

# Article Fixed-Time Incremental Neural Control for Manipulator Based on Composite Learning with Input Saturation

Yanli Fan, Haiqi Huang and Chenguang Yang \*

Key Laboratory of Autonomous Systems and Networked Control, College of Automation Science and Engineering, South China University of Technology, Guangzhou 510640, China \* Correspondence: cyang@ieee.org

\* Correspondence: cyang@ieee.org

Abstract: In this paper, an adaptive incremental neural network (INN) fixed-time tracking control scheme based on composite learning is investigated for robot systems under input saturation. Firstly, by integrating the composite learning method into the INN to cope with the inevitable dynamic uncertainty, a novel adaptive updating law of NN weights is designed, which does not need to satisfy the stringent persistent excitation (PE) conditions. In addition, for the saturated input, differing from adding the auxiliary system, this paper introduces a hyperbolic tangent function to deal with the saturation nonlinearity by converting the asymmetric input constraints into the symmetric ones. Moreover, the fixed-time control approach and Lyapunov theory are combined to ensure that all the signals of the robot closed-loop control systems converge to a small neighborhood of the origin in a fixed time. Finally, numerical simulation results verify the effectiveness of the fixed-time control and composite learning algorithm.

Keywords: fixed-time control; INN control; composite learning; input saturation



Citation: Fan, Y.; Huang, H.; Yang, C. Fixed-Time Incremental Neural Control for Manipulator Based on Composite Learning with Input Saturation. *Actuators* **2022**, *11*, 373. https://doi.org/10.3390/ act11120373

Academic Editor: Ioan Ursu

Received: 30 October 2022 Accepted: 7 December 2022 Published: 10 December 2022

**Publisher's Note:** MDPI stays neutral with regard to jurisdictional claims in published maps and institutional affiliations.



**Copyright:** © 2022 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) license (https:// creativecommons.org/licenses/by/ 4.0/).

## 1. Introduction

With the development of automation technology towards intelligent manufacturing, robots have been widely used in military rescue, intelligent assembly, human robot cooperation tasks, and other industrial manufacturing fields. On many occasions, robots and humans share the same workspace. If robot tasks are not performed accurately, unexpected accidents will occur. Therefore, how to accurately control the robots to perform the required tasks has always been an exceedingly important research direction [1-3]. Many scholars have proposed the model-based control schemes in which the inherent dynamics of the robot are fully known. However, in practice, due to the uncertainty of the environment and the coupling effect, it is difficult to determine the manipulator model and obtain the required control accuracy correctly and quickly. Thus, in recent years, a great deal of control schemes based on universal approximation such as neural networks (NNs) [4–7] and fuzzy logic systems (FLSs) [8–10] have emerged to deal with the unknown nonlinearity terms in the manipulator systems. If the network nodes were selected unreasonably, this would lead to insufficient learning of the nonlinear terms and the degradation of approximation performance, which was the common weakness in the above-mentioned NN approximation. An effective way to settle this issue is to dynamically increase the number of nodes, which is called the INN [11]. No matter whether the NN or INN, it should be pointed out that the network weight converges to the real value when the sufficient condition for accurate approximation is established, and the PE condition is an important condition to ensure the convergence of estimation [12]. The PE condition is difficult to achieve, so it is often not feasible in practical systems. It is worth emphasizing that the partial PE condition proposed in [13] relaxed the traditional PE, and it was pointed out that any recursive NN input trajectory of the radial basis function neural network (RBFNN) defined in a local regular lattice can be partially activated. In the works [14-17], this idea was utilized in the

adaptive control design of nonlinear systems such as a manipulator, to ensure the stability and realize the convergence of the NN weights. Based on the above discussion, how to solve the problem of insufficient learning caused by improper node selection on the basis of relaxing the strict PE condition still needs to be further discussed.

On the other hand, during the task of a manipulator, the input control torque cannot exceed a certain upper limit value due to the inevitable constraints of electromechanical failure or mechanical structure change. If the torque required by the controller exceeds the torque provided by the actuator, unpredictable mechanical faults may occur, thereby damaging the system performance. Hence, it is a meaningful work to investigate the input saturation [18–24] for practical nonlinear systems. For uncertain nonlinear systems, considering the effect of input saturations, the authors in [18,19], by constructing an auxiliary system, developed two effective control schemes to restrain them. Subsequently, the works in [22–24] extended the above methods to robotic manipulators [22], ships [23], and suspension systems [24] and proposed several control schemes that can suppress the adverse effects of saturation on the actual systems. Obviously, these schemes compensated the saturations' nonlinearity by introducing an auxiliary system, yet the auxiliary system intensified the complexity of the system structure to a certain extent, expanded the amount of calculation, and may also lead to chattering. In order to amend these flaws, the works in [20,21] designed a hyperbolic tangent function to approximate the saturation nonlinearity for symmetric input constraints, which alleviated the vibration and large amount of calculation caused by the auxiliary system. Nevertheless, it must be emphasized that the above work did not consider the asymmetric saturation of the actuators or require the convergence time of the controlled system. To a certain extent, rapid convergence will lead to the increase of the vibration amplitude. Therefore, under the constraint of input nonlinearity, it is hard to obtain stability and good error convergence, which inspired our research.

In order to achieve high accuracy and fast convergence, the control approach of finitetime convergence is a preferred solution. The pioneering work [25] introduced the concept of finite-time stability and proved that it has high-level robustness and faster transient response. Subsequently, many scholars have followed this work, and many constructive works have emerged [26–29]. The work [26] presented a smooth finite-time control scheme of strict feedback nonlinear systems, and the work [27] investigated the issue of finite-time for nonlinear systems with unmeasurable states through a quantized feedback control strategy. The research [25–29] limited the convergence time to a range by finite-time control, but all relied on the initial state of the system. For the sake of breaking these restrictions, the authors in [30] proposed a fixed-time control scheme, which is independent of the initial conditions and can be adjusted by the controller parameters. To this end, a large number of adaptive intelligent control algorithms have been developed [31–35]. By applying universal barrier functions, a fixed-time control with symmetric constraints for MIMO uncertain systems was studied in [31]. In [32], the observer-based fixed-time tracking control problem of SISO nonlinear systems was considered. In [33,34], the fixed-time method was extended to multiagent systems and a flexible robotic manipulator. In [35], the authors developed fixed-time control with distrubance for manipulator system, and the work of this paper extends [35] and introduces composite learning approach on the fram work of the NN fixed-time theory.

Inspired by the above discussions, this paper proposes an adaptive INN fixed-time tracking control scheme based on composite learning for the manipulator system under saturated input. The main contributions are as follows:

1. A new INN adaptive algorithm based on the composite learning technique is designed. By dynamically generating network nodes and introducing some persistence conditions to adapt to the NN weights, the estimation error information can be properly integrated into the adaptive law, and the estimation performance is improved on the basis of relaxing the traditional PE conditions. Even though the works [14–17] studied composite learning control, they did not consider dynamically activating network nodes to adjust the NN input, let alone ensuring that the estimation error converged in a fixed time.

2. In the framework of the backstepping composite learning approach, the challenge of devising a fixed-time controller with asymmetric actuator saturation of the manipulator system is effectively tackled. Although the authors in [18,19,22–24] considered the problem of actuator input saturation, they all solved symmetric saturation by introducing auxiliary systems, which is not tenable for the asymmetric scenario. Instead, this paper not only proposes a feasible asymmetric saturation control scheme, but also ensures fixed-time convergence under the composite learning framework.

The rest of this paper is structured as follows. In Section 2, the system description and some useful lemmas are introduced. Section 3 shows the radial basis function NN and the INN. Section 4 introduces how to design the INN controller based on composite learning in the presence of input saturation and presents the stability analysis. The proposed control scheme in this paper is validated by the simulation in Section 5. Finally, Section 6 gives the conclusion of this paper.

## 2. System Description

The dynamic model of a manipulator with *n* degrees of freedom can be described as:

$$H(\theta)\ddot{\theta} + C(\dot{\theta},\theta)\dot{\theta} + G(\theta) = \tau \tag{1}$$

where  $\theta \in \mathbb{R}^n$  represents the joint position angle vector,  $H(\theta) \in \mathbb{R}^{n \times n}$  represents the inertia matrix,  $C(\dot{\theta}, \theta) \in \mathbb{R}^{n \times n}$  represents the Coriolis matrix,  $G(\theta) \in \mathbb{R}^n$  represents the gravity, and  $\tau \in \mathbb{R}^n$  represents the control torque and satisfies

$$\tau_{i} = \begin{cases} \tau_{imax}, & u_{i} > u_{imax} \\ f_{ri}(u_{i}), & 0 \le u_{i} \le u_{imax} \\ f_{li}(u_{i}), & -u_{imin} \le u_{i} \le 0 \\ -\tau_{imin}, & u_{i} < -u_{imin} \end{cases}$$
(2)

where  $u_i(i = 1, ..., n)$  is the *i*-th element of the control law  $u = [u_1, u_2, ..., u_n]^T$ .  $\tau_{max} = [\tau_{1max}, ..., \tau_{nmax}]^T$ ,  $\tau_{min} = [\tau_{1min}, ..., \tau_{nmin}]^T$ ,  $u_{max} = [u_{1max}, ..., u_{nmax}]^T$ , and  $u_{min} = [u_{1min}, ..., u_{nmin}]^T$  are the known saturation parameters; nonlinear functions  $f_{ri}(u_i)$ and  $f_{li}(u_i)$  are unknown, but smoothly continuous;  $f_{ri}(0) = f_{li}(0) = 0$  and the assumption that  $v_{ri0} \leq \dot{f}_{ri}(u_i) \leq v_{ri1}$ ,  $v_{li0} \leq \dot{f}_{li}(u_i) \leq v_{li1}$  are satisfied in  $u_i \in [0, u_{imax}]$  and  $u_i \in [-u_{imin}, 0]$ , respectively, where  $v_{ri0}, v_{ri1}, v_{li0}$ , and  $v_{li1}$  are positive constants.

The manipulator has the following two commonly used properties.

**Property 1** ([10]). For any  $\theta$ ,  $\dot{\theta} \in \mathbb{R}^n$ , H, C, and G are bounded and their first derivatives are available.

**Property 2** ([10]).  $\dot{H} - 2C$  is an antisymmetric matrix, which means, for any  $z \in \mathbb{R}^n$ ,  $z^T$   $(\dot{H} - 2C)z = 0$  are established.

**Lemma 1** ([31]). If a positive Lyapunov function V(x) satisfies:  $\dot{V} < -pV^{\alpha} - qV^{\beta} + h$ , where  $p > 0, q > 0, h > 0, 0 < \alpha < 1$ , and  $\beta > 1$ , then the system  $\dot{x} = f(x,t)$  is considered to be stable in fixed-time, whose solution will converge to a compact set  $\Omega = \{x | V \le \min([h/(p(1-\gamma))]^{1/\alpha}, [h/(q(1-\gamma))]^{1/\beta})\}$ , where  $0 < \gamma < 1$ , and the convergence time of the system satisfies:  $T \le T_{max} = \frac{1}{p\gamma(1-\alpha)} + \frac{1}{q\gamma(\beta-1)}$ .

**Lemma 2** ([31,33]). *For any*  $x_i \in \mathbb{R}$ *, we have* 

$$\sum_{i=1}^{n} |x_i|^r \ge \left(\sum_{i=1}^{n} |x_i|\right)^r, \ if \quad 0 < r \le 1$$
(3)

$$\sum_{i=1}^{n} |x_i|^r \ge n^{1-r} \left( \sum_{x=1}^{n} |x_i| \right)^r, \ if \quad r > 1$$
(4)

**Lemma 3** ([16]). The vector  $\xi_2$  is not used, then its estimate value can be obtained according to the following second-order filter:

$$\begin{cases} \hat{\xi}_1 = \hat{\xi}_2 \\ \hat{\xi}_2 = w^2(\xi_1 - \hat{\xi}_1) - \zeta w \hat{\xi}_2 \end{cases}$$
(5)

where  $\hat{\xi}_1$  and  $\hat{\xi}_2$  are estimates of  $\xi_1$  and  $\xi_2$ , respectively. Note that with the frequency w fit low enough, the estimation error of  $\xi_2$  is small enough to make  $\hat{\xi}_2 \approx \xi_2$ .

**Definition 1** ([13]). For an n-dimensional bounded column vector  $\Phi(t)$ , if there are positive constants  $a_1$ ,  $a_2$ , and  $t_0$  such that  $a_1I \leq \int_{t_0}^{t_0+\Delta t} \Phi(u)\Phi^T(u)du \leq a_2I$  is valid for arbitrary time interval  $\Delta t$ , then the bounded signal  $\Phi(t)$  is called PE.

## 3. Radial Basis Function NN Approximator

The RBFNN has a strong nonlinear approximation ability. As long as there are enough nodes, any nonlinear term can be approximated with any accuracy. However, due to the limited number of nodes in practical applications, there will be an approximation error. The expression of the RBFNN approximating the nonlinear term is

$$Y(x) = W^T \Phi(x) + \epsilon(x)$$
(6)

where  $W \in \mathbb{R}^{n \times k}$  denotes the network weight,  $\Phi(x) \in \mathbb{R}^k$  denotes the regression vector,  $\epsilon(x) \in \mathbb{R}^n$  denotes the approximation error,  $x \in \mathbb{R}^m$  denotes the input vector of the network, n denotes the dimension of the nonlinear term to be estimated, k denotes the number of nodes in the neural network, and m denotes the dimension of the network input vector.

The Gaussian function is the most commonly used basis function in the RBFNN because of its simple form and being differentiable at any order, and its expression is as follows:  $(x - u_i)^T (x - u_i)$ 

$$\Phi_i(x) = exp[\frac{-(x-\mu_i)^T(x-\mu_i)}{\sigma_i^2}], \ i = 1, \dots, k$$
(7)

where  $\mu_i \in \mathbb{R}^m$  denotes the *i*-th node and  $\sigma_i \in \mathbb{R}^+$  denotes the standard deviation.

μ(

In order to overcome the shortcomings of the inaccurate nodes manually set in advance, the method of incremental learning [11] to generate nodes is introduced, and the node addition process is shown in Figure 1. Firstly, the initial state of the system is set to the first node of the network, i.e.,

$$t_0) = x \tag{8}$$

In the following process, the network will decide whether to generate nodes according to the distance between the current input and the node set. After each sampling period T, the node set of the network is

$$\mu(t+T) = \begin{cases} [\mu(t), \mu_{new}], & dis > \varepsilon \\ \mu(t), & dis \le \varepsilon \end{cases}$$
(9)

where  $\mu(t) \in \mathbb{R}^{m \times k}$  denotes the node set of the network at time t,  $\mu_{new} \in \mathbb{R}^{m \times 1}$  indicates the newly generated node, while  $\mu(t + T)$  denotes the node set at time t + T. *dis* denotes the Euclidean distance between the network input and node set, whose calculation formula is shown as (10), and  $\varepsilon$  is the distance threshold parameter to be designed.

$$dis = \|x - \bar{\mu}_{min}\| \tag{10}$$

where  $\bar{\mu}_{min}$  represents the weighted average node of the neural network, which is obtained according to the weighted average of *b* nodes closest to the current input vector, i.e.,

$$\bar{\mu}_{min} = \frac{\sum_{i=1}^{b} \arg\min_{\mu}\{||x - \mu_i||\}}{b}$$
(11)

Finally, the newly generated nodes of the network are  $\mu_{new} = \bar{\mu}_{min} + \delta(x - \bar{\mu}_{min})$ 

$$\iota_{new} = \bar{\mu}_{min} + \delta(x - \bar{\mu}_{min}) \tag{12}$$

where  $\delta > 0$  is the parameter to be designed.



Figure 1. The schematic diagram of the incremental neural network.

From the above process, it is not difficult to find that, compared with the traditional neural network, the neural network using incremental learning has a significant advantage in that it can ensure that the input is always in the compact domain of the neural network, enabling the maximum activation of the nodes. Since the degree of node activation depends on the Euclidean distance between the current input and the nodes and the nodes of the incremental neural network are generated according to the real-time input, when the current input is not in the network compact domain, the nodes will be automatically added to expand the compact domain, so the effectiveness of the incremental neural network can be better guaranteed. In contrast, the nodes in the traditional NN need to be set up artificially; when the prior knowledge of the complex system is not accurate, the designed node set is far away from the input of the system and cannot be activated effectively, which will lead to a significant reduction in the approximation performance of the network. We call the RBFNN with incremental learning the IRBFNN.

## 4. Controller Design and Stability Analysis

## 4.1. Controller Design

The proposed control scheme is shown in Figure 2. Let  $z_1 = \theta$  and  $z_2 = \dot{\theta}$ ; the dynamic equation of the system (1) can be rewritten as

$$H\dot{z}_2 + Cz_2 + G = \tau \tag{13}$$

The state space equation of the manipulator can be sorted out as

$$\dot{z}_1 = z_2, \quad \dot{z}_2 = H^{-1}(\tau - Cz_2 - G)$$
(14)

Thus, define the error signal of the system as

$$\tilde{z}_1 = z_1 - \theta_d, \quad \tilde{z}_2 = z_2 - z_{2d}$$
 (15)

where  $\theta_d \in \mathbb{R}^n$  denotes the desired joint position and  $z_{2d} \in \mathbb{R}^n$  denotes the virtual control law to be designed later.



Figure 2. The overview of the proposed control scheme.

The dynamic error equation of the system can be obtained as

$$H\dot{z}_{2} + C\tilde{z}_{2} = \tau - H\dot{z}_{2d} - Cz_{2d} - G \tag{16}$$

According to the backstepping control method and the theory of fixed-time stability, design the virtual control law  $z_{2d}$  as

$$z_{2d} = -K_{12}\tilde{z}_1 - K_{12}\sum_{i=1}^n w_i(\tilde{z}_{1,i}) - K_{13}Sig^\beta(\tilde{z}_1) + \dot{\theta}_d$$
(17)

In order to ensure that  $z_{2d}$  and its derivate are both continuous, the nonlinear function  $w_i(\tilde{z}_{1,i})$  is designed as

$$w_{i}(\tilde{z}_{1,i}) = \begin{cases} Sig^{\alpha}(\tilde{z}_{1,i}), & |\tilde{z}_{1,i}| \ge \vartheta \\ a_{1}\tilde{z}_{1,i} + a_{2}\tilde{z}_{1,i}^{3}, & |\tilde{z}_{1,i}| < \vartheta \end{cases}$$
(18)

where  $\vartheta$  is a small constant,  $Sig^{c}(.) = sign(.)|.|^{c}$ .

Using the IRBFNN to estimate the unknown model of the system, we obtain

$$Y(x) = W^T \Phi(x) + \epsilon(x) = -H\dot{z}_{2d} - Cz_{2d} - G$$
(19)

where  $x = [z_1, z_2, z_{2d}] \in \mathbb{R}^{3n \times 1}$  is the input of the IRBFNN and  $Y(x) \in \mathbb{R}^{n \times 1}$ ,  $W \in \mathbb{R}^{k \times n}$ ,  $\Phi(x) \in \mathbb{R}^{k \times 1}$ , and  $\epsilon(x) \in \mathbb{R}^{n \times 1}$  represent the item to be estimated, the ideal weight, the regression vector, and the approximation error, respectively. *k* is the number of nodes of the IRBFNN, and  $||\epsilon(x)|| \leq \bar{\epsilon}$ . Thus, the dynamic error equation can be rewritten as

$$H\dot{\tilde{z}}_2 + C\tilde{z}_2 = \tau + W^T \Phi(x) + \epsilon(x)$$
<sup>(20)</sup>

Define the estimated error  $\tilde{W}$  of the network weight as

$$\tilde{W} = \hat{W} - W \tag{21}$$

where  $\hat{W}$  is an estimate of the ideal network weight *W*.

In order to make the network weights converge in a fixed time without the strict PE condition, we designed the weight update law with composite learning technology [16]. Firstly, the error of the IRBFNN is defined as:

$$E_i(t) = B(t)\hat{W}_i - D_i(t) , \ i = 1, 2, \dots, n$$
(22)

where  $\hat{W}_i(t) \in \mathbb{R}^{k \times 1}$  represents the estimated network weight of the *i*-th joint and  $E_i(t) \in \mathbb{R}^{k \times 1}$  is the corresponding network estimation error, and the auxiliary terms  $B(t) \in \mathbb{R}^{k \times k}$  and  $D_i(t) \in \mathbb{R}^{k \times 1}$  are defined as follows:

$$\begin{cases} B(t) = \int_0^t e^{-r(t-u)} \Phi(x) \Phi^T(x) du\\ D_i(t) = \int_0^t e^{-r(t-u)} \Phi(x) (W_i^T \Phi(x) + \epsilon_i(x)) du \end{cases}$$
(23)

where  $B(t) \in \mathbb{R}^{k \times k}$  is a symmetric matrix, and  $B(t) = B^T(t)$ .

Combining (23) and (21), the estimation error of the network (22) can be further arranged as

$$E_{i} = B\hat{W}_{i} - D_{i}$$

$$= \int_{0}^{t} e^{-r(t-u)} \Phi(x) (\Phi^{T}(x)(\hat{W}_{i} - W_{i}) - \epsilon_{i}(x)) du \qquad (24)$$

$$= B\tilde{W}_{i} + \Psi_{i}$$

where  $\Psi_i = -\int_0^t e^{-r(t-u)} \Phi(x) \epsilon_i(x) du$  and  $\Psi_i$  is bounded. In (24),  $B\tilde{W}_i$  corresponds to the estimation error of the network weight and  $\Psi_i$  corresponds to the approximation error, so  $E_i$  corresponds to the total error of the IRBFNN.

To calculate  $W_i^T \Phi(x) + \epsilon_i(x)$  in Equation (23), from Equations (13) and (19), we can obtain

$$W_i^T \Phi(x) + \epsilon_i(x) = \tau_i \varphi_i \tag{25}$$

$$\varphi_{i} = \begin{bmatrix} \dot{\tilde{z}}_{2,i} + \dot{z}_{2d,i} \\ \tilde{z}_{2,i} + z_{2d,i} \\ 1 \end{bmatrix}^{\dagger} \begin{bmatrix} -\dot{\tilde{z}}_{2d,i} \\ -z_{2d,i} \\ -1 \end{bmatrix}$$
(26)

where  $[\cdot]^{\dagger}$  denotes the pseudo inverse of a vector. Since  $z_{2,i} = \ddot{\theta}_i$  is the angular acceleration of the manipulator and cannot be obtained directly, its estimated value is obtained through the second-order filter shown in Lemma 3, that is

$$\hat{\varphi}_{i} = \begin{bmatrix} \hat{z}_{2,i} + \dot{z}_{2d,i} \\ z_{2,i} + z_{2d,i} \\ 1 \end{bmatrix}^{\mathsf{T}} \begin{bmatrix} -\dot{z}_{2d,i} \\ -z_{2d,i} \\ -1 \end{bmatrix}$$
(27)

Then, the expression of  $D_i(t)$  in Equation (23) can be further arranged as

$$D_i(t) = \int_0^t e^{-r(t-u)} \Phi(x) \tau_i \hat{\varphi}_i du$$
(28)

Obviously, the expression (23) is a solution for the system of differential Equation (29), so the update laws of the auxiliary terms B, D are set as:

$$\begin{cases} \dot{B} = -rB + \Phi(x)\Phi^{T}(x) \\ \dot{D}_{i} = -rD_{i} + \Phi(x)\tau_{i}\hat{\varphi}_{i} \end{cases}$$
(29)

Based on the composite learning approach, design  $\dot{W}_i$  by the IRBFNN, as shown in (30):

$$\hat{W}_i = \Gamma_i^{-1}(\Phi(x)\tilde{z}_{2,i} - K_{32,i}Sig^{\alpha}(E_i) - K_{33,i}Sig^{\beta}(E_i))$$
(30)

where  $\Gamma_i \in \mathbb{R}^{k \times k}$ ,  $K_{32,i} \in \mathbb{R}^{k \times k}$ , and  $K_{33,i} \in \mathbb{R}^{k \times k}$  are all full-rank diagonal constant matrices and their elements are all positive.

**Remark 1.** It is not difficult to find that the weight composite learning law in the Formula (30) combines the position tracking error signal of the current system with the total error signal of the IRBFNN. The idea is to ensure the convergence of the weight by using the feedback effect of the historical error generated by the system and the updating of the weight. It should be noted that, compared with the existing results, the composite learning law designed by us can guarantee not only the weight convergence, but also convergences at a fixed time.

In order to satisfy the input saturation effect of the actuator, introduce the variable  $\Xi$  to assist the torque design; one has

$$\Xi = diag[\bar{\tau}_i/\bar{u}_i], i = 1, 2, \dots, n$$
(31)

where  $diag[\cdot]$  denotes a diagonal matrix,  $\bar{\tau}_i$  represents the upper limit of the torque that the actuator can accept, while  $\bar{u}_i$  represents the parameter related to the expected maximum control law, and

$$\begin{cases} \bar{\tau}_i = \tau_{imax}, \bar{u}_i = u_{imax}, \quad \tau_{imax}/u_{imax} \ge \tau_{imin}/u_{imin} \\ \bar{\tau}_i = \tau_{imin}, \bar{u}_i = u_{imin}, \quad \tau_{imax}/u_{imax} < \tau_{imin}/u_{imin} \end{cases}$$
(32)

According to (20), we can obtain

$$\dot{\tilde{z}}_2 = H^{-1}(\Xi \Xi^{-1}\tau + W^T \Phi(x) + \epsilon) - H^{-1}C\tilde{z}_2$$
(33)

In the proposed controller, the relation between the actual control torque  $\tau$  and the designed control law *u* is as follows:

$$\tau = \chi(u) + \Delta \tau \tag{34}$$

where  $\Delta \tau$  represents the difference between the actual input saturation effect curve and the ideal value and the assumption that  $|\Delta \tau| \leq \Delta \overline{\tau}$  and  $\chi(u)$  represents the conversion value of the control law *u* after the saturation effect. Use the hyperbolic tangent function to solve the input saturation effect, that is

$$\chi(u_i) = \frac{\bar{\tau}_i}{\eta_i} \arctan(\frac{\eta_i u_i}{\bar{u}_i}), i = 1, 2, \dots, n$$
(35)

where  $\eta_i = \pi/2$ . According to mean-value theorem, the following equation holds:

$$\Xi^{-1}\chi(u) = u - \Upsilon(u)u \tag{36}$$

$$Y(u) = diag[\frac{(v_i u_i \pi / 2\bar{u}_i)^2}{1 + (\pi v_i u_i / 2\bar{u}_i)^2}], i = 1, 2, \dots, n$$
(37)

where  $0 < v_i < 1$ .

According to (20) and (36), this can be arranged as

$$\dot{\tilde{z}}_2 = H^{-1}(\Xi u - \Xi Y(u)u + \Delta \tau + W^T \Phi(x) + \epsilon) - H^{-1}C\tilde{z}_2$$
(38)

According to the fixed-time control theory, the control law of *u* is constructed as

$$u = \Xi^{-1}u_o + u_a, \quad u_a = -\frac{\|u_o\|\tilde{z}_2}{\|\tilde{z}_2\|(1-l)}, \|\tilde{z}_2\| \neq 0$$
(39)

$$u_{o} = -\tilde{z}_{1} - \tilde{z}_{2} - K_{22}Sig^{\alpha}(\tilde{z}_{2}) - K_{23}Sig^{\beta}(\tilde{z}_{2}) - \hat{W}^{T}\Phi(x)$$

where  $\lambda_{max}(Y(u)) < l < 1$ , and if  $||\tilde{z}_2|| = 0$ ,  $u_a = 0$ .

**Remark 2.** It should be mentioned that the control scheme in this paper can solve the trajectory tracking control task of single-arm manipulator systems with input saturation well. However, if the proposed control scheme is extended to dual-arm manipulator systems, it is necessary to analyze the internal force relationship between the two arms, which makes the controller design process more complex.

#### 4.2. Stability Analysis

The property of the proposed IRBFNN composite learning fixed-time control scheme can be summarized as the following theorem.

**Theorem 1.** Consider the manipulator system (1) with input saturation, and design the virtual controllers (17), the actual fixed-time controller (39), and the composite learning law (30), then it can be concluded that:

- (1) All the error signals are guaranteed to converge in a fixed time;
- (2) The position signal  $\theta$  converges to a small neighborhood of the desired position  $\theta_d$  in a fixed time;
- (3) The joint torque is guaranteed not to transgress the constraints sets.

Proof. According to the error signal in the controller, choosing the following three candidate Lyapunov functions, we obtain

$$V_{1} = \frac{1}{2} \tilde{z}_{1}^{T} \tilde{z}_{1}, \quad V_{2} = \frac{1}{2} \tilde{z}_{2}^{T} H \tilde{z}_{2}$$

$$V_{3} = \frac{1}{2} \sum_{i=1}^{n} ((B^{-1} E_{i})^{T} \Gamma_{i} B^{-1} E_{i})$$
(40)

Combine (15) and (17); we can obtain  $\dot{V}_1$  as

$$\dot{V}_{1} = \tilde{z}_{1}^{T} \dot{\tilde{z}}_{1} = \tilde{z}_{1}^{T} (\tilde{z}_{2} - K_{12} \tilde{z}_{1} - K_{12} \sum_{i=1}^{n} w_{i}(\tilde{z}_{1,i}) - K_{13} Sig^{\beta}(\tilde{z}_{1}))$$

$$= \tilde{z}_{1}^{T} \tilde{z}_{2} - \sum_{i=1}^{n} (k_{12,i} \tilde{z}_{1,i}^{2}) - \sum_{i=1}^{n} (k_{12,i} \tilde{z}_{1,i} w_{i}(\tilde{z}_{1,i})) - \sum_{i=1}^{n} (k_{13,i} |\tilde{z}_{1,i}|^{\beta+1})$$
(41)

In accordance with the definition of  $w_i(\tilde{z}_{1,i})$ , the following two situations about the term  $\Gamma = -\sum_{i=1}^{n} k_{12,i} \tilde{z}_{1,i} w_i(\tilde{z}_{1,i})$  should be considered: Case 1: If  $|\tilde{z}_{1,i}| \ge \vartheta$ , it follows that

$$\Gamma = -\sum_{i=1}^{n} (k_{12,i} |\tilde{z}_{1,i}|^{\alpha+1})$$
(42)

Case 2: If  $|\tilde{z}_{1,i}| < \vartheta$  and according to the inequality  $|c|^{\mu}|d|^{\rho} \leq \mu/(\mu+\rho)\bar{\gamma}|c|^{\mu+\rho}$  $+\rho/(\mu+\rho)\bar{\gamma}^{-\mu/\rho}|d|^{\mu+\rho}$  ( $\mu,\rho$  and  $\bar{\gamma}$  are positive), we have

$$\Gamma = -\sum_{i=1}^{n} k_{12,i} \tilde{z}_{1,i} (a_1 \tilde{z}_{1,i} + a_2 \tilde{z}_{1,i}^3) 
\leq -\sum_{i=1}^{n} (k_{12,i} |\tilde{z}_{1,i}|^{\alpha+1}) + \sum_{i=1}^{n} (k_{12,i} \tilde{z}_{1,i}^2) 
+ \zeta (1-\alpha) \bar{\gamma} - \sum_{i=1}^{n} k_{12,i} (a_1 + a_2) \tilde{z}_{1,i}^2$$
(43)

where  $\zeta = min\{k_{12,1}, k_{12,2}, \dots, k_{12,i}\}$ . Considering that  $a_1 + a_2 > 0$  and following the above two cases of analysis yield

$$\Gamma \le -\sum_{i=1}^{n} (k_{12,i} |\tilde{z}_{1,i}|^{\alpha+1}) + \sum_{i=1}^{n} (k_{12,i} \tilde{z}_{1,i}^2) + \zeta(1-\alpha)\bar{\gamma}$$
(44)

Combining (41) with (44), we obtain

$$\dot{V}_{1} \leq \tilde{z}_{1}^{T} \tilde{z}_{2} - \sum_{i=1}^{n} (k_{12,i} | \tilde{z}_{1,i} |^{\alpha+1}) - \sum_{i=1}^{n} (k_{13,i} | \tilde{z}_{1,i} |^{\beta+1}) + \zeta (1-\alpha) \bar{\gamma}$$
(45)

From (20) and (39) and Property 2,  $\dot{V}_2$  can be expressed as

$$\begin{split} \dot{V}_{2} &= \tilde{z}_{2}^{T} H \dot{\bar{z}}_{2} + \frac{1}{2} \tilde{z}_{2}^{T} \dot{H} \tilde{z}_{2} \\ &= \tilde{z}_{2}^{T} (u_{o} + \Xi u_{a} - \Xi Y(u)u + \Delta \tau - H \dot{z}_{2d} - C z_{2d} - G) \\ &\leq \tilde{z}_{2}^{T} (-\tilde{z}_{1} - \tilde{z}_{2} - K_{22} Sig^{\alpha}(\tilde{z}_{2}) - K_{23} Sig^{\beta}(\tilde{z}_{2}) \\ &- \hat{W}^{T} \Phi(x) + W^{T} \Phi(x) + \epsilon + \Delta \tau + \Xi u_{a} - \Xi Y(u)u) \end{split}$$
(46)  
$$&\leq - \sum_{i=1}^{n} (k_{22,i} |\tilde{z}_{2,i}|^{\alpha+1}) - \tilde{z}_{2}^{T} \tilde{z}_{1} - \sum_{i=1}^{n} (k_{23,i} |\tilde{z}_{2,i}|^{\beta+1}) - \tilde{z}_{2}^{T} \tilde{z}_{2} \\ &- \tilde{z}_{2}^{T} \tilde{W}^{T} \Phi(x) + \tilde{z}_{2}^{T} (\epsilon + \Delta \tau) + \tilde{z}_{2}^{T} (\Xi u_{a} - \Xi Y(u)u) \end{split}$$

According to (24), we have

$$B^{-1}E_i = \tilde{W}_i + B^{-1}\Psi_i \tag{47}$$

From Equation (23) and the definition of matrix *B*, one has

$$(B^{-1}E_i)^T = \tilde{W}_i^T + \Psi_i^T B^{-1}$$
(48)

Taking the derivative of (47) with respect to time, we obtain

$$\frac{\partial (B^{-1}E_i)}{\partial t} = \frac{\partial (\tilde{W}_i + B^{-1}\Psi_i)}{\partial t}$$

$$= \dot{W}_i + B^{-1}\dot{B}B^{-1}\Psi_i + B^{-1}\dot{\Psi}_i$$

$$= \dot{\hat{W}}_i - \dot{W}_i + \Psi'_i$$

$$= \dot{\hat{W}}_i + \Psi'_i$$
(49)

From Equation (23) and combined with the definition of  $\Psi$ , it is known that  $\dot{\Psi}_i$  is bounded, so  $\Psi'_i$  is bounded.

According to (30), (40), (47), and (49), one can obtain

$$\begin{split} \dot{V}_{3} &= \sum_{i=1}^{n} (B^{-1}E_{i})^{T} \Gamma_{i} (\dot{\tilde{W}}_{i} + \Psi_{i}') \\ &= \sum_{i=1}^{n} (B^{-1}E_{i})^{T} \Gamma_{i} \dot{\tilde{W}}_{i} + \sum_{i=1}^{n} (B^{-1}E_{i})^{T} \Gamma_{i} \Psi_{i}' \\ &= \sum_{i=1}^{n} (B^{-1}E_{i})^{T} (\Phi(x)\tilde{z}_{2,i} - K_{32,i}Sig^{\alpha}(E_{i})) \\ &- K_{33,i}Sig^{\beta}(E_{i}))) + \sum_{i=1}^{n} (B^{-1}E_{i})^{T} \Gamma_{i} \Psi_{i}' \\ &= \sum_{i=1}^{n} E_{i}^{T} B^{-1} (-K_{32,i}Sig^{\alpha}(E_{i}) - K_{33,i}Sig^{\beta}(E_{i})) \\ &+ \sum_{i=1}^{n} (\tilde{W}_{i}^{T} + \Psi_{i}^{T}B^{-1}) \Phi(x)\tilde{z}_{2,i} + (B^{-1}E_{i})^{T} \Gamma_{i} \Psi_{i}') \\ &= \sum_{i=1}^{n} \sum_{j=1}^{k} (-k'_{32,i,j}|E_{i,j}|^{\alpha+1} - k'_{33,i,j}|E_{i,j}|^{\beta+1}) \\ &+ \sum_{i=1}^{n} (\tilde{W}_{i}^{T} \Phi(x)\tilde{z}_{2,i} + \Psi_{i}^{T}B^{-1} \Phi(x)\tilde{z}_{2,i} + E_{i}^{T}B^{-1} \Gamma_{i} \Psi_{i}') \end{split}$$

Choose a general candidate Lyapunov function V as

$$V = V_1 + V_2 + V_3 \tag{50}$$

For the convenience and conciseness of the proof process, let

$$A = -A1 - A2 - A3$$

$$A1 = \sum_{i=1}^{n} (k_{12,i} |\tilde{z}_{1,i}|^{\alpha+1} + k_{13,i} |\tilde{z}_{1,i}|^{\beta+1}) - \zeta(1-\alpha)\bar{\gamma}$$

$$A2 = \sum_{i=1}^{n} (\tilde{z}_{2,i}^{2} + k_{22,i} |\tilde{z}_{2,i}|^{\alpha+1} + k_{23,i} |\tilde{z}_{2,i}|^{\beta+1})$$

$$A3 = \sum_{i=1}^{n} \sum_{i=1}^{k} (k'_{32,i,j} |E_{i,j}|^{\alpha+1} + k'_{33,i,j} |E_{i,j}|^{\beta+1})$$
(51)

Based on (45), (46), (50), and (51) and Young's inequality, it can be obtained that

$$\begin{split} \dot{V} &= \dot{V}_{1} + \dot{V}_{2} + \dot{V}_{3} \\ &\leq A + \sum_{i=1}^{n} (\Psi_{i}^{T} B^{-1} \Phi(x) \tilde{z}_{2,i} + E_{i}^{T} B^{-1} \Gamma_{i} \Psi_{i}' + \tilde{z}_{2}^{T} (\epsilon_{i} + \Delta \tau)) \\ &\leq A + \sum_{i=1}^{n} (\frac{\|\Psi_{i}^{T} B^{-1} \Phi(x)\|^{2}}{2} + \frac{\tilde{\epsilon}_{i}^{2} + \Delta \bar{\tau}_{i}^{2}}{2} + \frac{\|E_{i}\|^{2} + \|B^{-1} \Gamma_{i} \Psi_{i}'\|^{2}}{2} + \tilde{z}_{2,i}^{2}) \\ &\leq -\sum_{i=1}^{n} (k_{12,i} |\tilde{z}_{1,i}|^{\alpha+1} + k_{13,i} |\tilde{z}_{1,i}|^{\beta+1}) + \zeta(1-\alpha)\bar{\gamma} \\ &- \sum_{i=1}^{n} (k_{22,i} |\tilde{z}_{2,i}|^{\alpha+1} + k_{23,i} |\tilde{z}_{2,i}|^{\beta+1}) \\ &- \sum_{i=1j=1}^{n} (k_{32,i,j} |E_{i,j}|^{\alpha+1} + k_{33,i,j}' |E_{i,j}|^{\beta+1} \\ &+ \sum_{i=1}^{n} (\frac{\|\Psi_{i}^{T} B^{-1} \Phi(x)\|^{2}}{2} + \frac{\|B^{-1} \Gamma_{i} \Psi_{i}'\|^{2}}{2} + \frac{\tilde{\epsilon}_{i}^{2} + \Delta \bar{\tau}_{i}^{2}}{2}) \\ &\leq -\kappa_{1} V^{\frac{\alpha+1}{2}} - \kappa_{2} V^{\frac{\beta+1}{2}} + v \end{split}$$

where  $\kappa_1$ ,  $\kappa_2$  satisfy  $\kappa_1 = min(2\lambda_{min}(K_{12}), \frac{2\lambda_{min}(K_{22})}{\lambda_{max}(H)}, \frac{2\lambda_{min}(K'_{32})}{\lambda_{max}(\Gamma)})$ ,  $\kappa_2 = min(2\lambda_{min}(K_{13})n^{\frac{1-\beta}{2}})$  $\frac{2\lambda_{\min}(K_{23})n^{\frac{1-\beta}{2}}}{\lambda_{\max}(H)}, \frac{2\lambda_{\min}(K'_{33})n^{\frac{1-\beta}{2}}}{\lambda_{\max}(\Gamma)}), K'_{32} = B^{-1}K_{32}, \text{ and } K'_{33} = B^{-1}K_{33}. \text{ According to the above analysis, it is known that } B^{-1}, \epsilon_i, \Phi(x), \text{ and } \Psi'_i \text{ are all bounded. That is, there exists a positive constant <math>\nu$  such that  $\nu = \sum_{i=1}^n \left(\frac{\|\Psi_i^T B^{-1}\Phi(x)\|^2}{2} + \frac{\|B^{-1}\Gamma_i\Psi_i'\|^2}{2} + \frac{\tilde{\epsilon}_i^2 + \Delta \tilde{\tau}_i^2}{2}\right) + \zeta(1-\alpha)\bar{\gamma}.$  In light of Lemma 1, the related signals  $\vartheta = [\tilde{z}_1, \tilde{z}_2, E]^T$  that make up the Lyapunov function V will converge to the compact set  $\Omega = \left\{\vartheta|V \leq \min\left(\frac{\nu}{\kappa_1(1-\gamma)}\frac{2}{\kappa+1}, \frac{\nu}{\kappa_2(1-\gamma)}\frac{2}{\beta+1}\right)\right\}$ 

in a fixed time, and the convergence time satisfies:  $T \leq T_{max} = \frac{2}{\kappa_1 \gamma (1-\alpha)} + \frac{2}{\kappa_2 \gamma (\beta-1)}$ , where  $0 < \gamma < 1$ .  $\Box$ 

## 5. Simulation Verification

#### 5.1. Simulation Settings

In order to verify the feasibility and effectiveness of the proposed control scheme, a simulation experiment was carried out on a two-DOF manipulator model [14], and the expression is shown in (53), while the model is shown in Figure 3. The meanings of the parameters in the expression are shown in Table 1. The desired trajectory is shown in (54). The mass and length parameters of the linkage manipulator were set as follows  $m_1 = 4$  kg,  $m_2 = 2 \text{ kg}, g = 9.81 \text{ N/kg}, l_1 = 0.5 \text{ m}, \text{ and } l_2 = 0.5 \text{ m}.$  The saturation torque of Joint 1 was selected as 100*N*, and the saturation torque of Joint 2 was select as 50*N*, i.e.,  $\bar{\tau} = [100, 50]^T$ . The saturation characteristic parameter  $\bar{u}$  is given as  $\bar{u} = [300, 300]^T$ .

$$\begin{bmatrix} H_{11} & H_{12} \\ H_{21} & H_{22} \end{bmatrix} \begin{bmatrix} \ddot{\theta}_1 \\ \ddot{\theta}_2 \end{bmatrix} + \begin{bmatrix} C_{11} & C_{12} \\ C_{21} & C_{22} \end{bmatrix} \begin{bmatrix} \dot{\theta}_1 \\ \dot{\theta}_2 \end{bmatrix} + \begin{bmatrix} G_1 \\ G_2 \end{bmatrix} = \begin{bmatrix} \tau_1 \\ \tau_2 \end{bmatrix}$$
(53)

$$\theta_d(t) = \begin{bmatrix} \theta_{d1} \\ \theta_{d2} \end{bmatrix} = \begin{bmatrix} -1 + 0.5 \times \sin(0.5\pi t) \\ 1 + 0.6 \times \cos(0.5\pi t) \end{bmatrix}$$
(54)

The parameters in the virtual controller and actual control law were chosen as  $K_{12} = diag[3,2], K_{13} = diag[4,2], K_{22} = diag[3,2], \text{ and } K_{23} = diag[4,2].$  The relevant parameters of the INN are given as follows:  $\sigma = 0.5$ ,  $\delta = 0.9$ . The maximum number of nodes in the network is 100,  $\varepsilon = 0.5$ . The parameters of the network weight update rate based on composite learning were chosen as  $K_{32} = diag[4, 4]$  and  $K_{33} = diag[2, 2]$ . The fixed-time-related parameters were set to  $\alpha = 0.6$  and  $\beta = 2$ .



Figure 3. The model of the 2-DOF manipulator.

In order to more intuitively compare the performance of the incremental neural network controller based on composite learning (CLIRBFNN), we set up a group of convincing comparative simulations. The comparative controllers adopted the fixed-time incremental neural network controller (IRBFNN), the general neural network controller (RBFNN), and the PD+ controller (PD+), and the weight update rate of the comparative controller adopted the traditional update rate, that is

$$\dot{\hat{W}} = \Gamma^{-1}(\Phi(x)\tilde{z}_2 - \lambda\hat{W}) \tag{55}$$

where  $\Gamma = diag[2, 2]$  and  $\lambda = diag[10, 10]$ . The control law of the compared PD+ controller is

$$u = -K_p \tilde{z}_1 - K_d \dot{\tilde{z}}_1 + G \tag{56}$$

$$\tau_i = \begin{cases} \bar{\tau}_i Sign(u_i), |u_i| > \bar{\tau}_i \\ u_i, \quad |u_i| \le \bar{\tau}_i \end{cases}$$
(57)

where the gravity term *G* is fully compensated. The values of  $K_p$  and  $K_d$  were selected as  $K_p = diag[100, 40]$  and  $k_d = diag[80, 40]$ .

Table 1. The description of the 2-joint manipulator.

$H_{11}$	$(m_1 + m_2)l_1^2 + m_2l_2^2 + 2m_2l_1l_2\cos\theta_2$
$H_{12}$	$m_2(l_2^2 + l_1 l_2 cos \theta_2)$
H <sub>21</sub>	$m_2(l_2^2 + l_1 l_2 cos \theta_2)$
H <sub>22</sub>	$m_2 l_2^2$
C <sub>11</sub>	$-m_2l_1l_2\dot{q}_2sin heta_2$
C <sub>12</sub>	$m_2 l_1 l_2 (\dot{ heta_2}+\dot{ heta_1}) sin  heta_2$
C <sub>21</sub>	$m_2 l_1 l_2 \dot{\theta_1} sin \theta_2$
C <sub>22</sub>	0
$G_1$	$m_2 l_2 gcos(\theta_1 + \theta_2) + (m_1 + m_2) l_1 gcos \theta_1$
G <sub>2</sub>	$m_2 l_2 g cos(\theta_1 + \theta_2)$

In order to specifically compare the controllers under different initial states, three groups of different initial states were set, i.e.,  $\theta(0) = [-1,1]$ ,  $\theta(0) = [-0.6, 0.6]$ , and  $\theta(0) = [-0.5, 0.5]$ , and the corresponding initial position errors were  $\tilde{z}_1(0) = [-2.6, 2.6]$ ,  $\tilde{z}_1(0) = [-1.5, 1.5]$ , and  $\tilde{z}_1(0) = [-0.4, 0.4]$ . The purpose of this setting was to fully investigate the performance of the controller when the initial error is more than 20-times the preset steady-state error. On the other hand, the performance of the two joints under positive and negative errors can be observed. Moreover, the difference between the performance of the controller under a small initial error and a large initial error can be seen.

#### 5.2. Result Analysis

The simulation results are presented in Figures 4–12. Figures 4 and 5 show the curves of the tracking error under different initial conditions. It can be seen that different initial

values have a certain impact on the transient response, but have little impact on the steadystate error. This means that the designed fixed-time controller has good performance under different initial conditions. Figures 6–9 present the tracking errors and control torques of Joint 1 and Joint 2 based on different controllers. It is obviously seen that the controller proposed in this paper can ensure that the error converges to a smaller neighborhood than the other compared controllers. Figure 10 shows the norm of the weight. We can see that the they are all bounded and can converge to constants. Figure 11 shows that each dimension weight value of the CLIRBFNN can eventually converge to the constant value, and adding new nodes to the network, the trained weights do not need to be retrained, which indicates that the CLIRBFNN adopted by the proposed controller has a high calculation efficiency. The weight update rate based on the IRBFNN and RBFNN was designed using Lyapunov stability, which was not applied to the error term of the network, so it can only be guaranteed to converge in a certain equilibrium interval. In terms of amplitude, the weight of the CLIRBFNN is at least three orders of magnitude higher than that of the IRBFNN and RBFNN, which expresses the effectiveness of composite learning in weight training. Figure 12 describes the network estimation error for Joint 1 and Joint 2 under the CLIRBFNN, IRBFNN, and RBFNN, respectively. For the CLIRBFNN, the network estimation error of each dimension will eventually converge to the smallest neighborhood near zero in a fixed time. For the IRBFNN and RBFNN, the network estimation error can only oscillate periodically in a large interval, and the oscillation period is positively related to the period of the desired trajectory.

In order to more quantitatively analyze the role of each control term in the control process, we analyzed the relative proportion of each control term. Blue refers to the first and second terms of  $u_0$  in (39); red is the third term of  $u_0$ ; green is the fourth term of  $u_0$ ; yellow is the NN's control term. Figure 12 depicts that, in the absence of composite learning control, i.e., the IRBFNN and RBFNN, the weights cannot be well trained, resulting in the proportion of network control terms being less than one-thousandth of the whole control process. Therefore, the approximation performance of the system is degraded, which is also the reason why the steady-state errors in Figures 6 and 7 are larger than that of the proposed controller in this paper. From Figure 13a,d, it can be seen that, in the transient regulation stage, the proportion terms (blue) and fixed-time terms (red and green) account for a large proportion, which indicates that the transient regulation mainly depends on the non-network terms. In the process of steady-state regulation, the proportion of the CLIRBFNN terms exceeds 80% of the total most of the time, and the proportion of the non-network control terms is very small, which shows that the CLIRBFNN compensates the uncertainty terms well and has excellent approximation performance.

Through the above analysis, the following conclusions can be drawn: the NN fixedtime controller based on composite learning can ensure that the network weights converge when the continuous excitation is not satisfied, and all error signals of the manipulator system can converge to a small neighborhood near zero in a fixed time with the input being saturated.



**Figure 4.** Error  $\tilde{z}_{1,1}$  with different initial values.



**Figure 5.** Error  $\tilde{z}_{1,2}$  with different initial values.



Figure 6. The tracking error of Joint 1.



**Figure 7.** The tracking error of Joint 2.



Figure 8. The torque of Joint 1.



Figure 9. The torque of Joint 2.



**Figure 10.** The norm of the weights for the two joints based on the CLIRBFNN, IRBFNN, and RBFNN. (**a**–**c**) show the norm of the weights for Joint 1. (**d**–**f**) show the norm of the weights for Joint 2.



**Figure 11.** The weight of two joints based on the CLIRBFNN, IRBFNN, and RBFNN. (**a**–**c**) show the weights of Joint 1. (**d**–**f**) show the weights of Joint 2.



**Figure 12.** The estimation error based on the CLIRBFNN, IRBFNN, and RBFNN. (**a**–**c**) show the estimation error of Joint 1. (**d**–**f**) show the estimation error of Joint 2.



**Figure 13.** The torque ratio law under different NN control. (**a**–**c**) show the proportion of Joint 1. (**d**–**f**) show the proportion of Joint 2.

## 6. Conclusions

In this paper, a fixed-time adaptive tracking control scheme with actuator saturation was proposed for the manipulator system under the composite learning framework, combined with the INN control scheme. A new incremental neural network adaptive algorithm based on compound learning was designed, so that the estimation error information could be properly integrated into the adaptive law, which improved the estimation performance and relaxed the PE condition. At the same time, a hyperbolic tangent function was designed to solve the problem of asymmetric input saturation, and a backstepping recursive fixed-time control scheme was employed to design the controller. In addition, the stability analysis was carried out by using the fixed-time Lyapunov theory, which not only suppressed the input saturation, but also ensured the fixed-time convergence of all signals. Our future work will extend the presented scheme to a dual-arm robot and verify the proposed control approach by using the Baxter robot.

**Author Contributions:** Conceptualization, Y.F. and C.Y.; methodology, Y.F. and H.H.; software, Y.F. and H.H.; validation, Y.F., H.H. and C.Y.; resources, C.Y.; writing—original draft preparation, Y.F. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported in part by the National Nature Science Foundation of China (NSFC) under Grant U20A20200 and the Major Research Grant No. 92148204, in part by Guangdong Basic and Applied Basic Research Foundation under Grants 2019B1515120076 and 2020B1515120054, and in part by the Industrial Key Technologies R & D Program of Foshan under Grant 2020001006308 and Grant 2020001006496.

Data Availability Statement: Not applicable.

**Conflicts of Interest:** The authors declare no conflict of interest.

### References

- 1. Li, Z.; Huang, B.; Ajoudani, A.; Yang, C.; Su, C.; Bicchi, A. Asymmetric bimanual control of dual-arm exoskeletons for human-cooperative manipulations. *IEEE Trans. Robot.* **2017**, *34*, 264–271. [CrossRef]
- Li, R.; Qiao, H. A survey of methods and strategies for high-precision robotic grasping and assembly tasks—Some new trends. IEEE/ASME Trans. Mechatron. 2019, 24, 2718–2732. [CrossRef]
- Zeng, C.; Yang, C.; Chen, Z. Bio-inspired robotic impedance adaptation for human-robot collaborative tasks. *Sci. China Inf. Sci.* 2020, 63, 170201. [CrossRef]
- Lewis, F.L.; Liu, K.; Yesildirek, A. Neural net robot controller with guaranteed tracking performance. *IEEE Trans. Neural Netw.* 1995, 6, 703–715. [CrossRef] [PubMed]
- Chen, Z.; Liu, Y.; He, W.; Qiao, H.; Ji, H. Adaptive-neural-network-based trajectory tracking control for a nonholonomic wheeled mobile robot with velocity constraints. *IEEE Trans. Ind. Electron.* 2020, *68*, 5057–5067. [CrossRef]

- 6. Yang, C.; Chen, C.; He, W.; Cui, R.; Li, Z. Robot learning system based on adaptive neural control and dynamic movement primitives. *IEEE Trans. Neural Netw. Learn. Syst.* 2018, 30, 777–787. [CrossRef]
- Zhao, K.; Song, Y. Neuroadaptive robotic control under time-varying asymmetric motion constraints: A feasibility-condition-free approach. *IEEE Trans. Cybern.* 2018, 50, 15–24. [CrossRef]
- Ma, Z.; Huang, P.; Kuang, Z. Fuzzy approximate learning-based sliding mode control for deploying tethered space robot. *IEEE Trans. Fuzzy Syst.* 2020, 29, 2739–2749. [CrossRef]
- Su, H.; Qi, W.; Chen, J.; Zhang, D. Fuzzy Approximation-based Task-Space Control of Robot Manipulators with Remote Center of Motion Constraint. *IEEE Trans. Fuzzy Syst.* 2022, 30, 1564–1573. [CrossRef]
- 10. Yang, C.; Jiang, Y.; Na, J.; Li, Z.; Cheng, L.; Su, C. Finite-time convergence adaptive fuzzy control for dual-arm robot with unknown kinematics and dynamics. *IEEE Trans. Fuzzy Syst.* **2018**, *27*, 574–588. [CrossRef]
- 11. Huang, H.; Zhang, T.; Yang, C.; Chen, C.P. Motor learning and generalization using broad learning adaptive neural control. *IEEE Trans. Ind. Electron.* **2019**, *67*, 8608–8617. [CrossRef]
- 12. Miao, Z.; Liu, Y.; Wang, Y.; Chen, H.; Zhong, H.; Fierro, R. Consensus with persistently exciting couplings and its application to vision-based estimation. *IEEE Trans. Cybern.* **2019**, *51*, 2801–2812. [CrossRef]
- 13. Wang, C.; Hill, D.J. Learning from neural control. *IEEE Trans. Neural Netw.* 2006, 17, 130–146. [CrossRef]
- 14. Yang, C.; Teng, T.; Xu, B.; Li, Z.; Na, J.; Su, C. Global adaptive tracking control of robot manipulators using neural networks with finite-time learning convergence. *Int. J. Control Autom. Syst.* **2017**, *15*, 1916–1924. [CrossRef]
- 15. Wang, C.; Wang, M.; Liu, T.; Hill, D.J. Learning from ISS-modular adaptive NN control of nonlinear strict-feedback systems. *IEEE Trans. Neural Netw. Learn. Syst.* **2012**, *23*, 1539–1550. [CrossRef]
- Pan, Y.; Sun, T.; Liu, Y.; Yu, H. Composite learning from adaptive backstepping neural network control. *Neural Netw.* 2017, 95, 134–142. [CrossRef]
- 17. Guo, K.; Pan, Y.; Yu, H. Composite learning robot control with friction compensation: A neural network-based approach. *IEEE Trans. Ind. Electron.* **2018**, *66*, 7841–7851. [CrossRef]
- Chen, M.; Ge, S.S.; Ren, B. Adaptive tracking control of uncertain MIMO nonlinear systems with input constraints. *Automatica* 2011, 47, 452–465. [CrossRef]
- 19. Bai, W.; Zhou, Q.; Li, T.; Li, H. Adaptive reinforcement learning neural network control for uncertain nonlinear system with input saturation. *IEEE Trans. Cybern.* **2019**, *50*, 3433–3443. [CrossRef]
- 20. Wen, C.; Zhou, J.; Liu, Z.; Su, H. Robust adaptive control of uncertain nonlinear systems in the presence of input saturation and external disturbance. *IEEE Trans. Autom. Control* 2011, *56*, 1672–1678. [CrossRef]
- 21. Zhou, Q.; Wang, L.; Wu, C.; Li, H.; Du, H. Adaptive fuzzy control for nonstrict-feedback systems with input saturation and output constraint. *IEEE Trans. Syst. Man Cybern. Syst.* **2016**, 47, 1–12. [CrossRef]
- He, W.; Dong, Y.; Sun, C. Adaptive neural impedance control of a robotic manipulator with input saturation. *IEEE Trans. Syst. Man Cybern. Syst.* 2015, 46, 334–344. [CrossRef]
- Du, J.; Hu, X.; Krstić, M.; Sun, Y. Robust dynamic positioning of ships with disturbances under input saturation. *Automatica* 2016, 73, 207–214. [CrossRef]
- 24. Li, Y.; Wang, T.; Liu, W.; Tong, S. Neural network adaptive output-feedback optimal control for active suspension systems. *IEEE Trans. Syst. Man Cybern. Syst.* 2022, 52, 4021–4032. [CrossRef]
- Bhat, S.P.; Bernstein, D.S. Finite-time stability of continuous autonomous systems. SIAM J. Control Optim. 2000, 38, 751–766. [CrossRef]
- 26. Cui, B.; Xia, Y.; Liu, K.; Shen, G. Finite-time tracking control for a class of uncertain strict-feedback nonlinear systems with state constraints: A smooth control approach. *IEEE Trans. Neural Netw. Learn. Syst.* **2020**, *31*, 4920–4932. [CrossRef]
- Wang, F.; Chen, B.; Lin, C.; Zhang, J.; Meng, X. Adaptive neural network finite-time output feedback control of quantized nonlinear systems. *IEEE Trans. Cybern.* 2017, 48, 1839–1848. [CrossRef]
- Fan, Y.; Li, Y.; Tong, S. Adaptive finite-time fault-tolerant control for interconnected nonlinear systems. *Int. J. Robust Nonlinear Control* 2021, 31, 1564–1581. [CrossRef]
- Lv, M.; Li, Y.; Pan, W.; Baldi, S. Finite-time fuzzy adaptive constrained tracking control for hypersonic flight vehicles with singularity-free switching. *IEEE/ASME Trans. Mechatron.* 2022, 27, 1594–1605. [CrossRef]
- Polyakov, A. Nonlinear feedback design for fixed-time stabilization of linear control systems. *IEEE Trans. Autom. Control* 2011, 57, 2106–2110. [CrossRef]
- Jin, X. Adaptive fixed-time control for MIMO nonlinear systems with asymmetric output constraints using universal barrier functions. *IEEE Trans. Autom. Control* 2018, 64, 3046–3053. [CrossRef]
- Zhang, Y.; Wang, F. Observer-based fixed-time neural control for a class of nonlinear systems. *IEEE Trans. Neural Netw. Learn.* Syst. 2021, 33, 2892–2902. [CrossRef]
- Huang, C.; Liu, Z.; Chen, C.P.; Zhang, Y. Adaptive Fixed-Time Neural Control for Uncertain Nonlinear Multiagent Systems. *IEEE Trans. Neural Netw. Learn. Syst.* 2022, *early access.* [CrossRef]

- 34. He, W.; Kang, F.; Kong, L.; Feng, Y.; Cheng, G.; Sun, C. Vibration Control of a Constrained Two-Link Flexible Robotic Manipulator With Fixed-Time Convergence. *IEEE Trans. Cybern.* **2021**, *52*, 5973–5983. [CrossRef]
- Huang, H.; Lu, Z.; Wang, N.; Yang, C. Fixed-time Adaptive Neural Control for Robot Manipulators with Input Saturation and Disturbance. In Proceedings of the 2022 27th International Conference on Automation and Computing (ICAC), Bristol, UK, 1–3 September 2022; p. 22136904.