

Article

Intrinsic Plasticity-Driven Neuroadaptive Asymptotic Tracking Control for a Class of Uncertain Robotic Manipulators

Qing Chen, Xiangyang Tan, Shuaicheng Hou, Meiyi Qing and Zuojin Li * 

School of Electronic and Electrical Engineering, Chongqing University of Science and Technology, Chongqing 401331, China; 2021049@cqust.edu.cn (Q.C.); 2024204028@cqust.edu.cn (X.T.); shuaichenghou@cqust.edu.cn (S.H.); qingmeiyi323@126.com (M.Q.)

* Correspondence: 2010014@cqust.edu.cn

Abstract

This paper proposes a novel neuroadaptive asymptotic tracking control method for a class of uncertain multi-input multi-output (MIMO) robotic manipulators. Firstly, an intrinsic plasticity (IP)-driven cycle echo state network (ESN) is constructed. The intrinsic plasticity mechanism can adaptively adjust neuronal excitability, enhancing the network's ability to capture complex time-varying dynamics. Meanwhile, the cycle reservoir structure significantly reduces the number of neural connections, thus improving computational efficiency. Secondly, the proposed IP-driven cycle ESN is integrated with the robust integral of the sign of the error (RISE) framework to form a neuroadaptive controller. The IP-driven cycle ESN serves to dynamically approximate the unknown nonlinearities inherent in the robotic system, whereas the RISE term compensates for approximation errors and external disturbances to ensure satisfactory robust performance. Then, a rigorous stability analysis is given to demonstrate that the proposed controller can achieve asymptotic convergence of the tracking error. Finally, simulation experiments are conducted on two typical two-joint manipulators to evaluate the performance of the proposed method. Comparative results demonstrate that, in contrast to the Radial Basis Function Neural Network (RBFNN)-based PI control method, the proposed method achieves faster error convergence rate, higher tracking precision, and smoother control inputs. The results highlight the effectiveness of combining the approximation capability of the IP-driven cycle ESN with the robust compensation capability of the RISE framework for high-precision control of uncertain nonlinear robotic systems.

Keywords: echo state network; intrinsic plasticity; asymptotic tracking control; cycle reservoir; RISE



Academic Editor: Asanka Perera

Received: 12 May 2026

Revised: 18 July 2026

Accepted: 20 July 2026

Published: 22 July 2026

Copyright: © 2026 by the authors.

Licensee MDPI, Basel, Switzerland.

This article is an open access article distributed under the terms and conditions of the [Creative Commons Attribution \(CC BY\)](https://creativecommons.org/licenses/by/4.0/) license.

1. Introduction

With the rapid advancement of industrial automation, intelligent manufacturing, medical rehabilitation, and service robotics, robotic systems have become deeply integrated into modern production and daily life [1]. However, in practical applications, these systems are inevitably subjected to complex uncertainties, including unmodeled dynamics, intrinsic parameter variations, and external time-varying disturbances. Traditional control strategies often struggle to maintain satisfactory control performance in the presence of these complex uncertainties. Consequently, achieving high-precision trajectory tracking under dynamic, uncertain conditions remains a critical challenge. To address this issue, a variety of advanced control strategies have been extensively investigated. For instance,

semi-model-free adaptive backstepping dynamic sliding mode control was introduced to handle uncertain manipulator dynamics [2]. Similarly, the fixed-time control method has been deployed to mitigate lumped disturbances and actuator faults [3], while global integral terminal sliding mode approaches have been developed to guarantee finite-time convergence under unknown parameters [4]. Despite their success, the majority of existing schemes remain constrained by the requirement of known system structures or predefined bounding functions. To circumvent these limitations, neural network (NN)-based control methods have attracted increasing attention in recent years, particularly in the field of robot control. Owing to their powerful nonlinear approximation capability, NNs are uniquely suited to reconstruct unknown dynamics in complex robotic systems [5,6]. Meanwhile, their inherent self-learning adaptability provides an effective framework for capturing unmodeled dynamics and real-time parameter variations [7,8]. By leveraging these features, NN-based controllers exhibit remarkable robustness against severe coupling effects and external disturbances [9–11], making them a cornerstone of modern intelligent robotic control.

Although considerable progress has been achieved, existing NN-based control schemes still face two major challenges when applied to complex robotic systems. On the one hand, most adaptive NN control methods can only guarantee Uniformly Ultimately Bounded (UUB) tracking performance rather than true asymptotic convergence of the tracking error. This implies that the system states converge merely to a finite neighborhood of the origin instead of approaching zero strictly, which limits their applicability in high-precision trajectory tracking tasks. On the other hand, the majority of existing approaches rely on feedforward neural network architectures, such as the RBFNN [12,13]. These networks provide only static input–output mappings and thus lack the capability to capture time-varying internal dynamics inherent to robotic systems [14,15]. Consequently, their approximation performance deteriorates in dynamically changing environments. Furthermore, RBFNN-based controllers often depend on offline training and manual parameter tuning, leading to slow adaptation, poor generalization, and degraded control accuracy under time-varying conditions [12]. Although some recent control methods in the literature are based on echo state networks, a type of recurrent neural network, these methods adopt random or complicated network topologies, leading to high computational complexity and limited practical applicability in real robotic systems [16,17].

The primary objective of this work is to bridge the gap between tracking precision and computational efficiency in the control of highly uncertain, dynamic MIMO robotic systems, thereby overcoming the limitations of conventional UUB tracking performance and high network complexity. To achieve this objective, a novel neuroadaptive asymptotic tracking control strategy is proposed by integrating an intrinsic plasticity-driven cycle echo state network with the robust integral of the sign of the error framework. Compared with existing control methods in the literature, the main contributions of this work are threefold:

- First, a novel recurrent reservoir featuring a simplified cycle structure and biological learning rules is constructed to enhance representation capability while minimizing computational overhead. Unlike conventional ESNs that utilize random or complex reservoir topologies which heavily increase the computational burden and limit real-time applicability, the reservoir in our scheme adopts a deterministic, simple cycle structure. This design significantly reduces the density of neural interconnections, effectively alleviating the computational burden and lowering algorithmic complexity to facilitate faster real-time calculation. Furthermore, to prevent any degradation in approximation performance due to the simplified structure, an intrinsic plasticity mechanism—a foundational biological learning rule—is embedded for online tuning of the excitability of each reservoir neuron [18,19]. This drastically improves the

reservoir's dynamic representation capability to accurately capture the highly coupled, complex, and time-varying internal nonlinearities inherent in robotic systems.

- Second, a composite neuroadaptive controller is synthesized to achieve robust approximation of unknown dynamics and disturbance rejection capabilities. The controller is developed by employing the proposed IP-driven cycle ESN to dynamically approximate unknown system nonlinearities, while leveraging the RISE framework to continuously compensate for residual approximation errors and external time-varying disturbances, thereby guaranteeing high-precision tracking.
- Third, a rigorous theoretical proof and two simulation scenarios are provided. A comprehensive stability analysis based on Lyapunov theory is carried out to provide a solid mathematical guarantee for the asymptotic stability of the closed-loop system. Furthermore, two simulations are conducted on typical two-joint manipulators to evaluate the performance of the proposed method over conventional RBFNN-based strategies in balancing tracking precision, convergence speed, and control smoothness.

It should be noted that the scope of this work is limited to numerical simulation studies. The proposed control strategy is validated only under smooth continuous reference trajectories. Practical issues, including experimental validation and trajectory tracking under abrupt or discontinuous reference trajectories, are beyond the scope of this paper and will be investigated in future work.

The remainder of this manuscript is structured as follows. Section 2 establishes the problem formulation of the uncertain multi-joint robotic systems and briefly reviews the preliminaries regarding the topological architecture and universal approximation capability of the proposed IP-driven cycle ESN. Section 3 presents the detailed design procedure of the composite neuroadaptive controller and its corresponding weight update laws, followed by a rigorous mathematical proof of the closed-loop stability and asymptotic tracking performance. In Section 4, two numerical simulation scenarios are executed on different two-joint robotic manipulators to demonstrate the effectiveness and comparative advantages of the developed scheme. Section 5 provides a thorough discussion of the simulation results and the underlying limitations. Finally, Section 6 concludes the paper with a summary of key findings and prospects for future research.

2. Problem Formulation and Preliminaries

2.1. Problem Formulation

Consider a class of fully actuated n -link rigid robotic systems modeled via the classical Euler–Lagrange formula [20]:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d(t) = u \quad (1)$$

where $q \in \mathbb{R}^n$ is the joint angular displacement vector, $M(q) \in \mathbb{R}^{n \times n}$ is the inertia matrix, $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$ denotes the centripetal and Coriolis terms, $G(q) \in \mathbb{R}^n$ represents the gravitational vector, $F(\dot{q}) \in \mathbb{R}^n$ signifies the friction force, $\tau_d(t) \in \mathbb{R}^n$ is the unknown external disturbance, and $u \in \mathbb{R}^n$ is the control input. To facilitate the controller design, the inherent structural properties of the dynamic model (1) must be considered. Specifically, the inertia matrix $M(q)$ is symmetric and positive-definite, satisfying $m_1 \|x\|^2 \leq x^T M(q)x \leq m_2 \|x\|^2$ for any arbitrary vector $x \in \mathbb{R}^n$, where m_1 and m_2 are positive constants. In the subsequent stability analysis, this specific property provides a solid foundation to guarantee the positive-definiteness of the formulated Lyapunov function.

A neuroadaptive asymptotic tracking control strategy is proposed in this paper to ensure that: ① the joint angular displacement q of system (1) asymptotically tracks the desired trajectory q_d ; and ② all signals are bounded in the closed-loop system.

To achieve the above objectives, the following assumptions are made.

Assumption 1. The joint angular displacement q and joint angular velocity $\dot{q}$ of system (1) are available for measurement.

Assumption 2. The desired trajectory q_d and its first, second and third derivatives are known, bounded and continuous functions.

Assumption 3. The system matrices $M(q)$, $C(q, \dot{q})$, $G(q)$, and $F(\dot{q})$ are assumed to be smooth. Meanwhile, the external disturbance $\tau_d(t)$ and its first two time derivatives are bounded by some constants $\varepsilon_{\tau 0}$, $\varepsilon_{\tau 1}$, $\varepsilon_{\tau 2}$, satisfying $\|\tau_d\| \leq \varepsilon_{\tau 0}$, $\|\dot{\tau}_d\| \leq \varepsilon_{\tau 1}$, and $\|\ddot{\tau}_d\| \leq \varepsilon_{\tau 2}$.

Hereafter, for the sake of simplicity, the function arguments are occasionally omitted when there is no risk of ambiguity, that is: $f(t)$ refers to any function vector of t and f is the same to $f(t)$.

Then, define the tracking error e_1 and the filtering error r

$$e_1 = q_d - q \quad (2)$$

$$e_2 = \dot{e}_1 + \alpha_1 e_1 \quad (3)$$

$$r = \dot{e}_2 + \alpha_2 e_2 \quad (4)$$

where $\alpha_1 > 0.5$ and $\alpha_2 > 0.5$ are design parameters.

By multiplying both sides of Equation (4) by $M(q)$, and then substituting Equations (2) and (3), it can be found that

$$\begin{aligned} M(q)r &= M(\ddot{e}_1 + \alpha_1 \dot{e}_1 + \alpha_2 e_2) \\ &= M(\alpha_1 \dot{e}_1 + \alpha_2 e_2 + \ddot{q}_d) \\ &\quad + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d - u \\ &= Q(\cdot) - u \end{aligned} \quad (5)$$

where $Q(\cdot) = M(q)(\alpha_1 \dot{e}_1 + \alpha_2 e_2 + \ddot{q}_d) + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d$ lumps all the model-dependent terms.

Differentiating Equation (5) with respect to time gives

$$M(q)\dot{r} = -\dot{M}(q)r - \dot{u} + \dot{Q}(\cdot).$$

By splitting the term $-\dot{M}(q)r$ into two equal parts and adding and subtracting e_2 , the above equation can be rewritten as

$$\begin{aligned} M\dot{r} &= -\frac{1}{2}\dot{M}r - \dot{u} - e_2 + \dot{Q}(\cdot) - \frac{1}{2}\dot{M}r + e_2 \\ &= -\frac{1}{2}\dot{M}r - \dot{u} - e_2 + N(\cdot) \end{aligned} \quad (6)$$

where $N(\cdot) = \dot{Q}(\cdot) - \frac{1}{2}\dot{M}r + e_2$ represents the remaining unknown time-varying terms that can be approximated by the neural network as illustrated in Section 2.2.

2.2. Intrinsic Plasticity-Driven Cycle Echo State Network and Its Approximating Capability

In Equation (6), the term $N(\cdot)$ includes the inertia matrix $M(q)$, the Coriolis and centrifugal force term $C(q, \dot{q})$, the gravitational vector $G(q)$, the friction force $F(\dot{q})$ and the external disturbance τ_d . These unknown terms, as they cannot be accurately modeled

or compensated by traditional control methods, make the control task challenging. To address this challenge, in this paper, an intrinsic plasticity-driven cycle echo state network is proposed to approximate these complex and time-varying nonlinear functions. In this structure, the cycle reservoir topology significantly reduces the number of connections, while the intrinsic plasticity rule online adapts each neuron's gain and bias parameters. This combination not only enhances the reservoir's dynamic representation capability but also improves computational efficiency, making it particularly suitable for real-time robotic control under uncertainties.

The architecture of the proposed IP-driven cycle ESN is illustrated in Figure 1. The network structure consists of three components: input layer, reservoir and readout layer. The computational procedure is as follows:

$$h_{in}(t) = W_{in}^T z(t) + W_{res}^T h(t-1) \quad (7)$$

$$h(t) = \phi\left(a_j(t)h_{in}(t) + b_j(t)\right) \quad (8)$$

$$y_{out}(t) = W_{out}^T h(t) \quad (9)$$

$$a_j(t+1) = a_j(t) + \frac{\eta}{a_j(t)} + h_{in_j}(t)\Delta b_j(t) \quad (10)$$

$$b_j(t+1) = b_j(t) + \eta\left(1 - \left(2 + \frac{1}{\mu}\right)h_j(t) + \frac{1}{\mu}h_j^2(t)\right) \quad (11)$$

where $z(t) \in \mathbb{R}^{A \times 1}$ is the input of the ESN, $h_{in}(t) \in \mathbb{R}^{B \times 1}$ is the input of the neurons in the reservoir, $h(t) \in \mathbb{R}^{B \times 1}$ is the reservoir output state vector at the current moment, $W_{in} \in \mathbb{R}^{A \times B}$ is the weight from input to the reservoir, $W_{res} \in \mathbb{R}^{B \times B}$ is the weight of the reservoir layer, $W_{out} \in \mathbb{R}^{B \times C}$ is the weight from the reservoir to the readout and $y_{out} \in \mathbb{R}^{C \times 1}$ is the ESN output. The variables A , B and C denote the neuron counts in the input, reservoir and readout components, respectively. The function ϕ is the activation function of the reservoir component, and the sigmoid function is adopted in this work. Therefore, it yields $\|\phi\| \leq \varepsilon_{\phi 0}$ and $\|\dot{\phi}\| \leq \varepsilon_{\phi 1}$, where $\varepsilon_{\phi 0}$ and $\varepsilon_{\phi 1}$ are some positive constants. Parameters $a_j(t)$ and $b_j(t)$ represent the gain and bias of the activation function of the j -th neuron in the reservoir at time t , respectively. μ is the expected distribution mean, η is the intrinsic plasticity learning rate, $h_{in_j}(t)$ is the input to the activation function of the j -th neuron in the reservoir at time t , $h_j(t)$ is the reservoir output of the j -th neuron in the reservoir, and $\Delta b_j(t) = b_j(t) - b_j(t-1)$. In this paper, the readout weight matrix W_{out} is obtained by the adaptive laws, while the input layer weight matrix W_{in} and the reservoir weight matrix W_{res} are set as follows:

$$W_{in} = \begin{bmatrix} \nu & \nu & \cdots & \nu \\ \nu & \nu & \cdots & \nu \\ \vdots & \vdots & \ddots & \vdots \\ \nu & \nu & \cdots & \nu \end{bmatrix}_{A \times B}, W_{res} = \begin{bmatrix} 0 & \cdots & 0 & \rho \\ \rho & \cdots & 0 & 0 \\ \vdots & \ddots & \vdots & \vdots \\ 0 & \cdots & \rho & 0 \end{bmatrix}_{B \times B}$$

where ν and ρ are some positive constants. It is worth mentioning that unlike conventional echo state networks that suffer from heavy computational burdens due to randomized or fully connected topologies, the proposed deterministic cycle structure establishes a highly decoupled and sparse state propagation mechanism. This structural simplification directly translates into a significant reduction in algorithmic complexity during the online control process. To further illustrate the advantages of this design from a computational perspective, the following remark is provided.

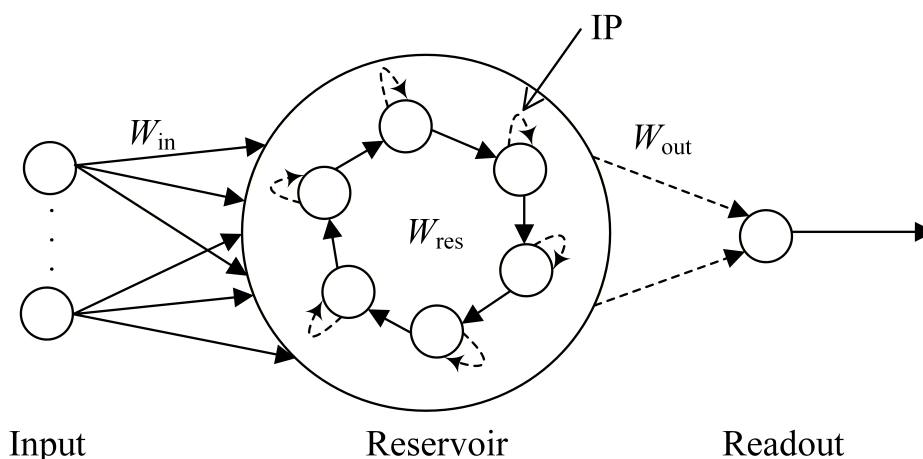


Figure 1. The architecture of the IP-driven cycle ESN.

Remark 1. Let N denote the number of internal neurons within the reservoir. For a fully connected reservoir, the internal weight matrix $W_{res} \in \mathbb{R}^{N \times N}$ is dense. Updating reservoir states involves full matrix-vector multiplication, leading to a quadratic computational complexity of $\mathcal{O}(N^2)$ per control step. In contrast, the cycle reservoir adopts a deterministic cycle topology with a highly sparse weight matrix, where non-zero elements only lie on the sub-diagonal and the corner entry. Benefiting from this cycle connection, the state of each neuron is only related to its adjacent neuron. The original matrix-vector multiplication is simplified into N scalar operations, and the computational complexity is reduced to $\mathcal{O}(N)$ per control step. Meanwhile, the number of effective connections decreases from N^2 to N . This significant reduction greatly lowers the computational overhead, making the proposed algorithm well applicable to high-frequency real-time robotic control under limited hardware resources.

According to the Universal Approximation Theorem for neural networks, given any compact set $\Omega_N \in \mathbb{R}$, for all $z \in \Omega_N$, $N(\cdot)$ can be presented by the ESN as

$$N(\cdot) = W^T \phi(z) + \varepsilon \quad (12)$$

where W is the output weight and ε is approximation error, which satisfies the assumption that $\|\varepsilon\| \leq \varepsilon_{n0}$, ε_{n0} is an unknown positive constant. Moreover, in this work, it is assumed that $\|\dot{\varepsilon}\| \leq \varepsilon_{n1}$, ε_{n1} is an unknown positive constant.

Then Equation (6) can be re-expressed as

$$M(q)\dot{r} = -\frac{1}{2}\dot{M}(q)r + W^T \phi(z) + \varepsilon - \dot{u} - e_2. \quad (13)$$

It is worth mentioning that as W is unknown, then the approximation of $N(\cdot)$ by ESN can be redefined as

$$\hat{N}(\cdot) = \hat{W}^T \phi(z) \quad (14)$$

where $z(t) = [q^T(t), \dot{q}^T(t), q_d^T(t), \dot{q}_d^T(t), \ddot{q}_d^T(t), e_1^T(t), e_2^T(t), \dot{e}_1^T(t)]^T \in \mathbb{R}^{8n}$, $\hat{W}$ is the estimated value of W .

3. Controller Design and Stability Analysis

Based on the system dynamics in (13), the control scheme is given by

$$u = (k_s + 1)e_2(t) - (k_s + 1)e_2(0) + \int_0^t \left[(k_s + 1)\alpha_2 e_2(\tau) + \beta \operatorname{sgn}(e_2(\tau)) + \hat{W}^T \phi(z(\tau)) \right] d\tau \quad (15)$$

where k_s and β are some positive constants. The overall architecture of the proposed controller is illustrated in Figure 2.

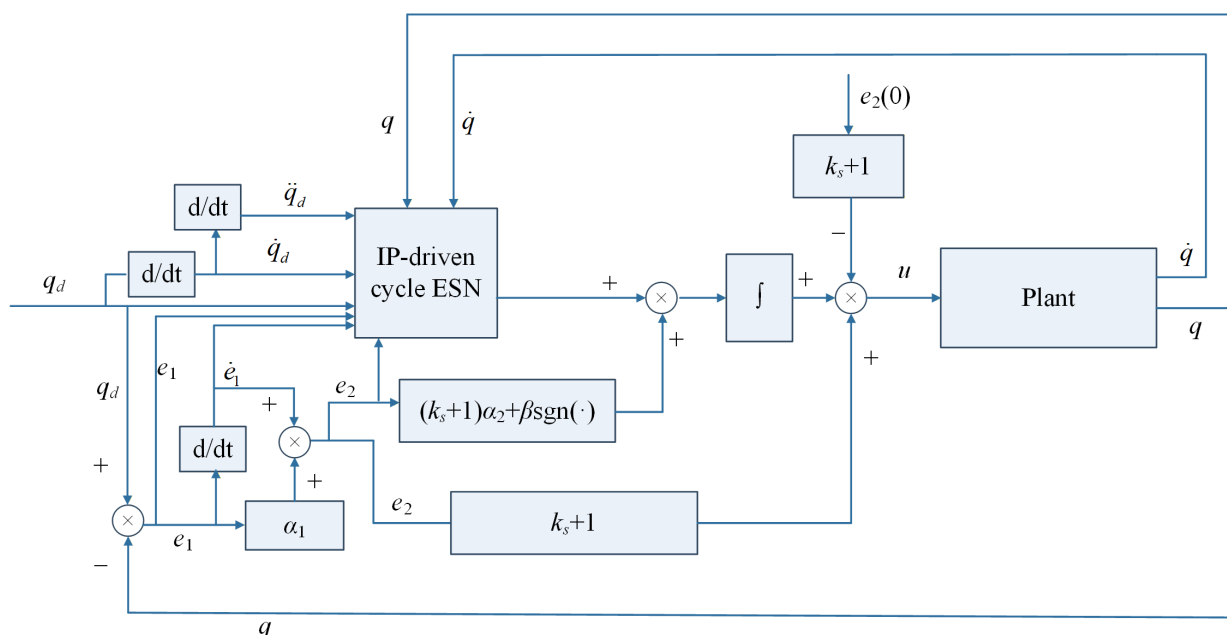


Figure 2. Overall architecture of the proposed control scheme.

The corresponding adaptive law for $\hat{W}$ is given as

$$\dot{\hat{W}} = k_n \alpha_2 \phi e_2^T - k_w k_n \|e_2\| \hat{W} \quad (16)$$

where k_n and k_w are positive designed parameters. The first term $k_n \alpha_2 \phi e_2^T$ is used for neural compensation of the error dynamics, while the second term $-k_w k_n \|e_2\| \hat{W}$ is introduced to prevent parameter drift and ensure boundedness of the weight estimates under approximation errors and external disturbances.

Substitute Equation (15) into Equation (13), and the close-loop tracking dynamics of the system can be obtained:

$$M(q)\dot{r} = -\frac{1}{2}\dot{M}(q)r - (k_s + 1)r - \beta \operatorname{sgn}(e_2) - \tilde{W}^T \phi + \varepsilon - e_2 \quad (17)$$

where $\tilde{W} = \hat{W} - W$ is the weight estimation error.

To facilitate the stability analysis, the following Lemma 1 and Lemma 2 are introduced.

Lemma 1. Give an auxiliary function $L(t)$ defined as

$$L(t) = r^T (\varepsilon - \beta \operatorname{sgn}(e_2)) + \frac{1}{4} k_w W_n^2 e_2^T \operatorname{sgn}(e_2) - \dot{e}_2^T H \quad (18)$$

where W_n is the upper bound of $\|W\|$, $H = \tilde{W}^T \phi$. H and its time derivative $\dot{H}$ are bounded [21]. Specifically, there exist positive constants ε_{h0} and ε_{h1} satisfying $\|H\| \leq \varepsilon_{h0}$ and $\|\dot{H}\| \leq \varepsilon_{h1}$. Provided that the control gain β is chosen to fulfill the following sufficient condition,

$$\beta > \varepsilon_{n0} + \max \left\{ \varepsilon_{h0}, \frac{1}{4\alpha_2} k_w W_n^2 + \frac{1}{\alpha_2} \varepsilon_{n1} + \frac{1}{\alpha_2} \varepsilon_{h1} \right\} \quad (19)$$

then it follows that

$$\int_0^t L(\tau) d\tau \leq \zeta \quad (20)$$

where $\zeta = \beta \|e_2(0)\|_1 - e_2^T(0)(\varepsilon(0) - H(0))$.

Proof. After expanding $L(t)$ and integrating, it is observed that

$$\int_0^t L(\tau) d\tau = \int_0^t r^T(\tau) (\varepsilon(\tau) - \beta \operatorname{sgn}(e_2(\tau))) d\tau + \frac{1}{4} k_w W_n^2 \int_0^t e_2^T(\tau) \operatorname{sgn}(e_2(\tau)) d\tau - \int_0^t \dot{e}_2^T(\tau) H(\tau) d\tau \quad (21)$$

Substituting (4) into the above equation and rearranging terms leads to

$$\begin{aligned} \int_0^t L(\tau) d\tau &= \int_0^t \alpha_2 e_2^T(\tau) (\varepsilon(\tau) - \beta \operatorname{sgn}(e_2(\tau))) d\tau + \int_0^t \dot{e}_2^T(\tau) (\varepsilon(\tau) - \beta \operatorname{sgn}(e_2(\tau))) d\tau \\ &\quad + \frac{1}{4} k_w W_n^2 \int_0^t e_2^T(\tau) \operatorname{sgn}(e_2(\tau)) d\tau - \int_0^t \dot{e}_2^T(\tau) H(\tau) d\tau \\ &= \int_0^t \alpha_2 e_2^T(\tau) (\varepsilon(\tau) - \beta \operatorname{sgn}(e_2(\tau)) + \frac{1}{4\alpha_2} k_w W_n^2 \operatorname{sgn}(e_2(\tau))) d\tau \\ &\quad + \int_0^t \dot{e}_2^T(\tau) (\varepsilon(\tau) - H(\tau)) - \beta \operatorname{sgn}(e_2(\tau)) d\tau \end{aligned} \quad (22)$$

By applying integration by parts to the second term of the above equation, it follows that

$$\begin{aligned} \int_0^t L(\tau) d\tau &= \int_0^t \alpha_2 e_2^T(\tau) (\varepsilon(\tau) - \beta \operatorname{sgn}(e_2(\tau)) + \frac{1}{4\alpha_2} k_w W_n^2 \operatorname{sgn}(e_2(\tau))) d\tau \\ &\quad + (e_2^T(\tau)(\varepsilon(\tau) - H(\tau))) \Big|_0^t - \int_0^t (e_2^T(\tau) \dot{\varepsilon}(\tau) - \dot{e}_2^T(\tau) H(\tau)) d\tau - \beta \int_0^t \dot{e}_2^T \operatorname{sgn}(e_2) d\tau \\ &= \int_0^t \alpha_2 e_2^T(\tau) (\varepsilon(\tau) - \beta \operatorname{sgn}(e_2(\tau)) + \frac{1}{4\alpha_2} k_w W_n^2 \operatorname{sgn}(e_2(\tau))) d\tau \\ &\quad + e_2^T(t) (\varepsilon(t) - H(t)) - e_2^T(0) (\varepsilon(0) - H(0)) \\ &\quad - \int_0^t (e_2^T(\tau) \dot{\varepsilon}(\tau) - \dot{e}_2^T(\tau) H(\tau)) d\tau - \beta \|e_2(t)\|_1 + \beta \|e_2(0)\|_1 \end{aligned} \quad (23)$$

Utilizing the Cauchy–Schwarz inequality and the relationship $\|\cdot\| \leq \|\cdot\|_1$, and further collecting like terms, it follows that

$$\begin{aligned} \int_0^t L(\tau) d\tau &\leq \int_0^t \alpha_2 \|e_2(\tau)\| \left(\|\varepsilon(\tau)\| - \beta + \frac{1}{4\alpha_2} k_w W_n^2 + \frac{\|\dot{\varepsilon}(\tau)\|}{\alpha_2} + \frac{\|\dot{H}(\tau)\|}{\alpha_2} \right) d\tau \\ &\quad + \|e_2(t)\|_1 (\|\varepsilon(t)\| + \|H(t)\| - \beta) - e_2^T(0) (\varepsilon(0) - H(0)) + \beta \|e_2(0)\|_1 \end{aligned} \quad (24)$$

Given $\|\varepsilon\| \leq \varepsilon_{n0}$, $\|\dot{\varepsilon}\| \leq \varepsilon_{n1}$, $\|H\| \leq \varepsilon_{h0}$, and $\|\dot{H}\| \leq \varepsilon_{h1}$ it follows that:

$$\begin{aligned} \int_0^t L(\tau) d\tau &\leq \int_0^t \alpha_2 \|e_2(\tau)\| \left(\varepsilon_{n0} - \beta + \frac{1}{4\alpha_2} k_w W_n^2 + \frac{\varepsilon_{n1}}{\alpha_2} + \frac{\varepsilon_{h1}}{\alpha_2} \right) d\tau \\ &\quad + \|e_2(t)\|_1 (\varepsilon_{n0} + \varepsilon_{h0} - \beta) - e_2^T(0) (\varepsilon(0) - H(0)) + \beta \|e_2(0)\|_1 \end{aligned} \quad (25)$$

Finally, when β is chosen to satisfy (19), inequality (20) follows directly. $\square$

Lemma 2 ([22]). Consider the nonlinear time-varying system

$$\dot{\zeta} = f(\zeta, t), \quad (26)$$

where $\zeta \in \mathbb{R}^m$ and $f : \mathbb{R}^m \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}^m$ is continuous, ensuring the existence of a unique solution. Define the region

$$\mathcal{D} := \{\zeta \in \mathbb{R}^m : \|\zeta\| < \varsigma\}, \quad (27)$$

where ς is a positive constant. Suppose that there exists a continuously differentiable function $V : \mathcal{D} \times \mathbb{R}_{\geq 0} \rightarrow \mathbb{R}_{\geq 0}$ satisfying

$$U_1(\zeta) \leq V(\zeta, t) \leq U_2(\zeta), \quad \dot{V}(\zeta, t) \leq -U(\zeta), \quad (28)$$

for all $t \geq 0$ and $\zeta \in \mathcal{D}$, where U_1 and U_2 are continuous positive-definite functions, and U is a uniformly continuous positive-semidefinite function. Then, for every initial condition $\zeta(0)$ belonging to

$$\mathcal{S} := \{\zeta \in \mathcal{D} : U_2(\zeta) \leq \delta\}, \quad \text{where } \delta < \min_{\|\zeta\|=\varsigma} U_1(\zeta), \quad (29)$$

it follows that

$$U(\zeta(t)) \rightarrow 0 \quad \text{as} \quad t \rightarrow \infty. \quad (30)$$

Now we are prepared to introduce the subsequent result concerning the neuroadaptive control strategy for the robotic manipulators.

Theorem 1. Consider the system (1) with error dynamics described by (2)–(4). Suppose that Assumptions 1–3 hold. If the control law is designed as (15) together with the ESN weight update law (16), then all system signals are bounded, and the tracking error converges to zero asymptotically, $\|e_1(t)\| \rightarrow 0$ as $t \rightarrow \infty$.

Proof. Consider the following Lyapunov function candidate:

$$V(y, t) = \frac{1}{2} r^T M r + Q + \frac{1}{2} e_1^T e_1 + \frac{1}{2} e_2^T e_2 + P(t) \quad (31)$$

where $Q = \frac{1}{2k_n} \text{tr}(\tilde{W}^T \tilde{W})$. The auxiliary non-negative function $P(t)$ is defined as follows:

$$P(t) = \xi - \int_0^t L(\tau) d\tau \quad (32)$$

where ξ and L are defined in (18)–(20). It can be verified that $P(t) \geq 0$. And $y(t)$ is defined as

$$y(t) := \begin{bmatrix} r^T(t) & e_1^T(t) & e_2^T(t) & \sqrt{Q} & \sqrt{P} \end{bmatrix}^T, \quad (33)$$

According to the positive definiteness of the inertia matrix $M(q)$, there exist positive constants m_1 and m_2 such that

$$m_1 \|r\|^2 \leq r^T M(q) r \leq m_2 \|r\|^2. \quad (34)$$

Therefore, the Lyapunov function can be bounded as

$$\lambda_1 \|y\|^2 \leq V(y, t) \leq \lambda_2 \|y\|^2, \quad (35)$$

where

$$\lambda_1 := \frac{1}{2} \min\{m_1, 1\}, \quad \lambda_2 := \max\left\{\frac{1}{2} m_2, 1\right\}. \quad (36)$$

Differentiating $V(t)$ with respect to time yields

$$\dot{V}(t) = r^T \dot{M} r + \frac{1}{2} r^T \dot{M} r + \frac{1}{k_n} \text{tr}(\tilde{W}^T \dot{\tilde{W}}) + e_1^T \dot{e}_1 + e_2^T \dot{e}_2 + \dot{P} \quad (37)$$

Substituting (16)–(18) into $\dot{V}(t)$ results in

$$\dot{V}(t) = -(k_s + 1) r^T r - k_w \|e_2\| \text{tr}(\tilde{W}^T \dot{\tilde{W}}) + e_1^T e_2 - \alpha_1 e_1^T e_1 - \alpha_2 e_2^T e_2 - k_w \frac{W_n^2}{4} e_2^T \text{sgn}(e_2) \quad (38)$$

Furthermore, noting that $\hat{W} = \tilde{W} + W$, yielding

$$\begin{aligned} -\text{tr}(\tilde{W}^T \hat{W}) &= -\text{tr}(\tilde{W}^T (\tilde{W} + W)) \\ &= -\|\tilde{W}\|^2 - \text{tr}(\tilde{W}^T W) \\ &\leq -\|\tilde{W}\|^2 + \|\tilde{W}\| \|W\| \\ &= -\left(\|\tilde{W}\| - \frac{\|W\|}{2}\right)^2 + \frac{1}{4}\|W\|^2 \end{aligned} \quad (39)$$

Next, by invoking Young's inequality, it follows that

$$e_1^T e_2 \leq \frac{1}{2} e_1^T e_1 + \frac{1}{2} e_2^T e_2 \quad (40)$$

By combining like terms with $-\alpha_1 e_1^T e_1 - \alpha_2 e_2^T e_2$, the grouped terms $-\left(\alpha_1 - \frac{1}{2}\right) e_1^T e_1$ and $-\left(\alpha_2 - \frac{1}{2}\right) e_2^T e_2$ are obtained. Furthermore, from $\|W\| \leq W_n$, it follows that

$$\begin{aligned} \dot{V}(t) &\leq -(k_s + 1)r^T r - \left(\alpha_1 - \frac{1}{2}\right) e_1^T e_1 - \left(\alpha_2 - \frac{1}{2}\right) e_2^T e_2 - k_w \|e_2\| \left(\|\tilde{W}\| - \frac{\|W\|}{2}\right)^2 \\ &\quad + \frac{1}{4} k_w \|e_2\| \|W\|^2 - \frac{1}{4} k_w \|e_2\|_1 W_n^2 \\ &\leq -(k_s + 1)r^T r - \left(\alpha_1 - \frac{1}{2}\right) e_1^T e_1 - \left(\alpha_2 - \frac{1}{2}\right) e_2^T e_2 \end{aligned} \quad (41)$$

Consequently, $\dot{V}(t)$ can be upper-bounded as

$$\dot{V}(t) \leq -U(y) \quad (42)$$

where

$$U(y) = (k_s + 1)r^T r + \left(\alpha_1 - \frac{1}{2}\right) e_1^T e_1 + \left(\alpha_2 - \frac{1}{2}\right) e_2^T e_2.$$

Since $k_s > 0$, $\alpha_1 > 1/2$, and $\alpha_2 > 1/2$, as specified in (4) and (15), $U(y)$ is a positive semidefinite function. From Equations (31) and (42), it is straightforward to see that $V(t) \in \mathcal{L}_\infty$. From the definition of $V(t)$, we can conclude that the signals r , e_1 , e_2 , $\tilde{W}$ (and thus $\hat{W}$), and P are all bounded. According to Equations (3) and (4) and (15)–(17), this further guarantees the boundedness of $\dot{e}_1$, $\dot{e}_2$, $\dot{\tilde{W}}$ ($\dot{\hat{W}}$), $\dot{r}$, and the control input u . Therefore, all closed-loop signals remain bounded.

Furthermore, define

$$D := \{y \in \mathbb{R}^{3n+2} : \|y\| < \varsigma\},$$

where $\varsigma > 0$ is sufficiently large such that the $y(t)$ remains inside D for all $t \geq 0$, which follows from the boundedness of all closed-loop signals established above. Correspondingly, let

$$S := \{y \in D : U_2(y) \leq \delta\},$$

where

$$\delta < \min_{\|y\|=\varsigma} U_1(y).$$

Then, from (35), the Lyapunov function satisfies

$$U_1(y) \leq V(y, t) \leq U_2(y),$$

where $U_1(y) = \lambda_1 \|y\|^2$ and $U_2(y) = \lambda_2 \|y\|^2$ are continuous positive-definite functions. Moreover, (42) gives $\dot{V}(y, t) \leq -U(y)$, where $U(y)$ is a continuous positive semidefinite

function. Since all closed-loop signals have been shown to be bounded, the $y(t)$ evolves in the compact set S , where $U(y)$ is uniformly continuous. Therefore, according to Lemma 2, it follows that

$$\lim_{t \rightarrow \infty} U(y(t)) = 0.$$

Since $k_s + 1 > 0$, $\alpha_1 > 1/2$, and $\alpha_2 > 1/2$, it follows that

$$\lim_{t \rightarrow \infty} \|r(t)\| = 0, \quad \lim_{t \rightarrow \infty} \|e_1(t)\| = 0, \quad \lim_{t \rightarrow \infty} \|e_2(t)\| = 0.$$

Hence, the closed-loop system is asymptotically stable. $\square$

Based on the rigorous stability analysis and the derived control laws, the detailed real-time implementation procedure of the proposed control scheme is systematically summarized in Table 1.

Table 1. Implementation steps of the proposed control scheme.

Step	Core Implementation Task
Step 1	Initialize robot initial states, controller gains and IP-cycle ESN hyperparameters.
Step 2	Measure joint position $q(t)$ and velocity $\dot{q}(t)$; construct tracking error e_1 , filtered error e_2 and auxiliary filtered error r .
Step 3	Build network input $z(t) = [q^T(t), \dot{q}^T(t), q_d^T(t), \dot{q}_d^T(t), \ddot{q}_d^T(t), e_1^T(t), e_2^T(t), \dot{e}_1^T(t)]^T$, update reservoir neuron gain $a_j(t)$ and bias $b_j(t)$ via intrinsic plasticity online learning rules.
Step 4	Use IP-driven cycle ESN to approximate unknown lumped nonlinearity $N(\cdot)$, and update ESN output weight $\hat{W}$ by the adaptive law.
Step 5	Compute the composite control input u , and apply the torque to the manipulator.
Step 6	Repeat Step 2 to Step 5 in real time until the tracking error asymptotically converges to zero.

4. Simulation

In this section, two simulation scenarios are provided to verify the effectiveness and superiority of the designed control scheme. The entire simulation framework is established and carried out utilizing MATLAB R2023b on a personal computer equipped with a 64-bit Windows 11 Professional operating system, an AMD Ryzen 7 7435H CPU@3.10 GHz, and 40 GB of system memory. The specific dynamic parameters, initial states, and reference trajectories of the manipulator system are systematically provided in each scenario.

• Simulation Scenario 1

In this scenario, the simulation is performed on the following two-joint robotic system [23].

$$\begin{bmatrix} M_{11} & M_{12} \\ M_{12} & M_{22} \end{bmatrix} \begin{bmatrix} \ddot{q}_1 \\ \ddot{q}_2 \end{bmatrix} + \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} \dot{q}_1 \\ \dot{q}_2 \end{bmatrix} + \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} + \begin{bmatrix} F_1 \\ F_2 \end{bmatrix} + \tau_d = u$$

where $M_{11} = (m_1 + m_2)r_1^2 + m_2r_2^2 + 2m_2r_1r_2 \cos(q_2) + J_1$, $M_{12} = m_2r_2^2 + m_2r_1r_2 \cos(q_2)$, $M_{22} = m_2r_2^2 + J_2$, $C_{11} = -m_2r_1r_2 \sin(q_2)\dot{q}_2$, $C_{12} = -2m_2r_1r_2 \sin(q_2)(\dot{q}_1 + \dot{q}_2)$, $C_{21} = 0$, $C_{22} = m_2r_1r_2 \sin(q_2)\dot{q}_1$, $G_1 = (m_1 + m_2)gr_1 \cos(q_1) + m_2gr_2(q_1 + q_2)$, $G_2 = m_2gr_2 \cos(q_1 + q_2)$, $F_1 = 0.5\dot{q}_1 + 0.3 \sin(\dot{q}_1)$, $F_2 = 0.3\dot{q}_2 - 0.4 \sin(\dot{q}_2)$. m_1 and m_2 are the masses of the two joints, r_1 and r_2 are the lengths of the joints and g represents the gravitational constant. In this work, the system parameters are chosen as $m_1 = 0.5$ kg, $m_2 = 1.5$ kg, $r_1 = 1.0$ m, $r_2 = 0.8$ m, $J_1 = 5$ kg·m², $J_2 = 5$ kg·m² and $g = 9.81$ m/s². The perturbation term is set

to $\tau_d = [-0.4 \sin(t), 0.3 \sin(t)]^T$. The parameters in the controller are $k_s = 250$, $\alpha_1 = 3.2$, and $\alpha_2 = 5.75$. The reservoir dimension is 100, with input and reservoir weight matrix parameters set to $\nu = 0.2$ and $\rho = 0.96$, respectively. The update law parameters for the output weights are $k_n = 0.001$ and $k_w = 1.5$. For intrinsic plasticity, the learning rate is $\eta = 0.0008$ and the expected distribution mean is $\mu = 0.4$. The initial states are defined as $q(0) = [0.5, 1.2]^T$ and $\dot{q}(0) = [1.5, 1.5]^T$, while the desired trajectories are $q_{d1} = \sin(t)$ and $q_{d2} = \cos(t)$.

The simulation results are presented in Figures 3–6 and Tables 2–4. Specifically, Figure 3 illustrates the trajectory tracking performance, showing that the system effectively tracks the desired trajectory after a short transient period. Subsequently, Figures 4 and 5 compare the control performance of the RBFNN-based PI control method [24] with that of the developed IP-driven cycle ESN under the same system model, identical reference trajectories, and identical initial conditions. Figure 4 illustrates the tracking errors of the two joints under different control methods. It can be seen that the proposed method exhibits a significantly faster convergence rate and achieves smaller transient tracking errors. The same result can also be observed from Tables 2 and 3, where the tracking accuracy and convergence rate are quantitatively evaluated. Specifically, the integral of absolute error $\int |e_i(t)| dt$ ($i = 1, 2$) is employed to measure the tracking precision, while the integrated squared error $\int (e_i(t) - \bar{e}_i)^2 dt$ serves as a metric for tracking smoothness, with $\bar{e}_i$ denoting the mean error. The quantitative results verify that the proposed method significantly enhances the steady-state accuracy while achieving a faster convergence rate across all joints. To visually illustrate the control behaviour, the trajectories of the control inputs are plotted in Figure 5. It can be seen that the proposed method generates significantly smoother control inputs than the RBFNN-based PI control method, particularly during the initial transient stage, where it effectively suppresses abrupt control variations and avoids large control peaks. After the transient response, both methods produce similar control signals to accomplish trajectory tracking. The quantitative comparisons presented in Table 4 further confirm these observations, demonstrating that the proposed method achieves smoother control inputs while maintaining comparable control effort. Figure 6 shows the evolution of the output weight norms of the proposed method. The norms of the output weights for both joints show initial transient adjustments and then converge to bounded, stable values, verifying that the adaptive learning rule ensures the stability of the network parameters during the control process.

Table 2. Quantitative comparison of tracking performance indicators for Scenario 1.

	$\int e_1(t) dt$	$\int e_2(t) dt$	$\int (e_1(t) - \bar{e}_1)^2 dt$	$\int (e_2(t) - \bar{e}_2)^2 dt$
RBFNN-based PI control method [24]	0.4546	0.1822	0.1032	0.01652
Proposed method	0.2366	0.0947	0.0712	0.0117

Table 3. Comparison of settling times under different methods for Scenario 1.

	Settling Time of Joint 1	Settling Time of Joint 2
RBFNN-based PI control method [24]	5.6739	4.8318
Proposed method	2.1445	1.835

Table 4. Performance metrics for control signals under different methods for Scenario 1.

	$\int u_1(t) dt$	$\int u_2(t) dt$	$\int (u_1(t) - \bar{u}_1)^2 dt$	$\int (u_2(t) - \bar{u}_2)^2 dt$
RBFNN-based PI control method [24]	428.02	150.35	9717.4	2105.2
Proposed method	427.96	145.67	1913.9	799.37

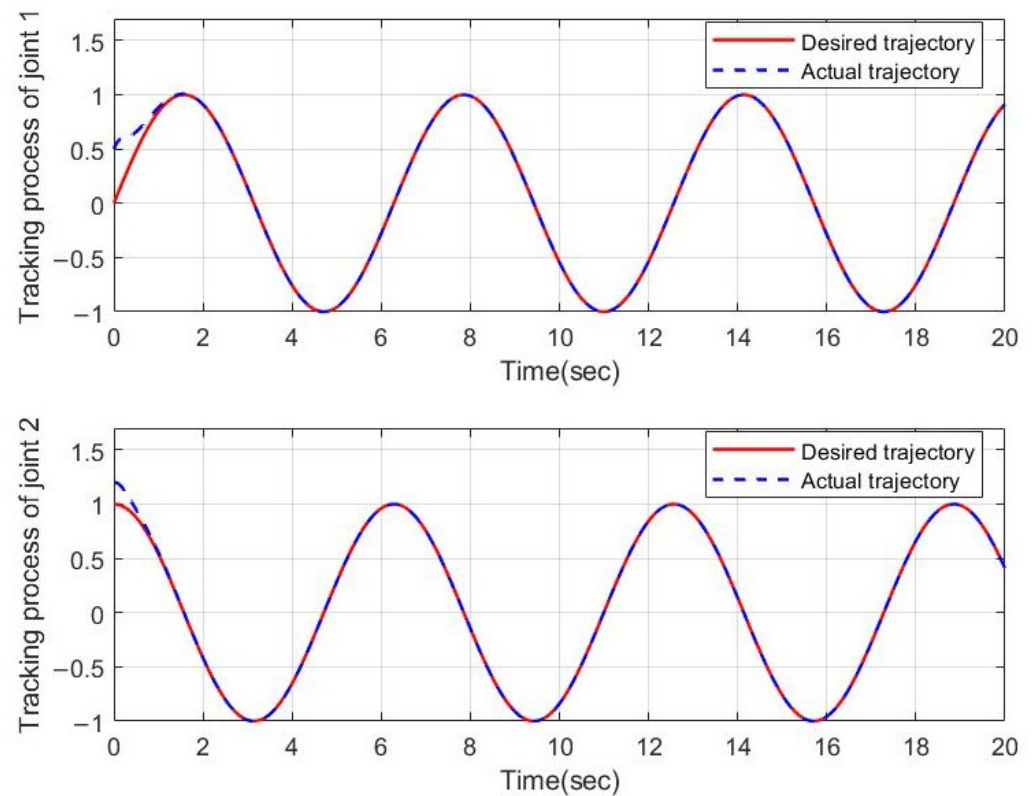


Figure 3. Tracking process under the proposed control scheme for Scenario 1.

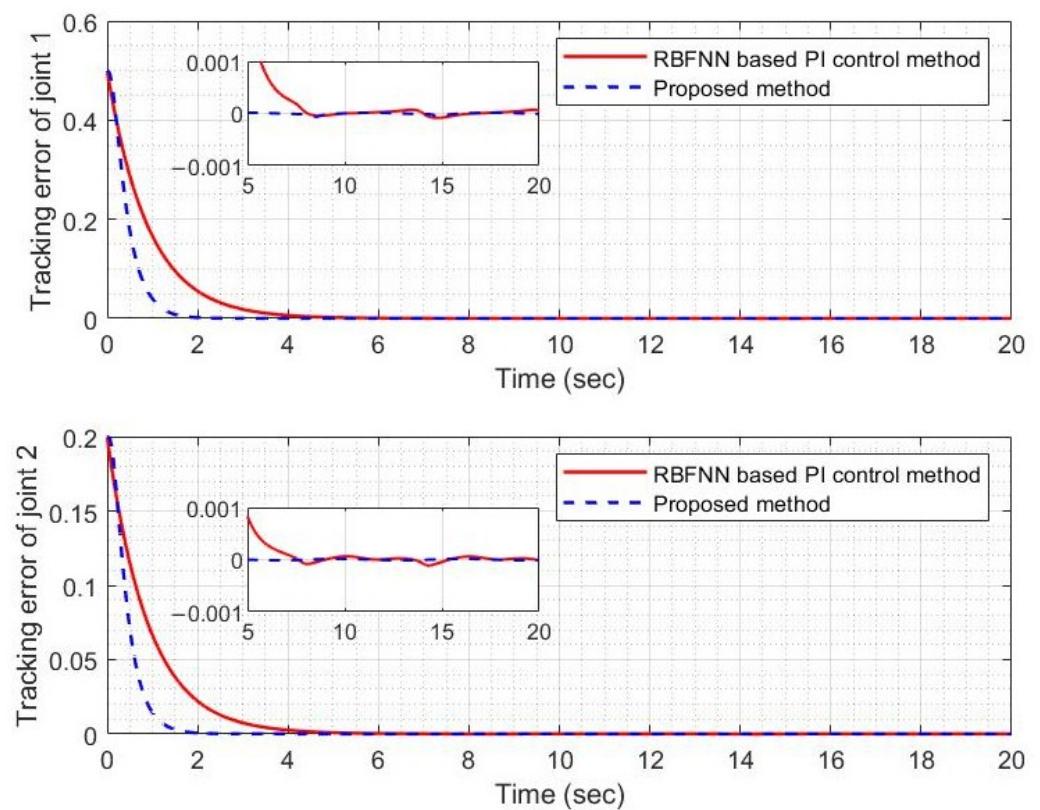


Figure 4. Tracking errors comparison between the RBFNN-based PI control method [24] and proposed method for Scenario 1.

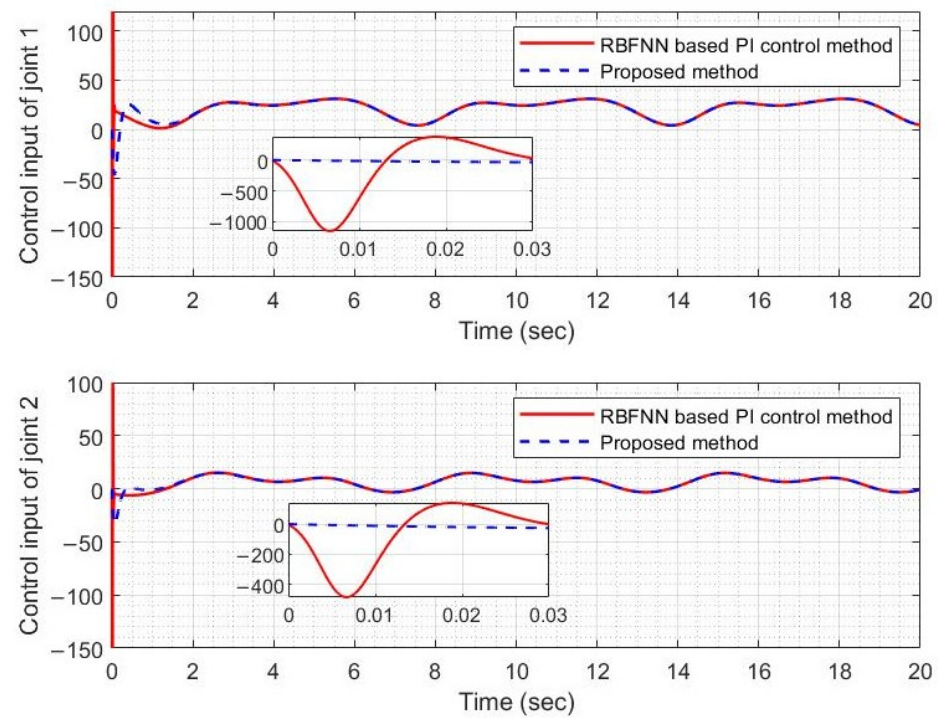


Figure 5. Control input comparison between the RBFNN-based PI control [24] method and the proposed method for Scenario 1.

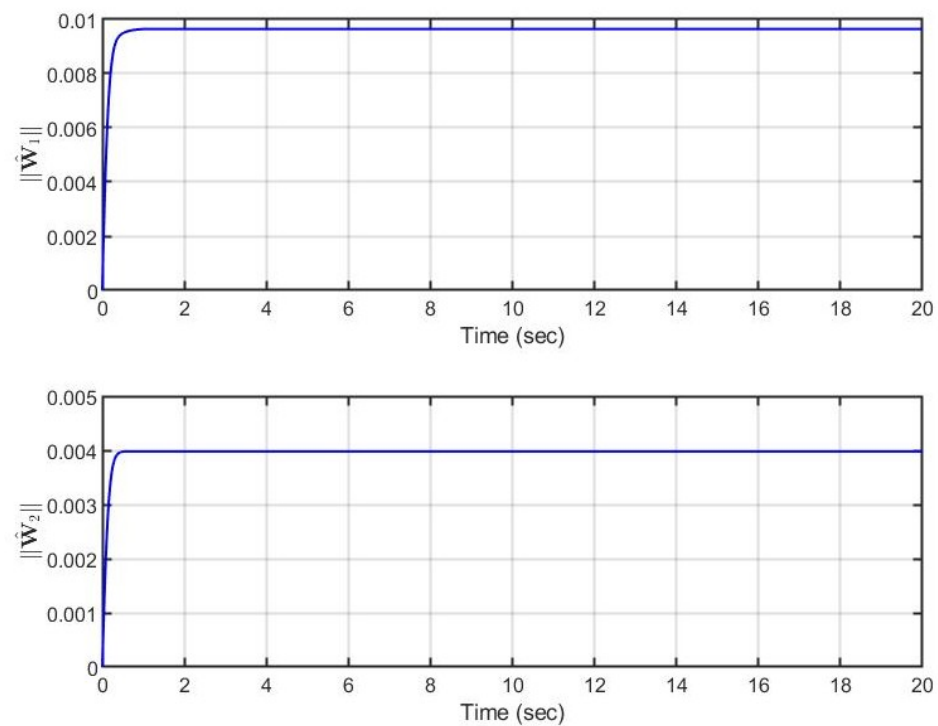


Figure 6. Norms of IP-driven cycle ESN output weight for Scenario 1.

- Simulation Scenario 2

In this scenario, the expressions for $M(q)$, $C(q, \dot{q})$, $G(q)$, $F(\dot{q})$ and τ_d are as follows [25]:

$$M(q) = \begin{bmatrix} p_1 + p_2 + 2p_3 \cos(q_2) & p_2 + p_3 \cos(q_2) \\ p_2 + p_3 \cos(q_2) & p_2 \end{bmatrix}$$

$$C(q, \dot{q}) = \begin{bmatrix} -p_3 \dot{q}_2 \sin(q_2) & -p_3 (\dot{q}_1 + \dot{q}_2) \sin(q_2) \\ p_3 \dot{q}_1 \sin(q_2) & 0 \end{bmatrix}$$

$$G(q) = \begin{bmatrix} p_4 \cos(q_1) + p_5 \cos(q_1 + q_2) \\ p_5 \cos(q_1 + q_2) \end{bmatrix}$$

$$F(\dot{q}) = 0.2 \sin(\dot{q})$$

$$\tau_d = [0.2 \sin(t); 0.2 \sin(t)]$$

$$p = [p_1, p_2, p_3, p_4, p_5] = [3.5, 0.76, 0.87, 3.04, 0.87].$$

The controller parameters are chosen as $k_s = 50$, $\alpha_1 = 2.4$, and $\alpha_2 = 3.5$. The neural network parameters are set as follows: the dimension of the reservoir is 150, the positive constants in the input layer and reservoir weight matrices are $\nu = 0.2$ and $\rho = 0.96$, the parameters of the neural network output weight update law are $k_n = 0.1$ and $k_w = 0.1$, the intrinsic plasticity learning rate is $\eta = 0.0008$, and the expected distribution mean is $\mu = 0.4$. The initial states q and $\dot{q}$ of the system are $q(0) = [0.5, 1.2]^T$ and $\dot{q}(0) = [1, 0]^T$, while the desired trajectories are set as $q_{d1} = \sin(t)$ and $q_{d2} = \cos(t)$.

The simulation results are illustrated in Figures 7–10 and Tables 5–7. As depicted in Figure 7, the system tracks the desired trajectory effectively. Similarly, Figure 8 illustrates the tracking error responses of both joints. Compared with the RBFNN-based PI control method, the proposed method provides faster transient convergence and smaller steady-state error bounds. The enlarged views further confirm its tighter convergence toward zero. This enhancement in tracking precision is mathematically verified by the tracking error performance metrics in Table 5, which demonstrate lower integrated absolute errors and smaller variances for the proposed approach. Furthermore, Table 6 evaluates the system's transient responsiveness. It shows that the proposed method achieves shorter settling times for both joints. Finally, to illustrate the corresponding control behavior, the trajectories of the control inputs are plotted in Figure 9. Although both methods achieve smooth steady-state control, the proposed method effectively reduces the initial control peaks, leading to a smoother transient response. This noticeable improvement in control smoothness is fully supported by the performance metrics presented in Table 7, where the reduced variance values confirm that the proposed scheme effectively suppresses undesired oscillations while maintaining a more economically efficient control energy expenditure. Figure 10 shows the evolution of the output weight norms of the IP-driven cycle ESN. The weight norms rapidly converge to constant values and remain bounded throughout the control process, demonstrating the stability of the proposed adaptive learning scheme.

Table 5. Quantitative comparison of tracking performance indicators for Scenario 2.

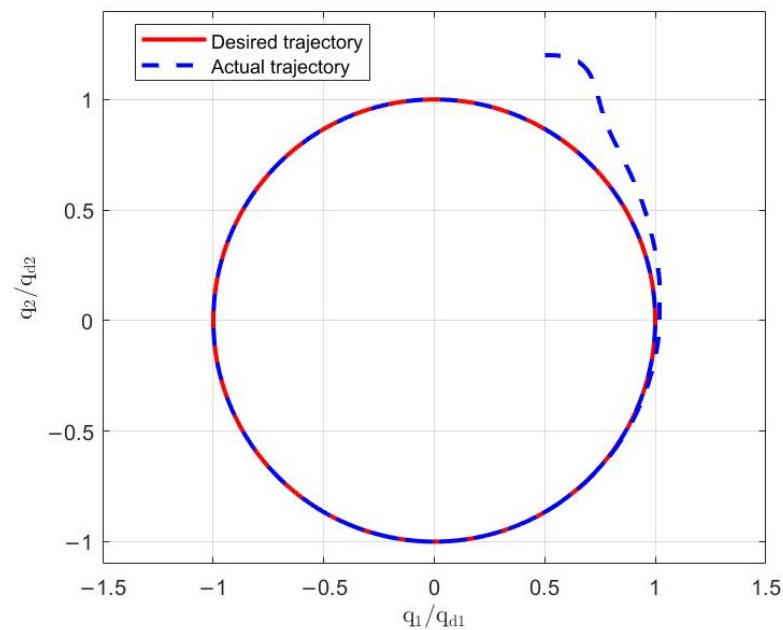
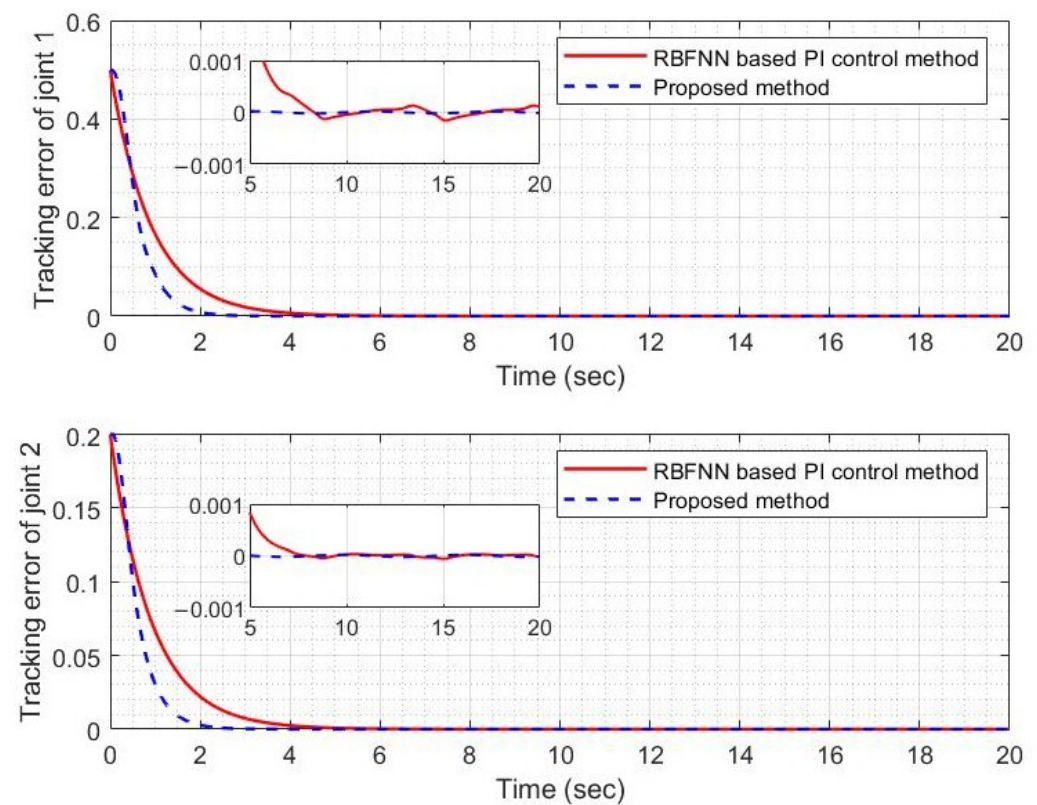
	$\int e_1(t) dt$	$\int e_2(t) dt$	$\int (e_1(t) - \bar{e}_1)^2 dt$	$\int (e_2(t) - \bar{e}_2)^2 dt$
RBFNN-based PI control method [24]	0.4552	0.1821	0.1033	0.0165
Proposed method	0.3236	0.1244	0.1004	0.0153

Table 6. Comparison of settling times under different methods for Scenario 2.

	Settling Time of Joint 1	Settling Time of Joint 2
RBFNN-based PI control method [24]	5.6974	4.8364
Proposed method	2.8468	2.4276

Table 7. Performance metrics for control signals under different methods for Scenario 2.

	$\int u_1(t) dt$	$\int u_2(t) dt$	$\int (u_1(t) - \bar{u}_1)^2 dt$	$\int (u_2(t) - \bar{u}_2)^2 dt$
RBFINN-based PI control method [24]	87.353	29.357	2947.2	210.78
Proposed method	84.162	28.495	345.13	48.375

**Figure 7.** Tracking process under the proposed control scheme for Scenario 2.**Figure 8.** Tracking errors comparison between the RBFINN-based PI control method [24] and proposed method for Scenario 2.

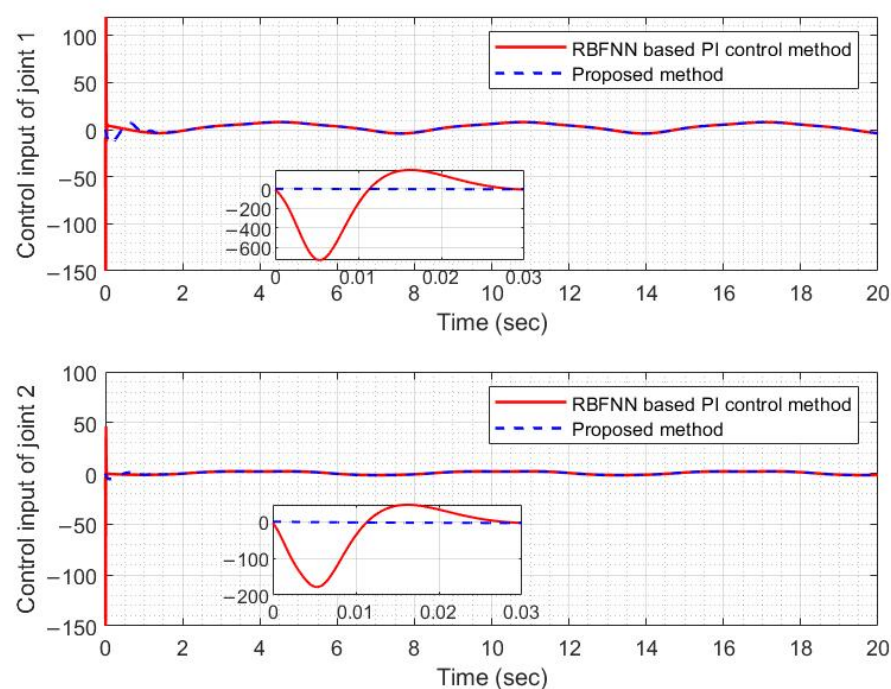


Figure 9. Comparison of the control inputs generated by the proposed method and RBFNN-based PI control method [24] for Scenario 2.

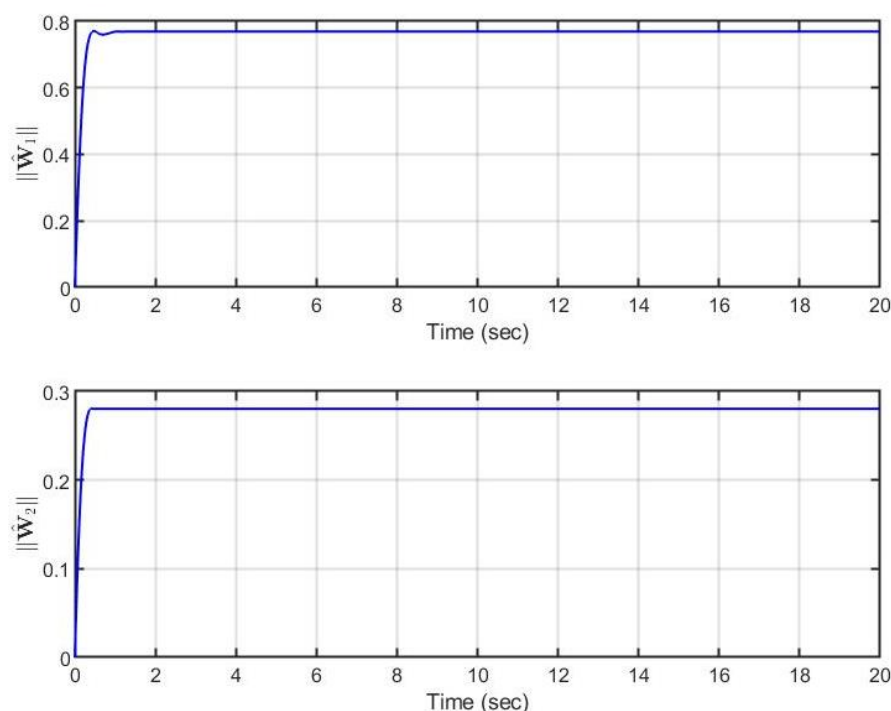


Figure 10. Norms of IP-driven cycle ESN output weight for Scenario 2.

5. Discussion

The simulation results presented in Section 4 demonstrate the superior performance of the proposed method compared with the RBFNN-based PI control approach in terms of tracking accuracy, convergence speed, and control input smoothness. As shown in Figures 3, 4, 7 and 8 and quantified in Tables 2 and 5, the proposed scheme achieves higher tracking precision. Moreover, the shorter settling times reported in Tables 3 and 6 indicate

a faster convergence rate. In addition, the smoothness of the generated control inputs is clearly demonstrated by Figures 5 and 9 and quantitatively validated by Tables 4 and 7.

Despite these advantages, several limitations of this study should be recognized. First, the theoretical validation and performance evaluations were established under ideal numerical simulation environments, which omit critical non-ideal practical constraints such as high-frequency sensor noise, actuator dead zones, and discrete sampling delays. Second, the reference trajectories tested in this work are entirely smooth continuous functions. In realistic industrial applications, controllers must frequently handle sudden trajectory modifications or non-smooth command abruptions. Addressing these practical gaps by enhancing the controller's resilience to discontinuous trajectories and conducting physical validation on an experimental hardware platform will be the primary focus of our subsequent research.

6. Conclusions

This paper develops an IP-driven cycle ESN integrated with RISE robust control to realize asymptotic trajectory tracking of uncertain robotic manipulators, where the cycle reservoir cuts network computation burden and intrinsic plasticity improves the network's approximation ability for time-varying nonlinearities, the adaptive law prevents neural weight drift and the RISE term compensates residual approximation errors and external disturbances, and Lyapunov analysis rigorously proves closed-loop signal boundedness and asymptotic convergence of tracking errors. Quantitative results from two simulation scenarios demonstrate that the proposed method outperforms the RBFNN-based PI control approach in terms of tracking accuracy, convergence speed, and control input smoothness. However, this study possesses several inherent limitations. First, all validation results were obtained under ideal numerical simulations without considering non-ideal factors such as practical sensor noise, actuator dead zones, and sampling delays. Second, the reference signals considered in this work are entirely smooth functions, without considering trajectories with some abrupt changes. To address these drawbacks, future work will be dedicated to improving our controller to adapt to abrupt trajectory changes and conducting physical hardware experiments.

Author Contributions: Q.C.: Methodology, Writing—Review and Editing. X.T.: Writing—Original Draft. S.H.: Writing—Review and Editing. M.Q.: Writing—Review and Editing. Z.L.: Methodology. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported in part by the National Natural Science Foundation of China under Grant 62203078, in part by the Science and Technology Research Program of Chongqing Municipal Education Commission under Grant KJQN202201542, in part by the Natural Science Foundation of Chongqing under Grant CSTB2023NSCQ-MSX0760, and in part by the Science and Technology Research Program of Chongqing Municipal Education Commission under Grant KJZD-M202301502 and KJZD-M202501504.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The simulation data generated and analyzed during the current study are available from the corresponding author upon reasonable request.

Conflicts of Interest: The authors declare no conflicts of interest.

References

1. Xiao, W.; Chen, K.; Fan, J.; Hou, Y.; Kong, W.; Dan, G. AI-driven rehabilitation and assistive robotic system with intelligent PID controller based on RBF neural networks. *Neural Comput. Appl.* **2023**, *35*, 16021–16035.

2. Jokar, H.; Naghipour, A.; Jeloudari, I. A new adaptive robust control scheme for trajectory tracking of robot manipulators with uncertain dynamics model. *IET Control Theory Appl.* **2025**, *19*, e70039.
3. Anjum, Z.; Sun, Z.; Chen, B. Disturbance-observer-based fault-tolerant control of robotic manipulator: A fixed-time adaptive approach. *IET Control Theory Appl.* **2024**, *18*, 1398–1413.
4. Hu, J.; Zhang, X.; Zhang, D.; Chen, Y.; Ni, H.; Liang, H. Finite-time adaptive super-twisting sliding mode control for autonomous robotic manipulators with actuator faults. *ISA Trans.* **2024**, *144*, 342–351.
5. Li, B.; Tian, W.; Zhang, C.; Hua, F.; Cui, G.; Li, Y. Positioning error compensation of an industrial robot using neural networks and experimental study. *Chin. J. Aeronaut.* **2022**, *35*, 346–360.
6. Liu, X.; Miao, Y.; Sun, C.; Ren, Z. Neural network sliding mode control of robotic manipulator based on exponential-trigonometric optimization algorithm. In *Proceedings of the 44th Chinese Control Conference (CCC)*; IEEE: New York, NY, USA, 2025; pp. 3447–3452.
7. Gao, X. Formation control, manipulator control and soft robot based on neural network control system. In *Proceedings of the 3rd International Symposium on Robotics & Intelligent Manufacturing Technology (ISRIMT)*; IEEE: New York, NY, USA, 2021; pp. 37–41.
8. Xu, K.; Wang, Z. The design of a neural network-based adaptive control method for robotic arm trajectory tracking. *Neural Comput. Appl.* **2023**, *35*, 8785–8795.
9. Zhou, Y.; Sun, Z. Control of stability in semi-passive robot based on RBF neural network. *Chaos Solitons Fractals* **2023**, *173*, 113623.
10. Liu, A.; Zhao, H.; Song, T.; Liu, Z.; Wang, H.; Sun, D. Adaptive control of manipulator based on neural network. *Neural Comput. Appl.* **2021**, *33*, 4077–4085.
11. Hao, J.; Lou, J. RBF neural network adaptive compensation control for robotic arms. In *Proceedings of the 11th International Conference on Control, Automation and Robotics (ICCAR)*; IEEE: New York, NY, USA, 2025; pp. 20–26.
12. Wang, Z.; Yao, X.; Li, T.; Zhang, H. Design of PID controller based on echo state network with time-varying reservoir parameter. *IEEE Trans. Cybern.* **2021**, *52*, 6615–6626.
13. Sun, W.; Wu, Y.; Lv, X. Adaptive neural network control for full-state constrained robotic manipulator with actuator saturation and time-varying delays. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *33*, 3331–3342.
14. Chen, Q.; Li, X.; Zhang, A.; Song, Y. Neuroadaptive tracking control of affine nonlinear systems using echo state networks embedded with multiclustered structure and intrinsic plasticity. *IEEE Trans. Cybern.* **2022**, *54*, 1133–1142.
15. Chen, Q.; Shi, H.; Sun, M. Echo state network-based backstepping adaptive iterative learning control for strict-feedback systems: An error-tracking approach. *IEEE Trans. Cybern.* **2019**, *50*, 3009–3022.
16. Wu, R.; Li, Z.; Pan, Y. Adaptive echo state network robot control with guaranteed parameter convergence. In *Proceedings of the International Conference on Intelligent Robotics and Applications*; Springer: Cham, Switzerland, 2021; pp. 587–595.
17. Atsuta, H.; Kawai, Y.; Asada, M. Enhancement of the robustness of redundant robot arms against perturbations by inferring dynamical systems using echo state networks. In *Proceedings of the International Joint Conference on Neural Networks (IJCNN)*; IEEE: New York, NY, USA, 2024; pp. 1–6.
18. Li, Y.; Li, C. Synergies between intrinsic and synaptic plasticity based on information theoretic learning. *PLoS ONE* **2013**, *8*, e62894.
19. Triesch, J. A gradient rule for the plasticity of a neuron's intrinsic excitability. In *Proceedings of the International Conference on Artificial Neural Networks*; Springer: Berlin, Germany, 2005; pp. 65–70.
20. Feng, G. A compensating scheme for robot tracking based on neural networks. *Rob. Auton. Syst.* **1995**, *15*, 199–206.
21. Yang, Q.; Jagannathan, S.; Sun, Y. Robust integral of neural network and error sign control of MIMO nonlinear systems. *IEEE Trans. Neural Netw. Learn. Syst.* **2015**, *26*, 3278–3286.
22. Khalil, H.K.; Grizzle, J.W. *Nonlinear Systems*; Prentice Hall: Upper Saddle River, NJ, USA, 2002.
23. Gao, M.; Ding, L.; Jin, X. ELM-based adaptive faster fixed-time control of robotic manipulator systems. *IEEE Trans. Neural Netw. Learn. Syst.* **2021**, *34*, 4646–4658.
24. Song, Y.; Guo, J.; Huang, X. Smooth neuroadaptive PI tracking control of nonlinear systems with unknown and nonsmooth actuation characteristics. *IEEE Trans. Neural Netw. Learn. Syst.* **2017**, *28*, 2183–2195.
25. Fierro, R.; Lewis, F.L. Control of a nonholonomic mobile robot using neural networks. *IEEE Trans. Neural Netw.* **1998**, *9*, 589–600.

Disclaimer/Publisher's Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.