A Systematic Trajectory Tracking Framework for Robot Manipulators: An Observer-Based Non-Smooth Control Approach

Linyan Han, Jianliang Mao, *Member, IEEE*, Chuanlin Zhang, *Senior Member, IEEE*, Robert W. Kay, Robert C. Richardson, and Chengxu Zhou

***Abstract*—The mechanical design of a robot often influences the choice of control strategy, especially for high-dimensional manipulator systems with multiple inputs and outputs. Striking a balance between hardware and software, it remains a significant challenge to design a control framework that is both easy to implement and high-performing. This paper addresses this concern by developing a systematic control architecture for trajectory tracking problems, focusing solely on position measurements. The approach involves constructing a dynamic model of the manipulator in joint space through parameter identification techniques. A non-smooth observer is then devised to estimate unmeasured states, unknown disturbances, and uncertain nonlinear functions in real-time, which are incorporated into a non-smooth feedback control design to provide a control solution. The stability of the system is ensured using the homogeneous system theory and Lyapunov theorems. To validate the effectiveness and feasibility of the proposed tracking control approach, extensive evaluations are conducted on a six-degree-of-freedom (6-DoF) manipulator, including tests for tracking performance and repeatability.**

***Index Terms*—robot manipulator, trajectory tracking, position feedback, non-smooth control, dynamics control.**

I. Introduction

MOTIVATED by the demand for superior performance, continuous trajectory tracking control is paramount in a broad spectrum of tasks, e.g., reaching tasks [1] and the repair of disabled satellites [2]. This necessity distinguishes it from point-to-point control, which concentrates solely on the accuracy of the final position. To attain exceptional tracking precision, numerous methodologies have been devised in recent years, including both independent joint control and multi-variable control [3]. Nevertheless, certain challenges remain inadequately addressed in these studies. The independent joint control method treats each joint of the robot individually and considers the coupling effects as external disturbances, akin to the philosophy of a single-input/single-output (SISO) system [4], [5]. Given that the inherent coupling characteristics of robotic systems are predominantly nonlinear and exert a substantial influence on robot performance, the oversimplified approach of independent joint control can impose restrictions on its applicability in systems that are strongly coupled and highly dynamic. Conversely, the multi-variable control methods offer the advantage of delivering rigorous system performance analysis, alongside ensuring the stability of the closed-loop system. This control structure presents a more holistic and integrative approach compared to independent joint control.

The majority of multi-variable control methodologies, such as Proportional-Derivative (PD) control with gravity compensation [6], [7] and Inverse Dynamics Control (IDC) [8], [9] rely heavily on full state feedback, necessitating measurements of both joint position and velocity. In practical applications, encoders provide highly precise joint position measurements. However, joint velocity measurements are commonly obtained by calculating the difference between position data, leading to noise and measurement errors [10]. Consequently, the overall performance of the system may be compromised, as the noise inherently limits the controller gains and restricts the achievable bandwidth. To obviate the need for joint velocity measurements, several techniques for developing an efficient tracking controller based on available measurement data have been proposed. For instance, a specific category of adaptive output feedback controllers for robots devoid of velocity measurements was developed in [11]–[13]. In [14], an output feedback PID regulator with an integral action was formulated, guaranteeing the global asymptotic stability for position tracking control of robots. However, these aforementioned studies are primarily concerned with the stability analysis of robotic systems in the presence of disturbances. The robustness of these systems is generally achieved at the expense of nominal control performance [15].

Observer-based output-feedback control offers an alternative

This work was supported in part by the Advanced Machinery and Productivity Institute [Innovate UK project 84646], in part by the Engineering and Physical Sciences Research Council [Grant EP/V026801/2], in part by the National Natural Science Foundation of China [Grants 62203292 and 62173221], and in part by the Fundamental Research Funds for the Central Universities [Grant 2242022k30038]. (*Corresponding author: Jianliang Mao; Chengxu Zhou*)

Linyan Han, Robert W. Kay, Robert C. Richardson, and Chengxu Zhou are with the School of Mechanical Engineering, University of Leeds, LS2 9JT, UK (e-mail: l.y.han@leeds.ac.uk; r.w.kay@leeds.ac.uk; r.c.richardson@leeds.ac.uk; c.x.zhou@leeds.ac.uk).

Jianliang Mao and Chuanlin Zhang are with the College of Automation Engineering, Shanghai University of Electric Power, Shanghai 200090, PR China (e-mail: jl_mao@shiep.edu.cn; clzhang@shiep.edu.cn).

Linyan Han and Jianliang Mao are also with the Key Laboratory of Measurement and Control of Complex Systems of Engineering, Ministry of Education, Southeast University, Nanjing 210096, China.

solution to overcome the limitations of the before-mentioned methods. Several observer techniques have been developed since the 1960s, such as unknown input observer (UIO) [16], disturbance observer (DOB) [17], extended state observer (ESO) [18] and generalized proportional integral observer (GPIO) [19]. These observers are combined with advanced feedback control to form observer-based control, which is used to compensate unmodeled dynamics and external disturbances and has been widely applied in robotics [20]–[23]. In [24], a nonlinear observer was employed to estimate both joint velocity and unknown disturbances simultaneously, utilizing a neural network to handle uncertain nonlinear functions. Similarly, the method in [25] exploited an adaptive fuzzy controller to estimate the nonlinear functions. These findings present an effective strategy for achieving position control for robots. However, they merely guarantee that the system is uniformly bounded, signifying that the system state trajectories converge near the equilibrium as the time approaches infinity. For tasks requiring high-speed and high-accuracy performance, these outcomes may not be sufficiently efficient.

A more potent alternative that provides rapid response, disturbance rejection, and high-precision capabilities is finite-time stabilization [26], which is essentially realized via a non-smooth control technique. Non-smooth control stands as a nonlinear control methodology that bridges the gap between smooth control and discontinuous control, demonstrating significant success across a wide spectrum of applications [27], [28]. It has been reported in [29] that a non-smooth observer is capable of estimating unknown disturbances and system states concurrently, exhibiting numerous advantages such as finite-time convergence and improved estimation accuracy. This makes it a more desirable choice for strategies aimed at achieving both high-precision and high-speed control for robot manipulators.

In summary, our objective is to establish a systematic tracking control framework that encompasses modeling, disturbance estimation and compensation, as well as non-smooth feedback control for robot manipulators to tackle the challenges mentioned earlier. Toward this end, we initially employ a parameter identification method to reconstruct the nominal dynamic model. Subsequently, a non-smooth observer is deployed to estimate unknown disturbances. In this context, both the modeled dynamics and estimated disturbances can be efficiently compensated through a feedforward approach. Additionally, the non-smooth feedback domination design is implemented to alleviate the impact of unmodeled dynamics and inestimable disturbances. The efficacy of the proposed approach is validated through rigorous theoretical analysis and extensive experiments conducted on trajectory tracking tasks using a real 6-DoF robot manipulator. The primary contributions of this paper are summarized as follows:

- Compared to traditional inverse dynamics control methods, the proposed control approach integrates model identification, disturbance estimation, and non-smooth feedback domination techniques, making it more capable of achieving high-precision control of robot manipulators.
- In contrast to the backstepping design approach suggested by [25], [30], this trajectory tracking framework employs a non-recursive design procedure [31], which is more practical to implement, particularly for robotic systems with multiple inputs and multiple outputs.
- From a theoretical perspective, the finite-time convergence of the trajectory tracking error is realized, and the stability analysis is conducted through the application of homogeneous system theory in conjunction with a Lyapunov function.

The paper is structured as follows. Section II presents the motivation, while Section III encompasses the main results, including modeling, observer design, and controller design. In Section IV, we evaluate the performance of the proposed approach in terms of precision, robustness, and repeatability. Section V concludes the paper, and the main stability analysis is provided in the Appendix.

II. Motivation

This section provides a discussion on existing control methods, highlighting their limitations and motivating our work to address the issues raised in the previous section.

A. Standard Inverse Dynamics Control

In the joint space, the dynamic equation of an n-DoF robot manipulator with all revolute joints can be formulated as

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}_{ric} = \boldsymbol{\tau} + \boldsymbol{\tau}_e \tag{1}$$

where $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}} \in \mathbb{R}^n$ represent joint position, velocity and acceleration, respectively. $\mathbf{M}(\mathbf{q}) \in \mathbb{R}^{n\times n}$ denotes the symmetric, positive-definite matrix, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) \in \mathbb{R}^{n\times n}$ represents the Coriolis and centrifugal matrix, $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$ is the vector of gravitational forces, and $\mathbf{f}_{ric} \in \mathbb{R}^n$ is the friction forces. The unknown external disturbance is denoted by $\boldsymbol{\tau}_e \in \mathbb{R}^n$ and the control input torque is denoted by $\boldsymbol{\tau} \in \mathbb{R}^n$.

By defining $\mathbf{x}_1 = \mathbf{q}$ and $\mathbf{x}_2 = \dot{\mathbf{q}}$, system (1) can be rearranged as

$$\begin{cases} \dot{\mathbf{x}}_1 = \mathbf{x}_2, \\ \dot{\mathbf{x}}_2 = \mathbf{M}(\mathbf{x}_1)^{-1}\boldsymbol{\tau} + \mathbf{f}(\mathbf{x}_1, \mathbf{x}_2) + \mathbf{d} \end{cases} \tag{2}$$

where

$$\mathbf{f}(\mathbf{x}_1, \mathbf{x}_2) = -\mathbf{M}(\mathbf{x}_1)^{-1}\left(\mathbf{C}(\mathbf{x}_1, \mathbf{x}_2)\mathbf{x}_2 + \mathbf{g}(\mathbf{x}_1) + \mathbf{f}_{ric}\right) \tag{3}$$

and $\mathbf{d} \in \mathbb{R}^n$ is defined as

$$\mathbf{d} = \mathbf{M}(\mathbf{x}_1)^{-1}\boldsymbol{\tau}_e. \tag{4}$$

It should be noted that matrix $\mathbf{M}(\mathbf{x}_1)$ is full-rank, allowing the determination of its inverse matrix for any manipulator configuration.

Given the desired trajectory $\mathbf{x}_r$ and its derivatives $\dot{\mathbf{x}}_r$ (first derivative) and $\ddot{\mathbf{x}}_r$ (second derivative), the tracking error can be defined as $\mathbf{z}_1 = \mathbf{x}_1 - \mathbf{x}_r$ and $\mathbf{z}_2 = \mathbf{x}_2 - \dot{\mathbf{x}}_r$. Taking into account (2), the error dynamics can be described as

$$\begin{cases} \dot{\mathbf{z}}_1 = \mathbf{z}_2, \\ \dot{\mathbf{z}}_2 = \mathbf{f}(\mathbf{x}_1, \mathbf{x}_2) + \mathbf{M}^{-1}(\mathbf{x}_1)\boldsymbol{\tau} + \mathbf{d} - \ddot{\mathbf{x}}_r. \end{cases} \tag{5}$$

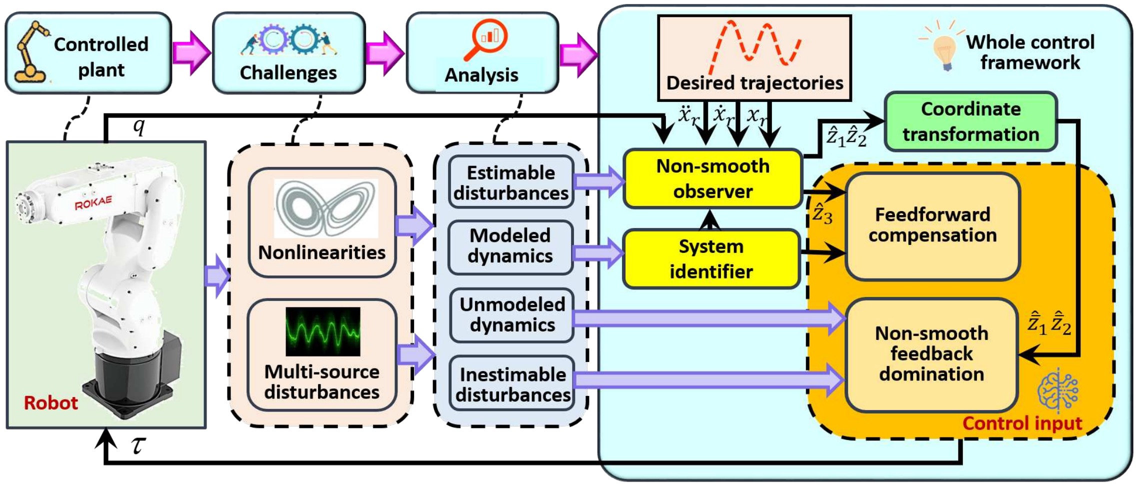

Fig. 1. Block scheme of systematic control framework, which mainly includes three parts: modelled dynamics identification, disturbance estimation and compensation as well as unmodelled dynamics and inestimable disturbance suppression. The system identifier provides the nominal model of robots. The non-smooth observer is used to estimate disturbances, which is compensated in the feedforward action. The non-smooth feedback controller is employed to suppress unmodeled dynamics and inestimable disturbances.

Following the concept of inverse dynamics control, also referred to as computed-torque control [3], the control law $\boldsymbol{\tau}$ can be expressed as

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{x}_1)(-\mathbf{f}(\mathbf{x}_1, \mathbf{x}_2) + \ddot{\mathbf{x}}_r - \mathbf{K}_p\mathbf{z}_1 - \mathbf{K}_d\mathbf{z}_2) \quad (6)$$

where $\mathbf{K}_p$ and $\mathbf{K}_d$ are diagonal positive definite matrices. Substituting (6) into (5) yields the differential equation for the position error dynamics:

$$\dot{\mathbf{z}}_2 + \mathbf{K}_d\mathbf{z}_2 + \mathbf{K}_p\mathbf{z}_1 = \mathbf{d}. \quad (7)$$

It should be noted that the implementation of this control protocol requires the online computation of the inertia matrix $\mathbf{M}(\mathbf{x}_1)$ and the nonlinear term $\mathbf{f}(\mathbf{x}_1, \mathbf{x}_2)$, as the control law is based on the current system state. However, this real-time calculation may impose significant constraints on the software and hardware architecture of the system. Therefore, it is natural to consider reducing the online computation burden associated with this control approach.

Furthermore, in (7), if $\mathbf{d} = \mathbf{0}$, the homogeneous error equation $\dot{\mathbf{z}}_2 + \mathbf{K}_d\mathbf{z}_2 + \mathbf{K}_p\mathbf{z}_1 = \mathbf{0}$ represents asymptotically stable error dynamics. While it is possible to obtain a nominal dynamic model of robot manipulators using advanced parameter identification techniques to achieve intact cancellation of dynamic terms, the model typically exhibits a level of uncertainty due to unmodeled dynamics, unknown external payloads, and imperfect knowledge of mechanical parameters. Consequently, full compensation is impractical, i.e., the position error will asymptotically converge to a bounded region.

B. Robust Inverse Dynamics Control

To enhance tracking accuracy, several robust inverse dynamics control (RIDC) methods were reported in [32], [33]. One such technique is the disturbance observer-based technique, which incorporates a nonlinear disturbance observer [34] into the inverse dynamics controller, thereby endowing the control system with robustness. The nonlinear disturbance observer is designed as follows:

$$\begin{cases} \hat{\mathbf{d}} = \mathbf{L}_d(\mathbf{x}_2 - \mathbf{P}), \\ \dot{\mathbf{P}} = \mathbf{M}(\mathbf{x}_1)^{-1}\boldsymbol{\tau} + \mathbf{f}(\mathbf{x}_1, \mathbf{x}_2) + \hat{\mathbf{d}} \end{cases} \quad (8)$$

where $\mathbf{L}_d = \mathbf{L}_d^{\mathrm{T}} > \mathbf{0}$ is the matrix gain of the nonlinear disturbance observer to be designed, $\mathbf{P}$ is the auxiliary state and $\hat{\mathbf{d}}$ is the estimation corresponding to $\mathbf{d}$. Subsequently, according to (6), the RIDC law is designed as

$$\boldsymbol{\tau} = \mathbf{M}(\mathbf{x}_1)(-\mathbf{f}(\mathbf{x}_1, \mathbf{x}_2) - \hat{\mathbf{d}} + \ddot{\mathbf{x}}_r - \mathbf{K}_p\mathbf{z}_1 - \mathbf{K}_d\mathbf{z}_2). \quad (9)$$

Compared to (6), the controller (9) incorporates an additional term, representing the estimated disturbance $\hat{\mathbf{d}}$ in the feedforward action. This illustrates that RIDC has the capability to compensate for the disturbance itself. Following a similar treatment as in (7), the error dynamics for RIDC can be formulated as follows:

$$\dot{\mathbf{z}}_2 + \mathbf{K}_d\mathbf{z}_2 + \mathbf{K}_p\mathbf{z}_1 = \tilde{\mathbf{d}} \quad (10)$$

where $\tilde{\mathbf{d}} = \mathbf{d} - \hat{\mathbf{d}}$.

A comparison between (7) and (10) reveals that RIDC exhibits the reduced position error compared to the standard IDC, as the magnitude of the term leading to tracking error is diminished. However, RIDC still necessitates online calculation of the feedforward compensation action. Consequently, it is highly desirable to design a control framework for robot manipulators that not only improves tracking accuracy and response time but also alleviates the burden of online computation. This serves as the motivation for our investigation into the corresponding solution.

III. Main Results

In this section, we present a systematic control framework for robot manipulators that satisfies the requirements of improving tracking accuracy, response time, and reducing online computation burden. The main components of this framework are described as follows:

- **Modelled Dynamics Identification:** System identification techniques (Section III-A) are employed to achieve precise cancellation control of the modelled dynamics for the feedforward compensation.
- **Disturbance Estimation and Compensation:** Non-smooth observer techniques (Section III-B) are utilized to estimate disturbances, which are then feedforward compensated.
- **Unmodelled Dynamics and Inestimable Disturbance Suppression:** Non-smooth feedback domination design (Section III-C) is adopted to suppress unmodelled dynamics and inestimable disturbances.

The resulting block scheme of the control framework is depicted in Fig. 1. Before delving into the details of this framework, we first present an assumption and some notations that will be used.

Assumption 1: The desired trajectory $\mathbf{x}_r$ and its derivatives $\dot{\mathbf{x}}_r$ and $\ddot{\mathbf{x}}_r$ are known to be continuous and bounded.

Notations:

- $\mathbf{x} = [x_1, x_2, \ldots, x_n]^{\mathrm{T}}$ denotes an n-dimensional vector, where x_i represents the ith component of $\mathbf{x}$ with $i = 1, 2, \ldots, n$.
- $\mathrm{sig}^r(\mathbf{x}) = \left[\lfloor x_1 \rceil^r, \lfloor x_2 \rceil^r, \ldots, \lfloor x_n \rceil^r\right]^{\mathrm{T}}$, where $\lfloor x_i \rceil^r \triangleq \mathrm{sign}(x_i)|x_i|^r$ with $r \geq 0$.
- $\langle a \rangle_N$ represents a saturation function with a threshold $N > 0$, given by:

$$\langle a \rangle_N = \begin{cases} \mathrm{sign}(a)N & \text{if } |a| > N \\ a & \text{if } |a| \leq N \end{cases}$$

- $\langle \mathbf{x} \rangle_N$ is defined as $\langle \mathbf{x} \rangle_N = \left[\langle x_1 \rangle_N, \langle x_2 \rangle_N, \ldots, \langle x_n \rangle_N\right]^{\mathrm{T}}$.

A. System Identification

The relationship between the motion of the robot manipulator and the joint torques can be described using the Newton-Euler formulation, as shown in (1). To begin with, we assume that there are no external disturbances ($\boldsymbol{\tau}_e = \mathbf{0}$). Based on the linearity in the parameters, we can express the dynamic model (1) in a linear form with a set of dynamic parameters:

$$\mathbf{M}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) + \mathbf{f}_{ric} = \mathbf{Y}_s(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\pi}_s = \boldsymbol{\tau} \quad (11)$$

where $\mathbf{Y}_s(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \in \mathbb{R}^{n \times s}$ is a regressor function of $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$, and $\boldsymbol{\pi}_s \in \mathbb{R}^s$ is a set of standard parameters. Each component of $\boldsymbol{\pi}_s$ typically corresponds to 14 standard parameters per joint (see [29] for details). However, note that not all 14 dynamic parameters per joint may appear in (11). To account for this, we can rewrite (11) in a more compact form by exploiting the column linear dependency of $\mathbf{Y}_s$ to reassemble the parameter set $\boldsymbol{\pi}_s$ [35] as:

$$\boldsymbol{\tau} = \mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}})\boldsymbol{\pi} \quad (12)$$

where $\mathbf{Y}(\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}) \in \mathbb{R}^{n \times r}$ represents a subset of the maximum linearly independent columns of $\mathbf{Y}_s$, and $\boldsymbol{\pi} \in \mathbb{R}^r$ is a set of base parameters. It is notable that the dimension of $\boldsymbol{\pi}_s$ and $\boldsymbol{\pi}$ satisfies $r \leq s = 14n$. If we have access to measurements of $\mathbf{q}, \dot{\mathbf{q}}, \ddot{\mathbf{q}}$, and $\boldsymbol{\tau}$ along an excitation trajectory at time instants $t_1, \ldots, t_N$, we can have

$$\mathcal{T} = \mathcal{Y}\boldsymbol{\pi} \quad (13)$$

where $\mathcal{T} = [\boldsymbol{\tau}(t_1)^{\mathrm{T}}\boldsymbol{\tau}(t_2)^{\mathrm{T}} \cdots \boldsymbol{\tau}(t_N)^{\mathrm{T}}]^{\mathrm{T}}$, $\mathcal{Y} = [\mathbf{Y}(t_1)^{\mathrm{T}}\mathbf{Y}(t_2)^{\mathrm{T}} \cdots \mathbf{Y}(t_N)^{\mathrm{T}}]^{\mathrm{T}}$. Computing (13) using a least-squares approach [36] gives us the solution:

$$\boldsymbol{\pi} = (\mathcal{Y}^{\mathrm{T}}\mathcal{Y})^{-1}\mathcal{Y}^{\mathrm{T}}\mathcal{T}. \quad (14)$$

This completes the process of parameter identification. For the computation of $\mathbf{M}(\mathbf{q})$, $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}}$, and $\mathbf{g}(\mathbf{q})$, please refer to [29] for detailed explanations.

B. Non-Smooth Observer Design

Although direct access to disturbances is not available, we can utilize the model knowledge obtained from the identification technique in subsection III-A to estimate them. Therefore, we construct a non-smooth observer based on (5) as follows:

$$\begin{cases} \dot{\hat{\mathbf{z}}}_1 = \hat{\mathbf{z}}_2 + \ell\mathbf{L}_1\mathrm{sig}^{m_2}(\mathbf{z}_1 - \hat{\mathbf{z}}_1), \\ \dot{\hat{\mathbf{z}}}_2 = \langle \hat{\mathbf{f}} \rangle_N + \mathbf{M}^{-1}(\mathbf{x}_1)\boldsymbol{\tau} + \hat{\mathbf{z}}_3 - \ddot{\mathbf{x}}_r \\ \qquad\qquad + \ell^2\mathbf{L}_2\mathrm{sig}^{m_3}(\mathbf{z}_1 - \hat{\mathbf{z}}_1), \\ \dot{\hat{\mathbf{z}}}_3 = \ell^3\mathbf{L}_3\mathrm{sig}^{m_4}(\mathbf{z}_1 - \hat{\mathbf{z}}_1) \end{cases} \quad (15)$$

where $\mathbf{L}_1, \mathbf{L}_2, \mathbf{L}_3$, and ℓ are the design parameters. The variables $\hat{\mathbf{z}}_1, \hat{\mathbf{z}}_2$, and $\hat{\mathbf{z}}_3$ represent estimations corresponding to $\mathbf{z}_1, \mathbf{z}_2$, and $\mathbf{d}$, respectively. The term $\langle \hat{\mathbf{f}} \rangle_N = \mathbf{f}(\langle \hat{\mathbf{x}}_1 \rangle_N, \langle \hat{\mathbf{x}}_2 \rangle_N)$, where $\hat{\mathbf{x}}_1 = \hat{\mathbf{z}}_1 + \mathbf{x}_r$ and $\hat{\mathbf{x}}_2 = \hat{\mathbf{z}}_2 + \dot{\mathbf{x}}_r$. The exponents $m_i = 1 + (i-1)\sigma$ for $(i = 2, 3, 4)$, with $\sigma \in (-\frac{1}{3}, 0)$.

It is worthy of noting that the non-smooth observer (15) is expected to outperform the nonlinear disturbance observer (8) due to the inclusion of the non-smooth term $\mathrm{sig}^{\alpha}(\cdot)$. This results in higher estimation accuracy and faster response. Furthermore, while the framework of the non-smooth observer may resemble the standard extended state observer (ESO) [18], they are fundamentally different. One notable distinction is that the non-smooth observer separately estimates nonlinear dynamics and disturbances, whereas ESO combines them into a single lumped estimate, which can lead to adverse effects. For instance, known nonlinear dynamics may exhibit large amplitudes during transient processes. To achieve better estimation performance, the gain of ESO needs to be increased, leading to noise amplification.

C. Non-Smooth Feedback Controller Design

With the availability of system states, disturbances, and nonlinear functions from system identification (Section III-A) and the non-smooth observer design (Section III-B), we can now proceed to propose the non-smooth controller to achieve the high-performance control objective. The detailed design procedure is explained below.

First, by denoting $\bar{\mathbf{z}}_1 = \mathbf{z}_1$ and $\bar{\mathbf{z}}_2 = \frac{\mathbf{z}_2}{\ell}$, then system (5) can be transformed into the following form:

$$\begin{cases} \dot{\bar{\mathbf{z}}}_1 = \ell\bar{\mathbf{z}}_2, \\ \dot{\bar{\mathbf{z}}}_2 = \dfrac{\mathbf{f}(\mathbf{x}_1, \mathbf{x}_2) + \mathbf{M}^{-1}(\mathbf{x}_1)\boldsymbol{\tau} + \mathbf{d} - \ddot{\mathbf{x}}_r}{\ell}. \end{cases} \quad (16)$$

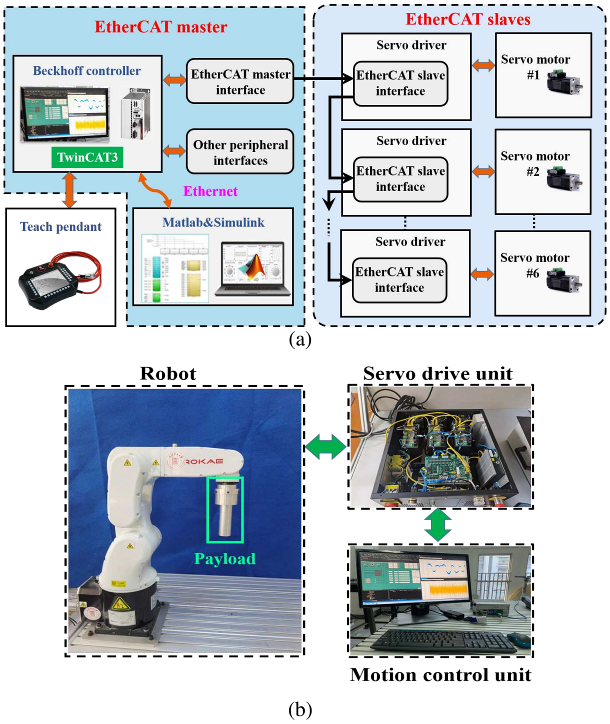

Fig. 2. Experimental platform. (a) System architecture. (b) Hardware diagram.

To enable offline pretreatment of feedforward dynamics, we replace $\mathbf{x}_1, \mathbf{x}_2$ in the nonlinear term $\mathbf{f}(\mathbf{x}_1, \mathbf{x}_2)$ with $\mathbf{x}_r, \dot{\mathbf{x}}_r$, denoted as $\tilde{\mathbf{f}} = \mathbf{f}(\mathbf{x}_r, \dot{\mathbf{x}}_r)$. Considering (15), the nonlinear feedforward compensation can be expressed as $\mathbf{M}\mathbf{u}^*$, where

$$\mathbf{u}^* = -\tilde{\mathbf{f}} - \hat{\mathbf{z}}_3 + \ddot{\mathbf{x}}_r. \tag{17}$$

Furthermore, the stabilizing nonlinear control action is designed as $\mathbf{M}\ell^2\mathbf{v}_c$, with

$$\mathbf{v}_c = -\mathbf{K}_1 \mathrm{sig}^{1+2\sigma}(\hat{\bar{\mathbf{z}}}_1) - \mathbf{K}_2 \mathrm{sig}^{\frac{1+2\sigma}{1+\sigma}}(\hat{\bar{\mathbf{z}}}_2) \tag{18}$$

where $\mathbf{K}_1$ and $\mathbf{K}_2$ are parameters to be designed, and $\hat{\bar{\mathbf{z}}}_1$ and $\hat{\bar{\mathbf{z}}}_2$ are intermediate states defined as $\hat{\bar{\mathbf{z}}}_1 = \hat{\mathbf{z}}_1$ and $\hat{\bar{\mathbf{z}}}_2 = \hat{\mathbf{z}}_2/\ell$.

Finally, combining (17) and (18), the non-smooth controller can be expressed as follows:

$$\boldsymbol{\tau} = \mathbf{M}(\ell^2\mathbf{v}_c + \mathbf{u}^*). \tag{19}$$

Before concluding this section, we provide a theorem of the main results:

Theorem 1: Considering the dynamic model (1) for the robotic system, under Assumption 1, the proposed non-smooth controller (17) - (19), in conjunction with dynamic parameter identification (14) and the non-smooth observer (15), guarantees that the trajectory tracking error of the closed-loop system will converge to a bounded region within a finite time.

Proof: Please refer to the Appendix for a detailed proof.

IV. Experimental Evaluations

This section presents several experiments conducted with a 6-DoF robot manipulator (the XB4 robot manufactured by ROKAE Technology Co., Ltd, as shown in Fig. 2) to illustrate the robustness and effectiveness of the proposed algorithm[1].

[1] Please refer to the supplementary material for a video demonstration of the evaluation details.

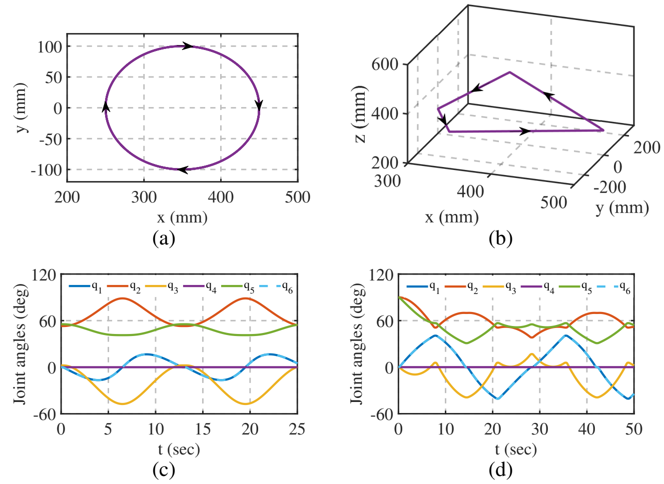

Fig. 3. Desired trajectories. (a) 2-D desired trajectory. (b) 3-D desired trajectory. (c) Joint trajectories of the 2-D curve. (d) Joint trajectories of the 3-D curve.

TABLE I
Control Gains Used on XB4 Robot

Controllers	Parameters
NSC	$\mathbf{L}_1 = \mathrm{diag}\{8,8,5,5,5,5\}$ $\mathbf{L}_2 = \mathrm{diag}\{50,50,40,40,40,40\}$ $\mathbf{L}_3 = \mathrm{diag}\{300,300,200,200,200,200\}$ $\ell = 10$ $\sigma = -0.1$ $\mathbf{K}_1 = \mathrm{diag}\{5,5,5,25,25,20\}$ $\mathbf{K}_2 = \mathrm{diag}\{1,1,1,15,15,15\}$
DSTC	$\mathbf{L}_d = \mathrm{diag}\{35,35,35,35,35,35\}$ $\mathbf{K}_1 = \mathrm{diag}\{50,50,50,100,100,100\}$ $\mathbf{K}_2 = \mathrm{diag}\{6,6,6,15,15,15\}$ $\mathbf{K}_3 = \mathrm{diag}\{25,25,25,50,50,50\}$
RIDC	$\mathbf{K}_p = \mathrm{diag}\{1000,1000,1000,10000,10000,10000\}$ $\mathbf{K}_d = \mathrm{diag}\{40,40,40,50,50,50\}$ $\mathbf{L}_d = \mathrm{diag}\{35,35,35,35,35,35\}$
IDC	$\mathbf{K}_p = \mathrm{diag}\{1000,1000,1000,10000,10000,10000\}$ $\mathbf{K}_d = \mathrm{diag}\{40,40,40,50,50,50\}$

Fig. 4. Payload on end-effector. (a) 0.25 kg. (b) 1.2 kg.

A. Experimental Testbed

The experimental testbed consists of three parts: the servo drive module, the mechanical body and the motion control module, as illustrated in Fig. 2. Specifically, the Beckhoff motion controller acts as the EtherCAT communication master, and the servo driver with EtherCAT communication function acts as the EtherCAT slaves, which is responsible for connecting with the servo motor to form a servo axis group. The servo driver operates in cyclic synchronous torque mode and is connected with the Beckhoff motion controller through EtherCAT serial topology to realize real-time high-speed data exchange. The mechanical body consists of six joints and a payload fixed to the manipulator's end-effector. In the motion control module, the proposed algorithm is implemented in

Matlab/Simulink via a model-based design (MBD) approach. The TE1400 module is used to import the C++ file generated by MBD into the Beckhoff controller, and the algorithm module is verified using TwinCAT software.

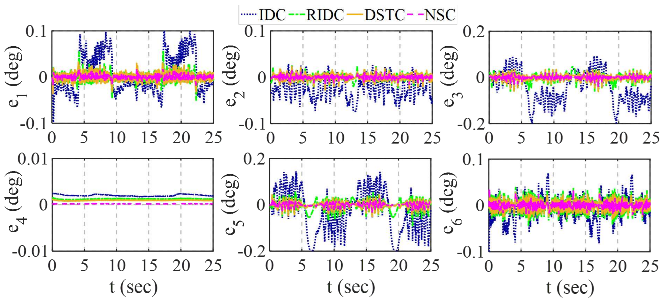

Fig. 5. Tracking errors over joint space using NSC, DSTC, RIDC, and IDC under Case I, respectively.

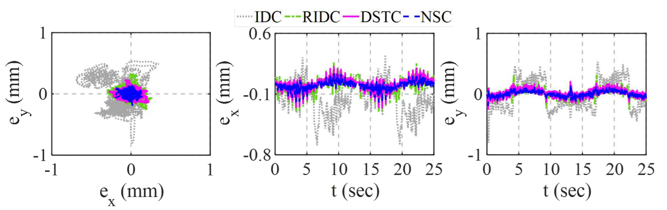

Fig. 6. Comparison of Cartesian space errors using NSC, DSTC, RIDC, and IDC under Case I, respectively.

TABLE II
CASE I: QUANTITATIVE RESULTS OF FOUR CONTROLLERS

	$\mathrm{MRMSE_J}(\mathrm{deg})$	$\mathrm{MRMSE_C}(\mathrm{mm})$
NSC	0.008	0.04
DSTC	0.01	0.06
RIDC	0.02	0.08
IDC	0.09	0.25

B. Experiment 1: Application in Tracking Control Tasks

In this experiment, various working conditions are considered to evaluate the tracking performance of the non-smooth control (NSC) algorithm. These conditions include tracking 2-D and 3-D curves under light-load and low-speed conditions, as well as high-speed and heavy-load conditions. Additionally, comparisons with vanilla inverse dynamics control (IDC) and robust inverse dynamics control (RIDC) under the same working conditions are also reported. Meanwhile, we also introduce the model-based super-twisting sliding mode control method reported in [37] for a deeper experimental comparison. To alleviate chattering, a nonlinear disturbance observer is introduced to this sliding mode control design. This composite control method is defined as DSTC, where the disturbance observer is designed as (8), and the DSTC law is designed as

$$\begin{cases} \boldsymbol{\tau} = \mathbf{M}(\mathbf{x}_1)(-\mathbf{f}(\mathbf{x}_1,\mathbf{x}_2) + \ddot{\mathbf{x}}_r - \mathbf{K}_2\mathrm{sig}^{\frac{1}{2}}(\mathbf{s})) \\ \qquad\qquad + \mathbf{M}(\mathbf{x}_1)(-\omega - \mathbf{K}_1\mathbf{z}_2 - \hat{\mathbf{d}}) \\ \dot{\omega} = \mathbf{K}_3\mathrm{sign}(\mathbf{s}) \end{cases} \tag{20}$$

with $\mathbf{K}_1$, $\mathbf{K}_2$, $\mathbf{K}_3$ being diagonal positive definite matrices and $\mathbf{s} = \mathbf{K}_1\mathbf{z}_1 + \mathbf{z}_2$ being the sliding surface. The parameters for NSC, DSTC, RIDC and IDC are summarized in Table I.

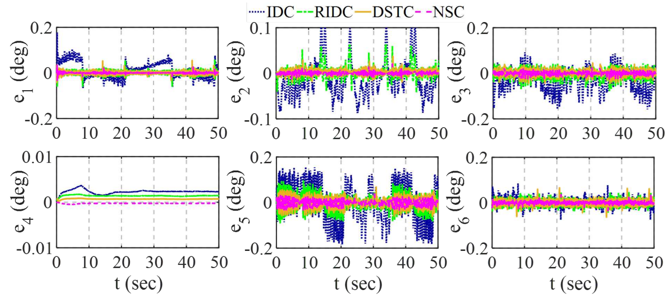

Fig. 7. Tracking errors over joint space using NSC, DSTC, RIDC, and IDC under Case II, respectively.

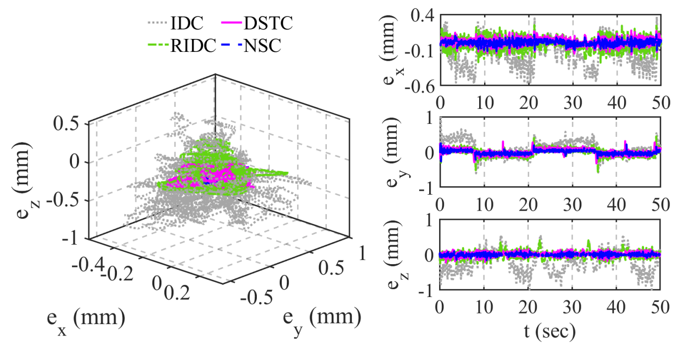

Fig. 8. Comparison of Cartesian space errors using NSC, DSTC, RIDC, and IDC under Case II, respectively.

TABLE III
CASE II: QUANTITATIVE RESULTS OF FOUR CONTROLLERS

	$\mathrm{MRMSE_J}(\mathrm{deg})$	$\mathrm{MRMSE_C}(\mathrm{mm})$
NSC	0.01	0.05
DSTC	0.02	0.09
RIDC	0.04	0.1
IDC	0.07	0.42

1) Case I: tracking 2-D curves under light load and low speed: In Case I, we perform tracking of 2-D curves under light load and low speed conditions. The 2-D curve is depicted in Fig. 3(a), and the corresponding joint angles are depicted in Fig. 3(c). Additionally, an extra payload with a weight of 0.25 kg is attached to the end-effector of the XB4 robot, as shown in Fig. 4(a). Fig. 5 illustrates the tracking error for the entire joint-space trajectory under the four control methods. The root mean square (RMS) performance index is adopted to quantitatively evaluate these control algorithms. For joint space, we computed RMS for each joint and then took the maximum. Here, we use $\mathrm{MRMSE_J}$ to define the performance index. As summarized in Table II, the performance index $\mathrm{MRMSE_J}$ is computed, resulting in values of 0.008 deg for NSC, 0.01 deg for DSTC, 0.02 deg for RIDC, and 0.09 deg for IDC. Since the robot performs tasks in Cartesian space, we further evaluate our method in Cartesian space. Similarly, the performance index is defined as $\mathrm{MRMSE_C}$ for Cartesian space. $\mathrm{MRMSE_C}$ values for NSC, DSTC, RIDC, and IDC are 0.04 mm, 0.06 mm, 0.08 mm, and 0.25 mm, respectively. Both Fig. 5 and Fig. 6 demonstrate that NSC achieves higher precision tracking compared to DSTC, RIDC and IDC.

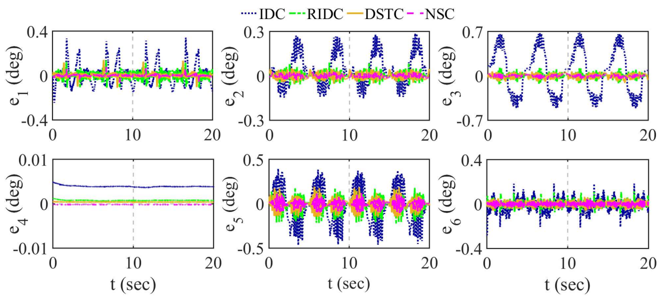

Fig. 9. Tracking errors over joint space using NSC, DSTC, RIDC, and IDC under Case III, respectively.

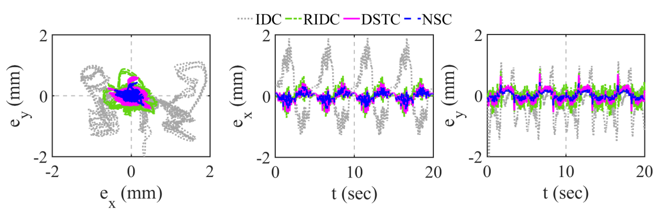

Fig. 10. Comparison of Cartesian space errors using NSC, DSTC, RIDC, and IDC under Case III, respectively.

TABLE IV
CASE III: QUANTITATIVE RESULTS OF FOUR CONTROLLERS

	$\mathrm{MRMSE_J(deg)}$	$\mathrm{MRMSE_C(mm)}$
NSC	0.02	0.09
DSTC	0.05	0.17
RIDC	0.04	0.25
IDC	0.36	0.84

2) Case II: tracking 3-D curves under light load and low speed: Similar to Case I, we evaluate the NSC method in tracking desired 3-D curves, as shown in Fig. 3(b), with their corresponding mapping in joint space illustrated in Fig. 3(d). Figs. 7-8 show the tracking error results in joint space and Cartesian space. As shown in Table III, in joint space, the maximum RMS components for NSC, DSTC, RIDC, and IDC are 0.01 deg, 0.02 deg, 0.04 deg, and 0.07 deg, respectively. In Cartesian space, these values are 0.05 mm for NSC, 0.09 mm for DSTC, 0.1 mm for RIDC, and 0.42 mm for IDC.

3) Case III: tracking 2-D curves under heavy load and high speed: In this working condition, we increased the payload to 1.2 kg (see Fig. 4(b)) and tripled the joint velocities compared to Case I. Figs. 9 and 10 demonstrate that NSC is more robust compared to DSTC, RIDC and IDC. The tracking error of RIDC and IDC is significantly larger, as observed when comparing Figs. 5-6 and Figs. 9-10. The corresponding numerical results in joint space are 0.02 deg for NSC, 0.05 deg for DSTC, 0.04 deg for RIDC, and 0.36 deg for IDC, while in Cartesian space, the values are 0.09 mm for NSC, 0.17 mm for DSTC, 0.25 mm for RIDC, and 0.84 mm for IDC, as shown in Table IV.

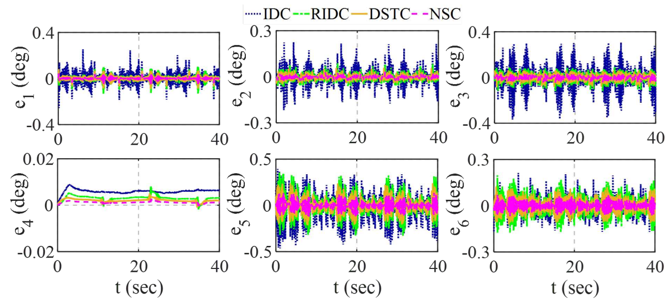

Fig. 11. Tracking errors over joint space using NSC, DSTC, RIDC, and IDC under Case IV, respectively.

4) Case IV: tracking 3-D curves under heavy load and high speed: In this case, we change the tracking trajectory to a 3-D curve compared to Case III. The relevant results are depicted in Figs. 11-12. The relevant numerical analysis results in joint space are 0.03 deg for NSC, 0.05 deg for DSTC, 0.08 deg for RIDC, and 0.12 deg for IDC, while in Cartesian space, the values are 0.11 mm for NSC, 0.18 mm for DSTC, 0.32 mm for RIDC, and 0.43 mm for IDC, as shown in Table V.

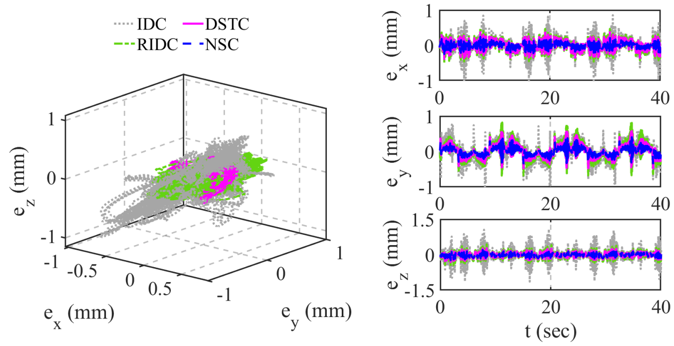

Fig. 12. Comparison of Cartesian space errors using NSC, DSTC, RIDC, and IDC under Case IV, respectively.

TABLE V
CASE IV: QUANTITATIVE RESULTS OF FOUR CONTROLLERS

	$\mathrm{MRMSE_J(deg)}$	$\mathrm{MRMSE_C(mm)}$
NSC	0.03	0.11
DSTC	0.05	0.18
RIDC	0.08	0.32
IDC	0.12	0.43

C. Experiment 2: Repeatability Test

We make a further evaluation on the proposed control method by conducting a repeatability test. We choose the 2-D position $(449.95, -0.03)$ mm of the robotic end-effector as the target position and repeat this operation by six times. As shown in Fig. 13, the green '⋆' represents the target location, the black solid '⋄' corresponds to the actual locations of the robot, the red solid '∘' denotes the center point of the six real positions, and the black dotted circle represents the smallest circle that encloses the total actual positions. Moreover, the target position is denoted by 'a' and the repeatability is denoted by 'r'. It is obvious that r is approximately equal to 0.015 mm, satisfying the requirements of repeatability.

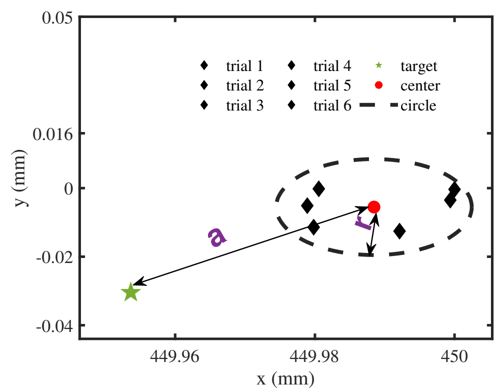

Fig. 13. Repeatability test result.

V. Conclusion

This paper has presented a systematic trajectory tracking control framework for robot manipulators, specifically designed to address the challenges of estimation and tracking issues using only position signals. The proposed control architecture ensures a high level of tracking performance within the closed-loop system. The effectiveness of the approach was validated through a series of tests, including the tracking of 2-D and 3-D trajectories at varying speeds and loads, as well as repeatability tests for individual points. Comparative experiments with standard IDC, RIDC and DSTC clearly demonstrate that the proposed approach outperforms them in terms of precision, robustness, and reliability. Looking ahead, our future work will focus on enhancing the safety of robot operations and developing an efficient fault-tolerant controller to handle situations where position sensors are absent.

Appendix

To begin with, the essential lemma is given for better explanation in Section III.

Lemma 1: [38] Let $a, b > 0$. For any $c > 0$, gives $\forall u, v \in \mathbb{R} : |u|^a|v|^b \leq \frac{a}{a+b}c|u|^{a+b} + \frac{b}{a+b}c^{\frac{-a}{b}}|v|^{a+b}$.

Proof: Define $\mathbf{e}_1 = \mathbf{z}_1 - \hat{\mathbf{z}}_1$, $\mathbf{e}_2 = \frac{\mathbf{z}_2 - \hat{\mathbf{z}}_2}{\ell}$, $\mathbf{e}_3 = \frac{\mathbf{d} - \hat{\mathbf{z}}_3}{\ell^2}$. Then, according to (5) and (15), the estimation error dynamics can be obtained by

$$\begin{cases} \dot{\mathbf{e}}_1 = \ell(\mathbf{e}_2 - \mathbf{L}_1 \mathrm{sig}^{m_2}(\mathbf{e}_1)), \\ \dot{\mathbf{e}}_2 = \ell(\mathbf{e}_3 - \mathbf{L}_2 \mathrm{sig}^{m_3}(\mathbf{e}_1)) + \ell^{-1}(\mathbf{f} - \langle \hat{\mathbf{f}} \rangle_N), \\ \dot{\mathbf{e}}_3 = -\ell \mathbf{L}_3 \mathrm{sig}^{m_4}(\mathbf{e}_1) + \dfrac{\dot{\mathbf{d}}}{\ell^2}. \end{cases} \tag{21}$$

Denote $\mathbf{e} = \left[\mathbf{e}_1^{\mathrm{T}}, \mathbf{e}_2^{\mathrm{T}}, \mathbf{e}_3^{\mathrm{T}}\right]^{\mathrm{T}}$, then we can reformulate (21) as the following compact form

$$\dot{\mathbf{e}} = \ell \boldsymbol{\Psi} + \boldsymbol{\Lambda} + \boldsymbol{\Xi} \tag{22}$$

where $\boldsymbol{\Psi} = \begin{bmatrix} \boldsymbol{\Psi}_1 \\ \boldsymbol{\Psi}_2 \\ \boldsymbol{\Psi}_3 \end{bmatrix}$, $\boldsymbol{\Lambda} = \begin{bmatrix} \boldsymbol{\Lambda}_1 \\ \boldsymbol{\Lambda}_2 \\ \boldsymbol{\Lambda}_3 \end{bmatrix}$, $\boldsymbol{\Xi} = \begin{bmatrix} \boldsymbol{\Xi}_1 \\ \boldsymbol{\Xi}_2 \\ \boldsymbol{\Xi}_3 \end{bmatrix}$, $\boldsymbol{\Psi}_1 = \mathbf{e}_2 - \mathbf{L}_1 \mathrm{sig}^{m_2}(\mathbf{e}_1)$, $\boldsymbol{\Psi}_2 = \mathbf{e}_3 - \mathbf{L}_2 \mathrm{sig}^{m_3}(\mathbf{e}_1)$, $\boldsymbol{\Psi}_3 = -\mathbf{L}_3 \mathrm{sig}^{m_4}(\mathbf{e}_1)$, $\boldsymbol{\Lambda}_1 = \mathbf{0}$, $\boldsymbol{\Lambda}_2 = \frac{\mathbf{f} - \langle \hat{\mathbf{f}} \rangle_N}{\ell}$, $\boldsymbol{\Lambda}_3 = \mathbf{0}$, $\boldsymbol{\Xi}_1 = \mathbf{0}$, $\boldsymbol{\Xi}_2 = \mathbf{0}$, $\boldsymbol{\Xi}_3 = \frac{\dot{\mathbf{d}}}{\ell^2}$.

Consider a Lyapunov function as

$$V_e = (\lfloor \mathbf{e} \rceil_\kappa^{1-\frac{\sigma}{2}})^{\mathrm{T}} \mathbf{P}_1 \lfloor \mathbf{e} \rceil_\kappa^{1-\frac{\sigma}{2}} \tag{23}$$

where $\kappa = (\underbrace{1, \ldots, 1}_{n}, \underbrace{1+\sigma, \ldots, 1+\sigma}_{n}, \underbrace{1+2\sigma, \ldots, 1+2\sigma}_{n})$, $\lfloor \mathbf{e} \rceil_\kappa^{1-\frac{\sigma}{2}} = \left[(\mathrm{sig}^{\frac{1-\frac{\sigma}{2}}{1}}(\mathbf{e}_1))^{\mathrm{T}}, (\mathrm{sig}^{\frac{1-\frac{\sigma}{2}}{1+\sigma}}(\mathbf{e}_2))^{\mathrm{T}}, (\mathrm{sig}^{\frac{1-\frac{\sigma}{2}}{1+2\sigma}}(\mathbf{e}_3))^{\mathrm{T}}\right]^{\mathrm{T}}$, and $\mathbf{P}_1$ is a positive definite and symmetrical matrix satisfying $\mathbf{A}_1^{\mathrm{T}} \mathbf{P}_1 + \mathbf{P}_1 \mathbf{A}_1 = -\mathbf{I}$ with $\mathbf{A}_1 = \begin{bmatrix} -\mathbf{L}_1 & \mathbf{I} & \mathbf{0} \\ -\mathbf{L}_2 & \mathbf{0} & \mathbf{I} \\ -\mathbf{L}_3 & \mathbf{0} & \mathbf{0} \end{bmatrix}$.

Computing the time derivative of V_e along (22) gives

$$\dot{V}_e = \ell \sum_{i=1}^{3} \sum_{j=1}^{n} \frac{\partial V_e}{\partial \mathbf{e}_{i,j}} \boldsymbol{\Psi}_{i,j} + \sum_{j=1}^{n} \frac{\partial V_e}{\partial \mathbf{e}_{2,j}} \boldsymbol{\Lambda}_{2,j} + \sum_{j=1}^{n} \frac{\partial V_e}{\partial \mathbf{e}_{3,j}} \boldsymbol{\Xi}_{3,j}. \tag{24}$$

In light of the Homogeneity theory [31], V_e is essentially homogeneous of degree $2-\sigma$, expressed as $V_e \in \mathcal{H}^{2-\sigma}$. In addition, it can also be concluded that $(\sum_{i=1}^{3} \sum_{j=1}^{n} \frac{\partial V_e}{\partial \mathbf{e}_{i,j}} \boldsymbol{\Psi}_{i,j}) \in \mathcal{H}^2$, $\frac{\partial V_e}{\partial \mathbf{e}_{2,j}} \in \mathcal{H}^{1-2\sigma}$, $\boldsymbol{\Lambda}_{2,j} \in \mathcal{H}^{1+2\sigma}$, $\frac{\partial V_e}{\partial \mathbf{e}_{3,j}} \in \mathcal{H}^{1-3\sigma}$. subsequently, referring to [39], one can conclude that $(\ell \sum_{i=1}^{3} \sum_{j=1}^{n} \frac{\partial V_e}{\partial \mathbf{e}_{i,j}} \boldsymbol{\Psi}_{i,j} + \sum_{j=1}^{n} \frac{\partial V_e}{\partial \mathbf{e}_{2,j}} \boldsymbol{\Lambda}_{2,j}) \in \mathcal{H}^2 \implies (\ell \sum_{i=1}^{3} \sum_{j=1}^{n} \frac{\partial V_e}{\partial \mathbf{e}_{i,j}} \boldsymbol{\Psi}_{i,j} + \sum_{j=1}^{n} \frac{\partial V_e}{\partial \mathbf{e}_{2,j}} \boldsymbol{\Lambda}_{2,j}) \leq -(\alpha \ell - \breve{\alpha}) V_e^{\frac{2}{2-\sigma}}$ with $\alpha > 0$ and $\breve{\alpha} > 0$.

As for the third item in (24), we can employ Lemma 1 to obtain that

$$\begin{aligned} \sum_{j=1}^{n} \frac{\partial V_e}{\partial \mathbf{e}_{3,j}} \boldsymbol{\Xi}_{3,j} &\leq \sum_{j=1}^{n} (|\frac{\partial V_e}{\partial \mathbf{e}_{3,j}}|^{\frac{2}{1-3\sigma}})^{\frac{1-3\sigma}{2}} (\gamma^{\frac{2}{1+3\sigma}})^{\frac{1+3\sigma}{2}} \\ &\leq \sum_{j=1}^{n} \frac{1-3\sigma}{2} |\frac{\partial V_e}{\partial \mathbf{e}_{3,j}}|^{\frac{2}{1-3\sigma}} + 3(1+3\sigma)\gamma^{\frac{2}{1+3\sigma}} \\ &\leq \acute{\alpha} V_e^{\frac{2}{2-\sigma}} + \Delta_1 \end{aligned} \tag{25}$$

where $\gamma = \sup|\boldsymbol{\Xi}_{3,j}|$ for $j = 1, \ldots, n$, $\acute{\alpha} \geq 1 - 3\sigma$ and $\Delta_1 \geq 3(1+3\sigma)\gamma^{\frac{2}{1+3\sigma}}$.

Combining (24) and (25), we have

$$\dot{V}_e \leq -(\alpha \ell - \alpha^\star) V_e^{\frac{2}{2-\sigma}} + \Delta_1 \tag{26}$$

where $\alpha^\star = \breve{\alpha} + \acute{\alpha}$. Thus, it can be concluded that the system state estimation error will converge to a bounded region $\Omega_0 = \left\{\mathbf{e} \mid V_e(\mathbf{e}) \leq (\frac{\Delta_1}{\alpha \ell - \alpha^\star})^{\frac{2-\sigma}{2}}\right\}$ within a finite time.

Subsequently, substituting (19) into (16) yields

$$\begin{cases} \dot{\bar{\mathbf{z}}}_1 = \ell\bar{\mathbf{z}}_2, \\ \dot{\bar{\mathbf{z}}}_2 = \ell\mathbf{v}_c + \dfrac{\mathbf{f}-\tilde{\mathbf{f}}+\mathbf{d}-\hat{\mathbf{z}}_3}{\ell} \\ \quad = \ell(\mathbf{v}_c - \mathbf{v}_s + \mathbf{v}_s) + \dfrac{\mathbf{f}-\tilde{\mathbf{f}}+\mathbf{d}-\hat{\mathbf{z}}_3}{\ell} \\ \quad = \ell\mathbf{v}_s + \ell(\mathbf{v}_c - \mathbf{v}_s) + \dfrac{\mathbf{f}-\tilde{\mathbf{f}}+\mathbf{d}-\hat{\mathbf{z}}_3}{\ell} \end{cases} \tag{27}$$

where $\mathbf{v}_s = -\mathbf{K}_1\mathrm{sig}^{1+2\sigma}(\bar{\mathbf{z}}_1) - \mathbf{K}_2\mathrm{sig}^{\frac{1+2\sigma}{1+\sigma}}(\bar{\mathbf{z}}_2)$.

Denote $\bar{\mathbf{z}} = \left[\bar{\mathbf{z}}_1^{\mathrm{T}}, \bar{\mathbf{z}}_2^{\mathrm{T}}\right]^{\mathrm{T}}$. Then, rewriting (27) leads to

$$\dot{\bar{\mathbf{z}}} = \ell\Big(\mathbf{A}\bar{\mathbf{z}} + \begin{bmatrix}\mathbf{0}\\ \mathbf{I}\end{bmatrix}\mathbf{v}_s + \begin{bmatrix}\mathbf{0}\\ \mathbf{I}\end{bmatrix}(\mathbf{v}_c - \mathbf{v}_s)\Big) + \begin{bmatrix}\mathbf{0}\\ \frac{\mathbf{f}-\tilde{\mathbf{f}}+\mathbf{d}-\hat{\mathbf{z}}_3}{\ell}\end{bmatrix} \tag{28}$$

with $\mathbf{A} = \begin{bmatrix}\mathbf{0} & \mathbf{I}\\ \mathbf{0} & \mathbf{0}\end{bmatrix}$.

Consider a Lyapunov function candidate similar to (23) as

$$V_c = (\lfloor\bar{\mathbf{z}}\rceil_{\kappa_1}^{1-\frac{\sigma}{2}})^{\mathrm{T}}\mathbf{P}_2\lfloor\bar{\mathbf{z}}\rceil_{\kappa_1}^{1-\frac{\sigma}{2}} \tag{29}$$

with $\kappa_1 = (\underbrace{1,\ldots,1}_{n}, \underbrace{1+\sigma,\ldots,1+\sigma}_{n})$, $\mathbf{P}_2$ is a positive definite and symmetrical matrix satisfying $(\mathbf{A}-\mathbf{H}_2)^{\mathrm{T}}\mathbf{P}_2 + \mathbf{P}_2(\mathbf{A}-\mathbf{H}_2) = -\mathbf{I}$ with $\mathbf{H}_2 = \begin{bmatrix}\mathbf{0} & \mathbf{0}\\ \mathbf{K}_1 & \mathbf{K}_2\end{bmatrix}$.

Take the derivative of V_c, where the first two items are treated similarly to (24), so we have

$$\begin{aligned} \dot{V}_c \leq & -(\alpha\ell - \bar{\alpha}_1)V_c^{\frac{2}{2-\sigma}} + \frac{\partial V_c}{\partial\bar{\mathbf{z}}_2^{\mathrm{T}}}\ell(\mathbf{v}_c - \mathbf{v}_s) \\ & + \frac{\partial V_c}{\partial\bar{\mathbf{z}}_2^{\mathrm{T}}}\frac{\mathbf{f}-\tilde{\mathbf{f}}+\mathbf{d}-\hat{\mathbf{z}}_3}{\ell} \end{aligned} \tag{30}$$

with $\bar{\alpha}_1 > 0$.

From the theoretical analysis in [39], we can obtain

$$\frac{\partial V_c}{\partial\bar{\mathbf{z}}_2^{\mathrm{T}}}\ell(\mathbf{v}_c - \mathbf{v}_s) \leq \frac{\alpha}{2}\ell V_c^{\frac{2}{2-\sigma}} + \tilde{\alpha}\ell V_e^{\frac{2}{2-\sigma}}, \tag{31}$$

$$\frac{\partial V_c}{\partial\bar{\mathbf{z}}_2^{\mathrm{T}}}\frac{\mathbf{f}-\tilde{\mathbf{f}}+\mathbf{d}-\hat{\mathbf{z}}_3}{\ell} \leq \acute{\alpha}_1 V_c^{\frac{2}{2-\sigma}} + \Delta_2 \tag{32}$$

where $\tilde{\alpha}$, $\acute{\alpha}_1$ and Δ_2 are positive constants.

Combining (30), (31) and (32), we have

$$\dot{V}_c \leq -(\frac{\alpha}{2}\ell - \bar{\alpha})V_c^{\frac{2}{2-\sigma}} + \tilde{\alpha}\ell V_e^{\frac{2}{2-\sigma}} + \Delta_2 \tag{33}$$

with $\bar{\alpha} = \bar{\alpha}_1 + \acute{\alpha}_1$.

Construct a Lyapunov function candidate as

$$V = \frac{r_0 V_c}{r_0 + 1 - V_c} + \frac{\beta\mu V_e}{\mu + 1 - V_e} \tag{34}$$

where r_0, β and μ are positive constants and need to fulfill some necessary conditions. Following the results in [39], we can further obtain

$$\dot{V} \leq -\gamma_1(V_c^{\frac{2}{2-\sigma}} + V_e^{\frac{2}{2-\sigma}}) + \Delta \tag{35}$$

where $\gamma_1 > 0$ and $\Delta > 0$. Therefore, it can be deduced that the trajectory tracking error of the closed-loop system will converge to a region near zero in a finite time [39].

This completes the proof. ■

References

[1] Y. Huang and D. Caldwell, "A linearly constrained nonparametric framework for imitation learning," in *IEEE International Conference on Robotics and Automation*, 2020, pp. 4400–4406.

[2] Y. Zhu, J. Qiao, and L. Guo, "Adaptive sliding mode disturbance observer-based composite control with prescribed performance of space manipulators for target capturing," *IEEE Transactions on Industrial Electronics*, vol. 66, no. 3, pp. 1973–1983, 2018.

[3] M. Spong, S. Hutchinson, and M. Vidyasagar, *Robot modeling and control*. John Wiley & Sons, 2020.

[4] J. Slotine and W. Li, "Adaptive manipulator control: A case study," *IEEE Transactions on Automatic Control*, vol. 33, no. 11, pp. 995–1003, 1988.

[5] E. Agbaraji, H. Inyiama, and C. Okezie, "Dynamic modeling of a 3-dof articulated robotic manipulator based on independent joint scheme," *Physical Science International Journal*, pp. 1–10, 2017.

[6] Y. Zhang, D. Kim, Y. Zhao, and J. Lee, "PD control of a manipulator with gravity and inertia compensation using an RBF neural network," *International Journal of Control, Automation and Systems*, vol. 18, no. 12, pp. 3083–3092, 2020.

[7] A. De Luca and F. Flacco, "A PD-type regulator with exact gravity cancellation for robots with flexible joints," in *2011 IEEE International Conference on Robotics and Automation*, 2011, pp. 317–323.

[8] J. Reher and A. Ames, "Inverse dynamics control of compliant hybrid zero dynamic walking," in *IEEE International Conference on Robotics and Automation*, 2021, pp. 2040–2047.

[9] L. Righetti, J. Buchli, M. Mistry, and S. Schaal, "Inverse dynamics control of floating-base robots with external constraints: A unified view," in *IEEE International Conference on Robotics and Automation*, 2011, pp. 1085–1090.

[10] Q. Xu and Z. Cao, "Piezoelectric positioning control with output-based discrete-time terminal sliding mode control," *IET Control Theory & Applications*, vol. 11, no. 5, pp. 694–702, 2017.

[11] C. Hua, Y. Yang, and P. Liu, "Output-feedback adaptive control of networked teleoperation system with time-varying delay and bounded inputs," *IEEE/ASME Transactions on Mechatronics*, vol. 20, no. 5, pp. 2009–2020, 2014.

[12] M. Rayguru, R. Mohan, R. Parween, L. Yi, A. Le, and S. Roy, "An output feedback based robust saturated controller design for pavement sweeping self-reconfigurable robot," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 3, pp. 1236–1247, 2021.

[13] T. Yang, N. Sun, Y. Fang, X. Xin, and H. Chen, "New adaptive control methods for n-link robot manipulators with online gravity compensation: Design and experiments," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 1, pp. 539–548, 2021.

[14] Y. Su, P. Muller, and C. Zheng, "A global asymptotic stable output feedback pid regulator for robot manipulators," in *IEEE International Conference on Robotics and Automation*, 2007, pp. 4484–4489.

[15] J. Back and H. Shim, "Adding robustness to nominal output-feedback controllers for uncertain nonlinear systems: A nonlinear version of disturbance observer," *Automatica*, vol. 44, no. 10, pp. 2528–2537, 2008.

[16] C. Johnson, "Optimal control of the linear regulator with constant disturbances," *IEEE Transactions on Automatic Control*, vol. 13, no. 4, pp. 416–421, 1968.

[17] K. Ohishi, M. Nakao, K. Ohnishi, and K. Miyachi, "Microprocessor-controlled dc motor for load-insensitive position servo system," *IEEE Transactions on Industrial Electronics*, no. 1, pp. 44–49, 1987.

[18] J. Han, "From pid to active disturbance rejection control," *IEEE Transactions on Industrial Electronics*, vol. 56, no. 3, pp. 900–906, 2009.

[19] H. Sira-Ramirez and M. Oliver-Salazar, "On the robust control of buck-converter dc-motor combinations," *IEEE Transactions on Power Electronics*, vol. 28, no. 8, pp. 3912–3922, 2012.

[20] Y. Oh and W. Chung, "Disturbance-observer-based motion control of redundant manipulators using inertially decoupled dynamics," *IEEE/ASME Transactions on Mechatronics*, vol. 4, no. 2, pp. 133–146, 1999.

[21] W.-H. Chen, D. Ballance, P. Gawthrop, and J. O'Reilly, "A nonlinear disturbance observer for robotic manipulators," *IEEE Transactions on Industrial Electronics*, vol. 47, no. 4, pp. 932–938, 2000.

[22] R.-D. Xi, X. Xiao, T.-N. Ma, and Z.-X. Yang, "Adaptive sliding mode disturbance observer based robust control for robot manipulators towards assembly assistance," *IEEE Robotics and Automation Letters*, vol. 7, no. 3, pp. 6139–6146, 2022.

[23] S. Sinan, Y. Kali, R. Fareh, M. Saad, and M. Bettayeb, "Extended state observer-based synergetic control for n-dof robot manipulators using joint position measurements only," *Advanced Robotics*, vol. 37, no. 12, pp. 766–778, 2023.

[24] K. Shojaei, A. Kazemy, and A. Chatraei, "An observer-based neural adaptive PID^2 controller for robot manipulators including motor dynamics with a prescribed performance," *IEEE/ASME Transactions on Mechatronics*, vol. 26, no. 3, pp. 1689–1699, 2020.

[25] X. Yu, W. He, H. Li, and J. Sun, "Adaptive fuzzy full-state and output-feedback control for uncertain robots with output constraint," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 51, no. 11, pp. 6994–7007, 2020.

[26] J. Mao, J. Yang, X. Liu, S. Li, and Q. Li, "Modeling and robust continuous tsm control for an inertially stabilized platform with couplings," *IEEE Transactions on Control Systems Technology*, vol. 28, no. 6, pp. 2548–2555, 2019.

[27] S. Li, H. Liu, and S. Ding, "A speed control for a pmsm using finite-time feedback control and disturbance compensation," *Transactions of the Institute of Measurement and Control*, vol. 32, no. 2, pp. 170–187, 2010.

[28] S. Li, H. Du, and X. Lin, "Finite-time consensus algorithm for multi-agent systems with double-integrator dynamics," *Automatica*, vol. 47, no. 8, pp. 1706–1712, 2011.

[29] L. Han, J. Mao, P. Cao, Y. Gan, and S. Li, "Toward sensorless interaction force estimation for industrial robots using high-order finite-time observers," *IEEE Transactions on Industrial Electronics*, vol. 69, no. 7, pp. 7275–7284, 2021.

[30] T. Dierks and S. Jagannathan, "Neural network output feedback control of robot formations," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 40, no. 2, pp. 383–399, 2009.

[31] C. Zhang and J. Yang, *Nonrecursive Control Design for Nonlinear Systems: Theory and Applications*. CRC Press, 2023.

[32] M. Spong, J. Thorp, and J. Kleinwaks, "Robust microprocessor control of robot manipulators," *Automatica*, vol. 23, no. 3, pp. 373–379, 1987.

[33] Y. Liu, T. Li, J. Duan, X. Wu, H. Wang, Q. Fan, J. Lin, and Y. Hu, "On a hierarchical adaptive and robust inverse dynamic control strategy with experiment for robot manipulators under uncertainties," *Control Engineering Practice*, vol. 138, p. 105604, 2023.

[34] W.-H. Chen, J. Yang, L. Guo, and S. Li, "Disturbance-observer-based control and related methods—an overview," *IEEE Transactions on Industrial Electronics*, vol. 63, no. 2, pp. 1083–1095, 2015.

[35] M. Gautier and W. Khalil, "Direct calculation of minimum set of inertial parameters of serial robots," *IEEE Transactions on robotics and Automation*, vol. 6, no. 3, pp. 368–373, 1990.

[36] J. Jin and N. Gans, "Parameter identification for industrial robots with a fast and robust trajectory design approach," *Robotics and Computer-Integrated Manufacturing*, vol. 31, pp. 21–29, 2015.

[37] H. Saied, A. Chemori, M. Bouri, M. Rafei, and C. Francis, "Feedforward super-twisting sliding mode control for robotic manipulators: Application to pkms," *IEEE Transactions on Robotics*, vol. 39, no. 4, pp. 3167–3184, 2023.

[38] C. Qian and W. Lin, "A continuous feedback approach to global strong stabilization of nonlinear systems," *IEEE Transactions on Automatic Control*, vol. 46, no. 7, pp. 1061–1079, 2001.

[39] C. Zhang, J. Yang, C. Wen, L. Wang, and S. Li, "Realization of exact tracking control for nonlinear systems via a nonrecursive dynamic design," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 50, no. 2, pp. 577–589, 2017.

Linyan Han received her Ph.D. degree in Control Science and Engineering from the School of Automation, Southeast University, Nanjing, China, in 2022.

She is currently a postdoctoral researcher at the School of Mechanical Engineering, University of Leeds. Her interests include force control, nonlinear control theory and their applications to robotic systems.

Jianliang Mao (Member, IEEE) received the B.Sc. degree in automation, the M.Sc. degree in control engineering, and the Ph.D. degree in control theory and control engineering from the School of Automation, Southeast University, Nanjing, China, in 2011, 2014, and 2018, respectively.

From 2018 to 2021, he was with the Research and Development Institute, Estun Automation Co., Ltd. He was selected into "Jiangsu Province entrepreneurship and innovation plan" in 2019. He is currently with the College of Automation Engineering, Shanghai University of Electric Power, Shanghai, China. His research interests include model predictive control, sliding mode control, and vision based interactive control and their applications to electric drives and robot manipulators.

Chuanlin Zhang (Senior Member, IEEE) received the B.Sc. degree in mathematics and the Ph.D. degree in control theory and control engineering from the School of Automation, Southeast University, Nanjing, China, in 2008 and 2014, respectively.

From 2011 to 2012, he was a Visiting Ph.D. Student with the Department of Electrical and Computer Engineering, University of Texas at San Antonio, San Antonio, TX, USA. From 2016 to 2017, he was a Visiting Scholar with the Energy Research Institute, Nanyang Technological University, Singapore. From 2017 to 2018, he was a Visiting Scholar with Advanced Robotics Center, National University of Singapore. Since 2014, he has been with the College of Automation Engineering, Shanghai University of Electric Power, Shanghai, where he is currently a Professor. His research interests include nonlinear system control theory and applications for power systems.

Robert W. Kay received a Ph.D. degree in Electrical and Electronic Engineering from Heriot-Watt University, Edinburgh, U.K., in 2007.

Robert commercialized his doctoral research through the spinout company MicroStencil where he spent 7 years as CTO. In 2012 Robert started his first academic post as Senior Lecturer in Additive Manufacturing at Loughborough University, UK. Robert joined the University of Leeds in 2016 as an Associate Professor and was made full Professor in 2022.

Robert C. Richardson received a B.Eng. degree in mechatronics and a Ph.D. degree from the University of Leeds, Leeds, U.K., in 1997 and 2001, respectively.

He is a Professor of Robotics, in the School of Mechanical Engineering, University of Leeds. He is Director of the Real Robotics Lab, and Institute of Design, Robotics and Optimisation (iDRO) at the University of Leeds. He is the executive chair of the EPSRC UK-RAS network and the Innovation Director for the University of Leeds spinout company. His research interests cover a broad range of applied robotics including robotics for civil infrastructure inspection and repair, making smart bodies for smart robots, and robotics for 3D printing applications.

Chengxu Zhou received his Ph.D. degree in Robotics from the Italian Institute of Technology, Genova, Italy, in 2016.

He is a Lecturer in Mobile Robotics at the School of Mechanical Engineering, University of Leeds. His research focuses on developing intelligent motion generation for legged robots, with a particular emphasis on dynamic motion planning and whole-body control of articulated robots using optimization & machine learning technologies.