^{\frac{3}{4}} + \left(\frac{1}{2\lambda}\tilde{\sigma}^2\right)^{\frac{3}{4}}, \tag{60}$$

and

$$V^2 \leq 2 \left[\left(\frac{1}{2}s_1^2 + \frac{1}{2}s_2^2\right)^2 + \left(\frac{1}{2\lambda}\tilde{\sigma}^2\right)^2 \right]. \tag{61}$$

Then, substituting (60) and (61) into (59) yields

$$\dot{V} \leq -\frac{\gamma}{T_c} \alpha V^{\frac{3}{4}} - \frac{\gamma}{T_c} \beta V^2 + A_2. \tag{62}$$

Owing to the conditions of Lemma 2 and the definition of V , it can be concluded from (62) that

$$|s_1| < \min \left\{ 2^{\frac{1}{4}} \left(\frac{A_2}{1-\phi} \frac{T_c}{\alpha\gamma} \right)^{\frac{2}{3}}, 2^{\frac{1}{4}} \left(\frac{A_2}{1-\phi} \frac{T_c}{\beta\gamma} \right)^{\frac{1}{4}} \right\}, \forall t \geq T_c. \tag{63}$$

With $0 < \phi < 1$, it shows that the tracking error can converge to a neighborhood near the origin within the settling time T_c by choosing appropriate parameters. This completes our proof. \square

To avoid system singularities, the intermediate variable $\bar{\xi}_2$ in (37) can be modified to

$$\begin{aligned} \bar{\xi}_2 = & \frac{\gamma}{T_c} k_2 |s_2|^{\frac{1}{2}} \text{sign}(s_2) + \frac{\gamma}{T_c} c_2 s_2^3 + \frac{3}{2} s_2 \\ & + s_1 + \eta + \frac{s_1(\eta - \dot{y}_d) \text{sign}(s_2)}{\sqrt{s_2^2 + s_0^2}} \\ & + \frac{\hat{\sigma} s_2 \Psi^T \Psi}{2a^2}, \end{aligned} \tag{64}$$

where s_0 is a small positive constant.

Remark 3 Although the practical predefined-time stability can be guaranteed under actuator saturation, it does not

mean that the control torque can be increased indefinitely. Theoretically, the control inputs cannot be saturated for long periods, as it could lead to system instability or even damage. As shown during the proof, more severe torque saturation implies larger steady-state tracking errors. Therefore, control parameters should be reasonably chosen in practical applications of robotic manipulators to avoid actuator saturation.

Remark 4 The proposed controller contains several control parameters that should be carefully selected based on the following principles. First, p, q, ϕ, ρ are used to calculate the control parameter γ , and they can be selected as $p = 0.7, q = 1.2, \phi = 0.95, \rho = 0.5$, which is generally not required to be tuned. Then, for the other control parameters, the larger λ, k_i, c_i ($i = 1, 2$) and smaller a, l_i ($i = 1, 2$) contribute to a better tracking performance, but lead to a larger control torque. Therefore, when tuning these control parameters, a trade-off should be made between the control input and the control performance. Generally, c_i, a, λ can be chosen as $c_i = 1.7, a = 2, \lambda = 0.1$, and k_i, l_i should be chosen by trial-and-error for a good tracking performance.

4 Numerical simulation

In this section, we present the application of the proposed control scheme to two robotic manipulators, one with two joints and the other with nine joints. The dynamic models of these manipulators are unknown, and we utilize them to showcase the effectiveness of the control approach.

4.1 Simulation and comparison of a two-joint robotic manipulator

A two-joint robotic manipulator is employed to demonstrate the efficacy of the proposed control algorithm and its superiority over other existing control algorithms. The robotic model is consistent with that in [15]. The dynamic model parameters of the robotic manipulator are set as $l_1 = 1$ m, $l_2 = 0.8$ m, $m_1 = 0.5$ kg, $m_2 = 1.5$ kg, $I_1 = I_2 = 5$ kg · m², where l_i, m_i and I_i are the length, mass, and inertia of link i , with $i = 1, 2$. $g = 9.81$ m/s² denotes the acceleration of gravity. The nominal values of l_1^0, l_2^0 are defined as $l_1^0 = l_2^0 = 0.9$ m, and the nominal values of m_1^0, m_2^0 are $m_1^0 = 0.6$ kg, $m_2^0 = 1.8$ kg, and the nominal values of I_1^0, I_2^0 are $I_1^0 = I_2^0 = 5$ kg · m². The maximum permissible input control torque for each joint of the robotic manipulator is 300 Nm.

This study compares six existing control algorithms, comprising four model-free algorithms and two algorithms based on dynamical models. The four model-free control algorithms include the proportion integration

differentiation (PID), the adaptive radial basis function neural network (ARBFNN) in [48], the adaptive fuzzy backstepping tracking control (AFBTC) in [49], and the direct adaptive fuzzy control (DAFC) in [50]. Two control algorithms based on the system dynamic model are the second-order predefined-time sliding mode control (SOPSMC) in [30], and the singularity-free fixed-time sliding mode control (SFSMC) in [51]. In these controllers used for comparison, the control parameters of the PID controller are optimized using the PID module in Simulink, while the control parameters of the other controllers are chosen according to the corresponding references. The parameters of the proposed controller are chosen as $p = 0.7$, $q = 1.2$, $\phi = 0.95$, $\rho = 0.5$, $k_1 = 20$, $k_2 = 15$, $a = 2$, $c_1 = c_2 = 1.7$, $l_1 = l_2 = 0.01$, $r_1 = r_2 = \lambda = 0.1$, $T_c = 4$, $\eta(0) = \hat{\sigma}(0) = 0$. Moreover, the NN contains eleven nodes with the center θ_j evenly spaced in $[-4, 4]$ and the width $\varpi_j = 6$.

The desired trajectory of each joint is set as $q_d = 1.25 - (7/5) \exp(-t) + (7/20) \exp(-4t)$. The initial conditions are $q_1(0) = 0, q_2(0) = 0.4, \dot{q}_1(0) = \dot{q}_2(0) = 0$. The numerical simulation is conducted using the Simulink of MATLAB R2020a, with a fundamental step size of 5×10^{-3} s.

The simulation results are displayed in Figs. 2, 3, 4, 5. As can be observed from the joint position tracking trajectories and tracking error results in Figs. 2 and 3, all of the control algorithms successfully achieve tracking of the desired trajectory. Among the algorithms, the proposed control algorithm, DAFC, SOPSMC, and SFSMC demonstrate high tracking accuracy, but the DAFC exhibits significant overshoot during the tracking process, as demonstrated by the tracking error in Fig. 3. The SOPSMC and SFSMC display slightly more accurate tracking error

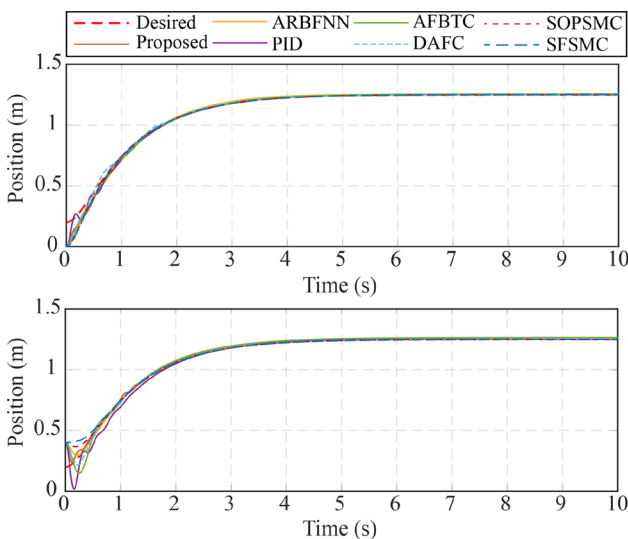


Fig. 2 Tracking trajectories and reference signals

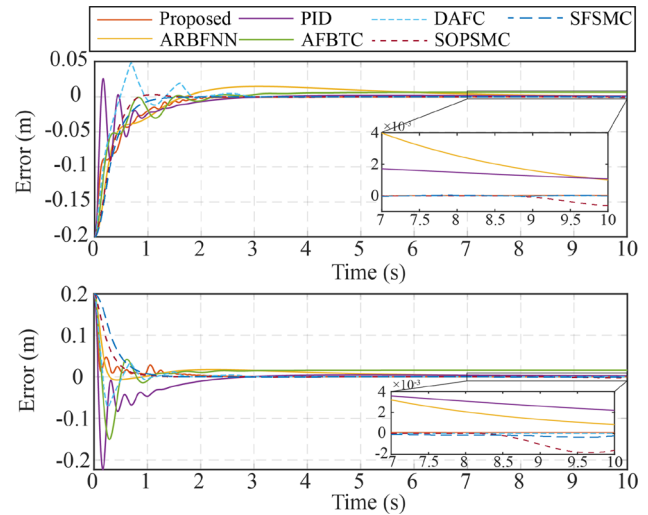


Fig. 3 Joint position tracking error

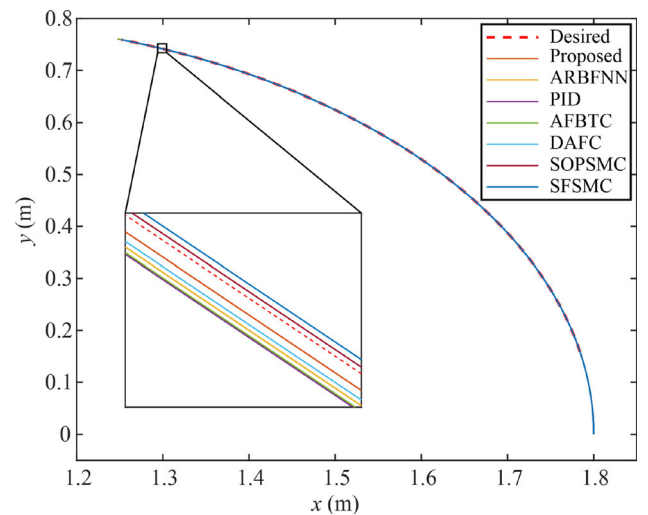


Fig. 4 Tracking trajectory of manipulator end-effector in workspace

of the manipulator end-effector in the workspace, as demonstrated by the tracking trajectory in Fig. 4. However, it is important to note that both the SOPSMC and SFSMC schemes require knowledge of the dynamic model of the robotic manipulator, and their control performance is highly dependent on the accuracy of the dynamic model, making them impractical in real-world applications. In contrast, the proposed control scheme demonstrates effective tracking error convergence within the prescribed time $T_c = 4$ s as shown in Fig. 5, despite the input torque saturation observed in both the proposed and PID controllers.

The performance of the proposed control scheme is further analyzed in Figs. 2, 6, 8. Figure 6 displays the square of the estimated NNs' ideal weights norm, while Fig. 7 demonstrates the norm of the Gaussian function. The dynamical model of the robotic manipulator is defined by

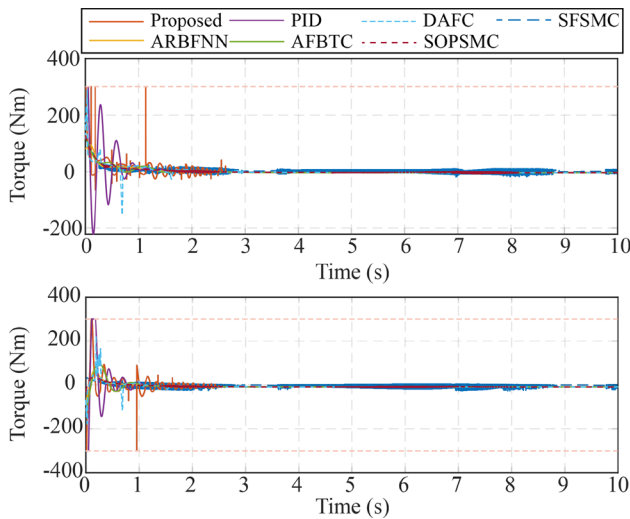


Fig. 5 Joint input torque

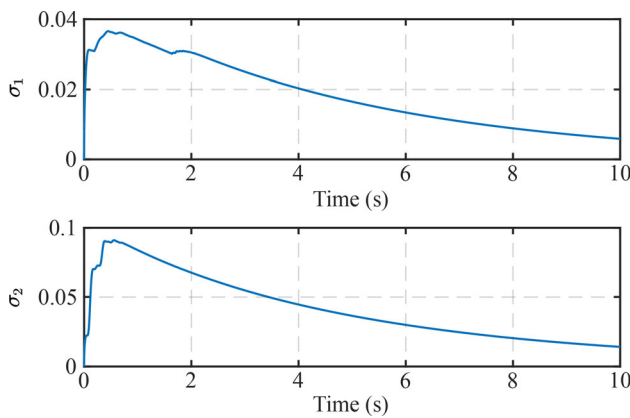


Fig. 6 Estimation weights

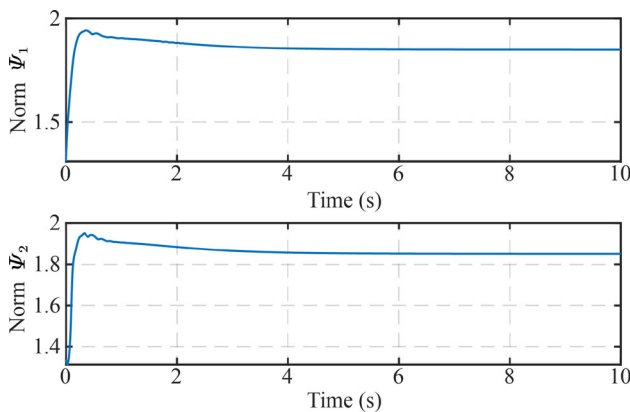


Fig. 7 Norm of the Gaussian function

$f_x = M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q)$, and its approximation by NNs is denoted by f_{xn} . As shown in Fig. 8, the proposed control scheme’s NNs can perform a fast and accurate approximation of the dynamic model.

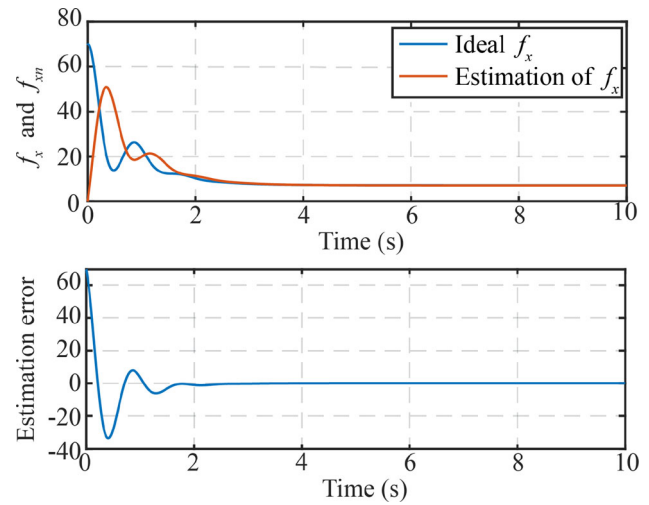


Fig. 8 Approximation values and error of the dynamic model

It is worth emphasizing that the proposed control scheme offers a powerful tool for achieving practical predefined-time stability in the field of robotics, as it addresses a major limitation of existing control algorithms. Specifically, the novel control approach does not require prior knowledge of the dynamic model of the robotic system, which represents a substantial advantage over existing predefined-time control methods. Furthermore, the proposed scheme ensures practical predefined-time stability, even in the presence of actuator saturation, which is a property that is not possessed by the aforementioned control algorithms. While the PID, ARBFNN, AFBTC and DAFC algorithms not require a priori knowledge of the model of the robotic manipulator, they are unable to provide an upper bound on the convergence time. Moreover, although SOPSMC and SFSMC can guarantee that the tracking error converges within a specified time, they require a known model of the system dynamics, rendering their performance invalid once the system becomes saturated.

Moreover, we examined the tracking performance of the proposed control scheme in response to variations in the load on the robotic manipulator. At $t > 5$ s, we modified the mass of link 2 from $m_2 = 1.5$ kg to 3 kg. The tracking error of the robotic joints is presented in Fig. 9. The results reveal that the tracking error of the proposed control scheme experiences a brief fluctuation and swiftly converges to almost zero. On the other hand, the tracking error of the SOPSMC scheme changes considerably, indicating its ineffective compensation for changes in robotic loads. Furthermore, the AFBTC scheme exhibits the lowest tracking accuracy when the load on the robotic manipulator varies. The PID and ARBFNN approaches require a more extended period to restore the tracking error to an accuracy comparable to that before the load alteration.

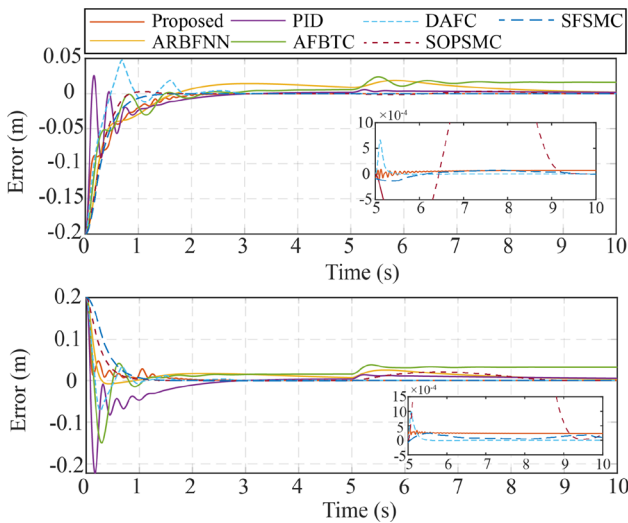


Fig. 9 Joint position tracking error with variable loads

4.2 Simulation of a nine-joint redundant robotic manipulator

As shown in Fig. 10, the controlled objective is a robotic manipulator with nine vertically crossed joints, and the joint positions, velocities, and desired joint positions and velocities are used as inputs to the controller. The robotic model is imported into Simscape and the automatically generated dynamic parameters as the actual parameters, which are only used to calculate the states of the joints and

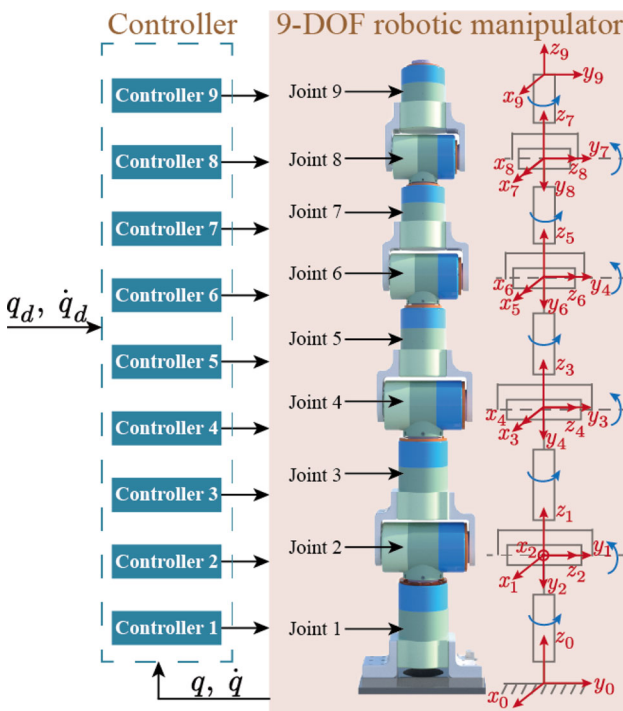


Fig. 10 Schematic of the nine-joint rigid robotic manipulator

not for the controller. The maximum allowed input torque is assumed to be 338 Nm for joints 1 and 2, 238 Nm for joints 3 and 4, 135 Nm for joints 5 and 6, and 69 Nm for joints 7, 8, and 9. Meanwhile, the gravity parameter of the robotic manipulator is set to $g = 9.8 \text{ m/s}^2$. The robotic manipulator shown in Fig. 2 is the initial pose, and the initial position and velocity of each joint are set to zero. To facilitate the presentation of simulation results, the desired trajectory of each joint is set to $q_d = -\frac{4}{5} + \exp(-t) - \frac{1}{4}\exp(-4t)$. The parameters of the controller are chosen as $p = 0.7, q = 1.2, \phi = 0.95, \rho = 0.5, k_1 = 20, k_2 = 5, a = 2, c_1 = c_2 = 1.7, l_1 = l_2 = r_1 = \lambda = 0.1, r_2 = 2, T_c = 8, \eta(0) = 0, \hat{\sigma}(0) = 10$. The parameters associated with the NN are the same as those in Sect. 4.1. The numerical simulation is conducted using the Simscape of MATLAB R2020a, with a fundamental step size of $5 \times 10^{-3} \text{ s}$.

The numerical simulation results are shown in Figs. 4, 11, 13. Figures 11 and 12 show the tracking trajectory and tracking error of each joint, respectively. It can be seen that all joints can quickly and accurately track the desired trajectory, and the tracking error converges to a neighborhood near zero within 4 s. The input torque of each joint in Fig. 13 shows that the control torques of joints 2, 4, 7 and 9 are saturated within 1 s but can still ensure the practical predefined-time stability of the tracking error, which is consistent with the description in Theorem 1. Moreover, the control torque has a significant chattering before 1.5 s, which is mainly due to NNs approximating the dynamic model of the robotic manipulator.

With the forward kinematic and joint angles of the robotic manipulator, the desired trajectory and the actual trajectory of the manipulator end-effector varying with time in the workspace can be calculated. As shown in Fig. 14, it shows that the manipulator end-effector has an accurate tracking of the desired trajectory after 1 s. Figure 15 shows the tracking errors of the position and attitude of the manipulator end-effector, indicating that the proposed controller can ensure good tracking performance for the manipulator end-effector. Therefore, combined with the inverse kinematics of the manipulator, the proposed control scheme can be used for the manipulator end-effector to accurately track the desired trajectory in the workspace within a predefined time.

5 Experiments results

In this section, the experimental results are used to demonstrate the effectiveness of the proposed control scheme based on a self-developed nine-joint redundant manipulator. The experimental system is shown in Fig. 16.

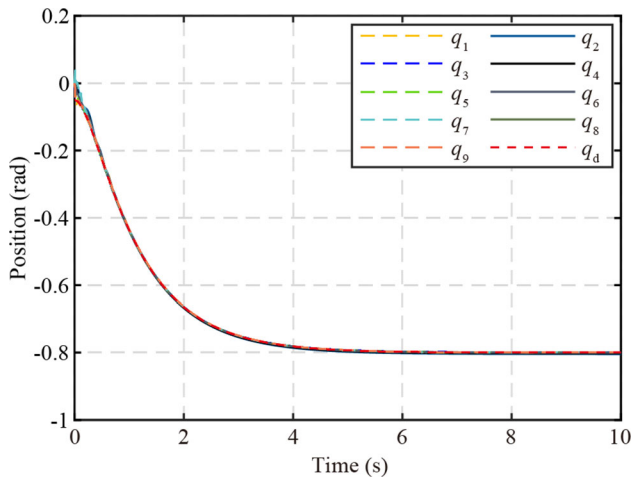


Fig. 11 Tracking trajectories and reference signal

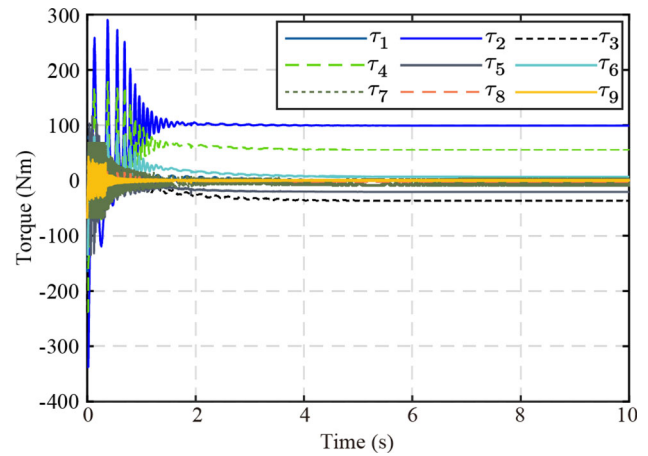


Fig. 13 Joint input torque

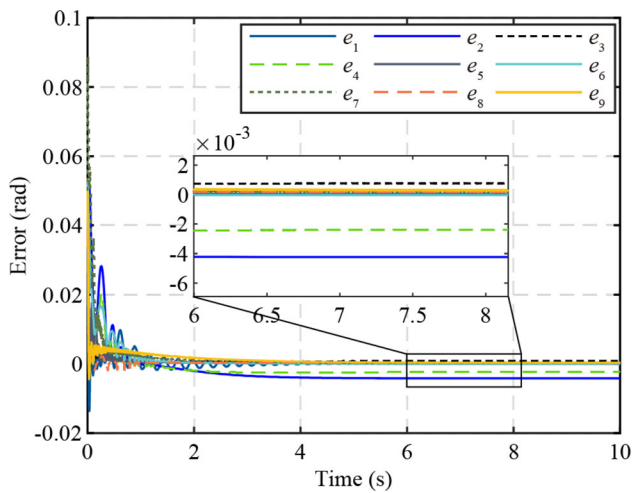


Fig. 12 Joint position tracking error

The nine-joint manipulator used in this experiment is consistent with the structure of the model used in the simulation, and each joint motor is equipped with a rotary encoder to measure the joint angle, and the joint velocity is obtained by differential calculation of the joint angle. The control system of the robotic manipulator is developed based on TwinCAT3, which supports PLC, C++ and MATLAB/Simulink for algorithm design and implementation. Then, the communication with the lower computer can be realized by EtherCAT, and the specific control instructions are sent to the motion controller at the time step of 5×10^{-3} s. To obtain the tracking trajectory of the manipulator end-effector, an optical tracking measurement device (Aicon-Movelspect-XR8) is used to provide position and attitude measurement for the manipulator end-effector, which allowed for more accurate acquisition of the actual position of the manipulator end-effector in the

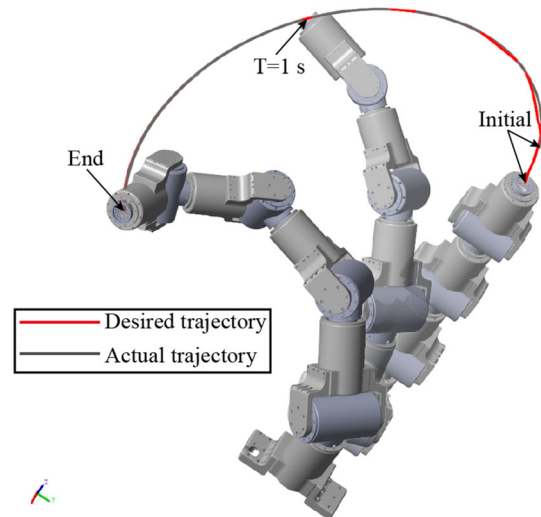


Fig. 14 Tracking trajectory of manipulator end-effector in workspace

working space than those obtained solely through joint position calculations.

Considering the possible damage to the robotic manipulator and illustrating the effectiveness of the proposed controller, the input torque range of joints 1 and 2 is bounded to 120 Nm. The maximum torque allowed by other joints is constrained to the maximum torque of the motor output, which is consistent with the setting in Sect. 4.2. The initial joint angle of the robotic manipulator is set to $q_0 = [0, 25^\circ, 0, 25^\circ, 0, 25^\circ, 0, 25^\circ, 0]^T$, and the initial velocity is zero. The desired joint trajectory is set to $q_{di} = 0.01 + 0.15 \sin(\frac{\pi}{30}t)$ ($i = 1, 3, 5, 7, 9$), $q_{d2} = q_{d4} = 0.01 + \frac{25}{180}\pi + 0.1 \sin(\frac{\pi}{30}t)$, $q_{d6} = q_{d8} = 0.01 + \frac{25}{180}\pi - 0.1 \sin(\frac{\pi}{30}t)$. The control parameters are set to $l_i = 0.01$ ($i = 1, 2$), $T_c = 10$, and other control parameters are consistent with those in the simulation.

Fig. 15 The three-axis position and attitude errors of the end-effector. **a** The position tracking error. **b** The attitude tracking error

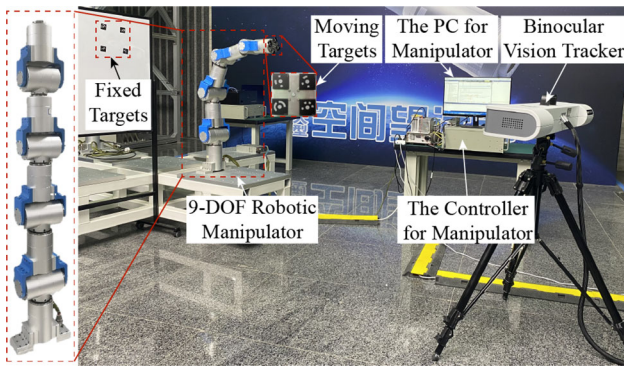
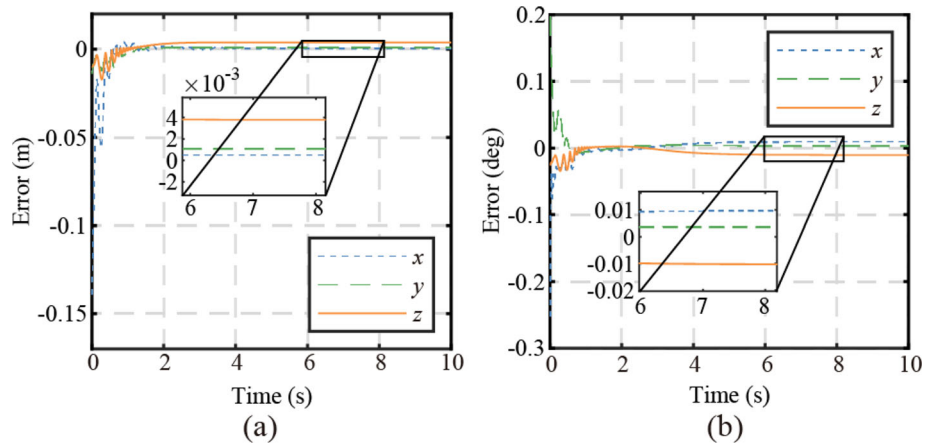
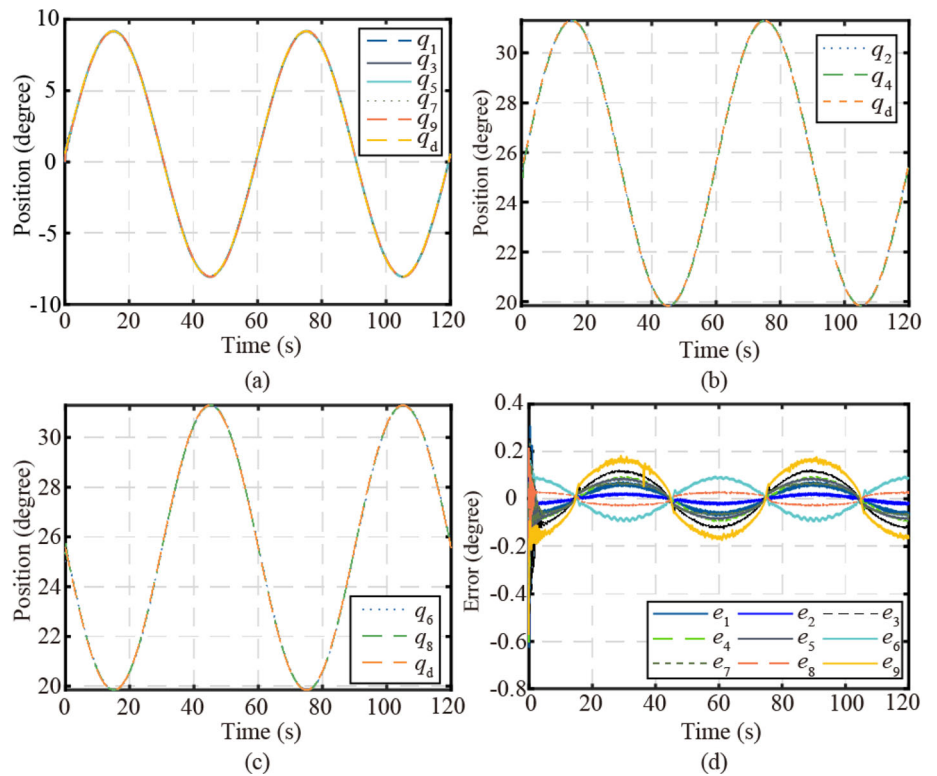


Fig. 16 The experimental system for a nine-joint robotic manipulator

The experimental results are presented in Figs. 17, 18, 19. Figure 17 shows that each joint has a good tracking performance for the desired trajectory, and the tracking error converges to a neighborhood near zero with a steady-state error of less than 0.2° within 6 s. Meanwhile, a transient saturation of the input torque of joint 2 is observed at the initial time. The three-axis position and attitude errors of the end-effector are shown in Fig. 19, and the maximum three-axis position and attitude errors are [0.0132 m, 0.0156 m, 0.0127 m] and [2.5559° , 0.6861° , 1.1480°], respectively. From Fig. 19, the maximum error in attitude tracking occurs during the initial tracking, which is caused by the chattering at the initial time. Considering the

Fig. 17 Tracking trajectory and error of the robotic manipulator. **a** Tracking trajectories of joints 1, 3, 5, 7, 9 and reference signal. **b** Tracking trajectories of joints 2, 4 and reference signal. **c** Tracking trajectories of joints 6, 8 and reference signal. **d** Joint position tracking error



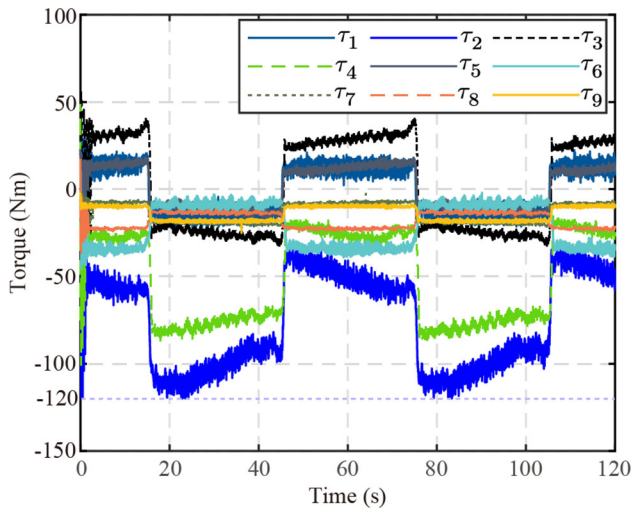


Fig. 18 Joint input torque

measurement error and the properties of the nine-joint manipulator, it can be concluded that the proposed control scheme has a good control performance in the tracking control of a multi-joint robotic manipulator.

After analyzing the simulation and experimental results presented above, it can be concluded that despite the proposed control scheme’s complexity, it still exhibits good real-time performance and satisfies the trajectory tracking control requirements for robotic systems.

6 Comparison with the existing results

In comparison with the existing finite-time control schemes, fixed-time control schemes and predefined-time control schemes, the novelty of the proposed control scheme is highlighted as follows.

(1) In contrast to existing finite-time control schemes [6–8, 52], the proposed control scheme offers a settling-time bound that depends solely on an adjustable control parameter. This feature allows the determination of an a priori tunable parameter that serves as an upper bound for

the settling time, regardless of the initial error of the robotic joints. This predefined-time stability is valuable for tasks with severe constraints on the settling time of the robotic manipulator, such as the grasping task of a space robotic manipulator [53] or scenarios where the initial state of the system is not available.

(2) The existing fixed-time control schemes proposed in [10–16] ensure that the upper bound for the settling time satisfies a complex function constructed from the control parameters, which makes the tuning of the settling-time bound complicated and unintuitive in practical applications. Moreover, the upper bound for the settling time tends to significantly overestimate the least upper one in the fixed-time controllers. Although some fixed-time control schemes have combined NN techniques to make them applicable to more complex systems, the above problems cannot be solved [17, 19, 20, 54]. Different from these fixed-time control schemes, the proposed controller ensures that the settling-time bound in the robotic system depends only on the settling time T_c and is much less conservative. This means that the upper bound for the settling time of the proposed control scheme is easier to set and more reasonable.

(3) Most of the existing predefined-time control schemes are designed based on system models, such as the predefined-time controllers for robotic manipulators in [22, 23, 27, 38]. However, the dynamic model of the multi-joint manipulator is quite complicated; thus, the above controllers have only been simulated or experimentally validated by a 2-DOF or 3-DOF robotic manipulator. The proposed practical predefined-time stability criterion allows us to apply the NN technique to approximate the dynamic model of an unknown system, which makes it easier to be applied to multi-joint robotic manipulators. Meanwhile, the saturation of actuators in predefined-time control is considered, which has important practical implications for the safety of the manipulator control.

Finally, Table 1 categorizes eleven cases based on the convergence time of the system, dependence on the robotic dynamic model, and consideration of system saturation,

Fig. 19 The three-axis position and attitude errors of the end-effector. a The position tracking error. b The attitude tracking error

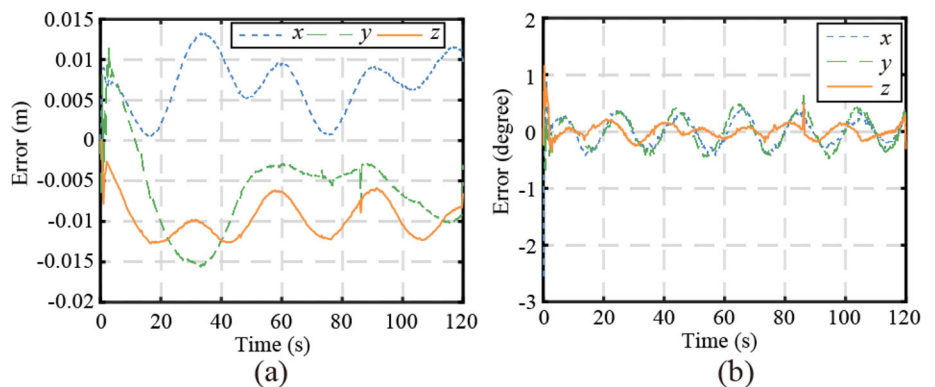


Table 1 Comparisons among different trajectory tracking schemes for robotic manipulators

	System stability	Whether the upper bound on the settling time can be estimated?	Whether the upper bound on the settling time is independent of the initial state?	Whether the upper bound on the settling time depends on a control parameter?	Whether to request dynamic models for robotic manipulators?	Whether actuator saturation is considered?
This paper	Predefined-time stable	Yes	Yes	Yes	No	Yes
Paper [55]	Asymptotically stable	No	No	No	Yes	No
Paper [56]	Asymptotically stable	No	No	No	No	No
Paper [57]	Asymptotically stable	No	No	No	No	Yes
Paper [52]	Finite-time stable	Yes	No	No	Yes	No
Paper [58]	Finite-time stable	Yes	No	No	No	No
Paper [59]	Finite-time stable	Yes	No	No	No	Yes
Paper [15]	Fixed-time stable	Yes	Yes	No	Yes	No
Paper [20]	Fixed-time stable	Yes	Yes	No	No	No
Paper [19]	Fixed-time stable	Yes	Yes	No	No	Yes
Paper [23]	Predefined-time stable	Yes	Yes	Yes	Yes	No

with representative works given for each. In practical applications of robotic manipulators, control engineering faces challenges regarding system models, unknown disturbances, and control input feasibility. However, the proposed control scheme circumvents these issues by not requiring a priori knowledge of the model and disturbances, nor concerns itself with excessive control inputs. As such, it proves to be a feasible control scheme for practical applications of robotic manipulators.

7 Conclusion

In this work, an adaptive practical predefined-time neural control scheme is designed for multi-joint manipulators with unknown dynamic parameters. Since the system uncertainty invalidates the predefined-time stability criterion, a novel practical predefined-time stability criterion is developed. The requirement for mechanical dynamic parameters is avoided by the neural network technique, while an adaptive term is used to compensate for the adverse effects on the control effect due to actuator saturation. Then, the practical predefined-time stability of the

robotic system is rigorously demonstrated based on the established stability criterion. The tracking error can converge to a small neighborhood near the origin within a predefined time, and the upper bound for the settling time is independent of the initial state of the system. The numerical simulation and experimental results of robotic manipulators emphasize the feasibility and reliability of the proposed control scheme, indicating the application potential of the predefined-time control theory in the control of the multi-joint robotic manipulator.

In future work, we aim to develop control algorithms with predefined-time stability for hyper-redundant or continuum manipulators and improve the estimation of settling time for robotic manipulators, as the current upper bound is always overestimated.

Supplementary Information The online version contains supplementary material available at <https://doi.org/10.1007/s00521-023-08797-2>.

Acknowledgements This work is partially supported by the National Natural Science Foundation of China (11972343).

Data availability Data sharing not applicable to this article as no datasets were generated or analyzed during the current study.

Declarations

Conflict of interest The authors declare no potential conflict of interests.

References

- Pan Y, Du P, Xue H, Lam HK (2020) Singularity-free fixed-time fuzzy control for robotic systems with user-defined performance. *IEEE Trans Fuzzy Syst* 29(8):2388–2398
- Wang H, Liu S, Yang X (2020) Adaptive neural control for non-strict-feedback nonlinear systems with input delay. *Inf Sci* 514:605–616
- Sai H, Xu Z, Xu C, Wang X, Wang K, Zhu L (2022) Adaptive local approximation neural network control based on extraordinariness particle swarm optimization for robotic manipulators. *J Mech Sci Technol* 36(3):1469–1483
- Haimo VT (1986) Finite time controllers. *SIAM J Control Optim* 24(4):760–770
- Shao K, Zheng J, Huang K, Wang H, Man Z, Fu M (2019) Finite-time control of a linear motor positioner using adaptive recursive terminal sliding mode. *IEEE Trans Industr Electron* 67(8):6659–6668
- Zhihong M, O'day M, Yu X (1999) A robust adaptive terminal sliding mode control for rigid robotic manipulators. *J Intell Robot Syst* 24(1):23–41
- Yu X, Zhihong M (2002) Fast terminal sliding-mode control design for nonlinear dynamical systems. *IEEE Trans Circuits Syst I Fundam Theory Appl* 49(2):261–264
- Yang L, Yang J (2011) Nonsingular fast terminal sliding-mode control for nonlinear dynamical systems. *Int J Robust Nonlinear Control* 21(16):1865–1879
- Polyakov A (2011) Nonlinear feedback design for fixed-time stabilization of linear control systems. *IEEE Trans Autom Control* 57(8):2106–2110
- Zuo Z (2015) Non-singular fixed-time terminal sliding mode control of non-linear systems. *IET Control Theory Appl* 9(4):545–552
- Zuo Z (2015) Nonsingular fixed-time consensus tracking for second-order multi-agent networks. *Automatica* 54:305–309
- Zuo Z, Han QL, Ning B, Ge X, Zhang XM (2018) An overview of recent advances in fixed-time cooperative control of multi-agent systems. *IEEE Trans Industr Inf* 14(6):2322–2334
- Ning B, Han QL, Zuo Z, Ding L, Lu Q, Ge X (2022) Fixed-time and prescribed-time consensus control of multi-agent systems and its applications: a survey of recent trends and methodologies. *IEEE Trans Ind Inf*
- Su Y, Zheng C, Mercorelli P (2020) Robust approximate fixed-time tracking control for uncertain robot manipulators. *Mech Syst Signal Process* 135:106379
- Sai H, Xu Z, Xia C, Sun X (2022) Approximate continuous fixed-time terminal sliding mode control with prescribed performance for uncertain robotic manipulators. *Nonlinear Dyn* 1–18
- Yang P, Su Y (2021) Proximate fixed-time prescribed performance tracking control of uncertain robot manipulators. *IEEE/ASME Trans Mechatron*
- Ba D, Li YX, Tong S (2019) Fixed-time adaptive neural tracking control for a class of uncertain nonstrict nonlinear systems. *Neurocomputing* 363:273–280
- Wang F, Lai G (2020) Fixed-time control design for nonlinear uncertain systems via adaptive method. *Syst Control Lett* 140:104704
- Hu Y, Yan H, Zhang H, Wang M, Zeng L (2022) Robust adaptive fixed-time sliding-mode control for uncertain robotic systems with input saturation. *IEEE Trans Cybern* 53(4):2636–2646
- Chen M, Wang H, Liu X (2021) Adaptive practical fixed-time tracking control with prescribed boundary constraints. *IEEE Trans Circuits Syst I Regul Pap* 68(4):1716–1726
- Sánchez-Torres JD, Gómez-Gutiérrez D, López E, Loukianov AG (2018) A class of predefined-time stable dynamical systems. *IMA J Math Control Inf* 35(Supplement_1):i1–i29
- Muñoz-Vázquez AJ, Sánchez-Torres JD, Gutiérrez-Alcalá S, Jiménez-Rodríguez E, Loukianov AG (2019) Predefined-time robust contour tracking of robotic manipulators. *J Franklin Inst* 356(5):2709–2722
- Munoz-Vazquez AJ, Sánchez-Torres JD, Jimenez-Rodriguez E, Loukianov AG (2019) Predefined-time robust stabilization of robotic manipulators. *IEEE/ASME Trans Mechatron* 24(3):1033–1040
- Ni J, Liu L, Tang Y, Liu C (2019) Predefined-time consensus tracking of second-order multiagent systems. *IEEE Trans Syst Man Cybern Syst* 51(4):2550–2560
- Sai H, Li Y, He S, Zhang E, Zhu M, Xu Z (2022) A nine-degree-of-freedom modular redundant robotic manipulator: development and experimentation. *Proc Inst Mech Eng Part C J Mech Eng Sci* 09544062221139968
- Xie Z, Jin L (2022) Hybrid control of orientation and position for redundant manipulators using neural network. *IEEE Trans Syst Man Cybern Syst*
- Muñoz-Vázquez AJ, Sánchez-Torres JD (2020) Predefined-time control of cooperative manipulators. *Int J Robust Nonlinear Control* 30(17):7295–7306
- Shuzhi SG, Hang CC, Woon L (1997) Adaptive neural network control of robot manipulators in task space. *IEEE Trans Ind Electron* 44(6):746–752
- Aldana-López R, Gómez-Gutiérrez D, Jiménez-Rodríguez E, Sánchez-Torres JD, Defoort M (2019) Enhancing the settling time estimation of a class of fixed-time stable systems. *Int J Robust Nonlinear Control* 29(12):4135–4148
- Sánchez-Torres JD, Defoort M, Muñoz-Vázquez AJ (2018) Second order sliding mode controller with predefined-time convergence. In: 15th International conference on electrical engineering, computing science and automatic control (CCE). *IEEE* 2018:1–4
- Wu C, Yan J, Shen J, Wu X, Xiao B (2021) Predefined-time attitude stabilization of receiver aircraft in aerial refueling. *IEEE Trans Circuits Syst II Express Briefs* 68(10):3321–3325
- Wang Z, Liang B, Sun Y, Zhang T (2019) Adaptive fault-tolerant prescribed-time control for teleoperation systems with position error constraints. *IEEE Trans Ind Inf* 16(7):4889–4899
- Liu B, Hou M, Wu C, Wang W, Wu Z, Huang B (2021) Predefined-time backstepping control for a nonlinear strict-feedback system. *Int J Robust Nonlinear Control* 31(8):3354–3372
- Sai H, Xu Z, He S, Zhang E, Zhu L (2022) Adaptive nonsingular fixed-time sliding mode control for uncertain robotic manipulators under actuator saturation. *ISA Trans* 123:46–60
- Zhu G, Du J (2020) Robust adaptive neural practical fixed-time tracking control for uncertain Euler–Lagrange systems under input saturations. *Neurocomputing* 412:502–513
- Jia S, Shan J (2019) Finite-time trajectory tracking control of space manipulator under actuator saturation. *IEEE Trans Ind Electron* 67(3):2086–2096
- Jiménez-Rodríguez E, Loukianov AG, Sánchez-Torres JD (2017) A second order predefined-time control algorithm. In: 2017 14th international conference on electrical engineering, computing science and automatic control (CCE). *IEEE*, pp 1–6

38. Zhang N, Wang S, Hou Y, Zhang L (2020) A robust predefined-time stable tracking control for uncertain robot manipulators. *IEEE Access* 8:188600–188610
39. Wu Y, Li G, Zuo Z, Liu X, Xu P (2020) Practical fixed-time position tracking control of permanent magnet DC torque motor systems. *IEEE/ASME Trans Mechatron* 26(1):563–573
40. Bateman H (1953) Higher transcendental functions [volumes i-iii]. vol. 1. McGraw-Hill Book Company
41. Jin X (2018) Adaptive fixed-time control for MIMO nonlinear systems with asymmetric output constraints using universal barrier functions. *IEEE Trans Autom Control* 64(7):3046–3053
42. Qian C, Lin W (2001) Non-Lipschitz continuous stabilizers for nonlinear systems with uncontrollable unstable linearization. *Syst Control Lett* 42(3):185–200
43. Sciacivico L, Siciliano B (2001) Modelling and control of robot manipulators. Springer
44. Jing C, Xu H, Niu X (2019) Adaptive sliding mode disturbance rejection control with prescribed performance for robotic manipulators. *ISA Trans* 91:41–51
45. Sai H, Xu Z, Han T, Wang X, Li H (2022) Observer-based free-will arbitrary time sliding mode control for uncertain robotic manipulators. *J Vib Control* 10775463221138327
46. Ge SS, Hang CC, Lee TH, Zhang T (2013) Stable adaptive neural network control, vol 13. Springer
47. Wen C, Zhou J, Liu Z, Su H (2011) Robust adaptive control of uncertain nonlinear systems in the presence of input saturation and external disturbance. *IEEE Trans Autom Control* 56(7):1672–1678
48. Sai H, Xu Z, Xu C, Wang X, Wang K, Zhu L (2022) Adaptive local approximation neural network control based on extraordinariness particle swarm optimization for robotic manipulators. *J Mech Sci Technol* 36(3):1469–1483
49. Chang W, Li Y, Tong S (2018) Adaptive fuzzy backstepping tracking control for flexible robotic manipulator. *IEEE/CAA J Autom Sin* 8(12):1923–1930
50. Chen B, Liu X, Liu K, Lin C (2009) Direct adaptive fuzzy control of nonlinear strict-feedback systems. *Automatica* 45(6):1530–1535
51. Zhang L, Wang Y, Hou Y, Li H (2019) Fixed-time sliding mode control for uncertain robot manipulators. *IEEE Access* 7:149750–149763
52. Jin L, Zhang F, Liu M, Xu SSD (2022) Finite-time model predictive tracking control of position and orientation for redundant manipulators. *IEEE Trans Ind Electron*
53. Li R, Qiao H (2019) A survey of methods and strategies for high-precision robotic grasping and assembly tasks—some new trends. *IEEE/ASME Trans Mechatron* 24(6):2718–2732
54. Sun W, Diao S, Su SF, Sun ZY (2021) Fixed-time adaptive neural network control for nonlinear systems with input saturation. *IEEE Trans Neural Netw Learn Syst*
55. Van M (2019) An enhanced tracking control of marine surface vessels based on adaptive integral sliding mode control and disturbance observer. *ISA Trans* 90:30–40
56. Santos JC, Gouttefarde M, Chemori A (2022) A nonlinear model predictive control for the position tracking of cable-driven parallel robots. *IEEE Trans Robot* 38(4):2597–2616
57. Wu Y, Wang Y (2020) Asymptotic tracking control of uncertain nonholonomic wheeled mobile robot with actuator saturation and external disturbances. *Neural Comput Appl* 32:8735–8745
58. Wang Y, Zhu K, Chen B, Jin M (2020) Model-free continuous nonsingular fast terminal sliding mode control for cable-driven manipulators. *ISA Trans* 98:483–495
59. Kong L, He W, Yang W, Li Q, Kaynak O (2020) Fuzzy approximation-based finite-time control for a robot with actuator saturation under time-varying constraints of work space. *IEEE Trans Cybern* 51(10):4873–4884

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

Springer Nature or its licensor (e.g. a society or other partner) holds exclusive rights to this article under a publishing agreement with the author(s) or other rightsholder(s); author self-archiving of the accepted manuscript version of this article is solely governed by the terms of such publishing agreement and applicable law.