This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

Neural Networks-Based Fault Tolerant Control of a Robot via Fast Terminal Sliding Mode

Shuang Zhang, *Member, IEEE*, Pengxin Yang, Linghuan Kong, Wenshi Chen, Qiang Fu, and Kaixiang Peng, *Member, IEEE*

***Abstract*—This article develops a robust fault tolerant (FT) control scheme for an *n*-link uncertain robotic system with actuator failures. In order to eliminate the influence of both the uncertainties and actuator failures on the system performance, the Gaussian radial basis function neural networks are used to compensate for the actuator failures and uncertain dynamics. An adaptive observer is designed to compensate for external disturbance. In addition, in order to accelerate the recovery of system stability after failure, a nonsingular fast terminal sliding mode is given. Finally, the simulation results on a two-link manipulator confirms the superior performance of the proposed neural networks-based FT controller, and the experiment results on the Baxter robot further verify the effectiveness of the control method.**

***Index Terms*—Adaptive observer, fault tolerant (FT) control, neural networks (NNs), nonsingular fast terminal sliding mode (NFTSM).**

I. Introduction

NOWADAYS, with the progress of robotic technology, more and more employees in high-risk industries are being replaced by robots. Due to the complex and changeable environment of these industries, system failures are very easy to occur. For purpose of ensuring the safe, efficient, and reliable operation of the robotic system, fault tolerant (FT) performance has inevitably become an indispensable part of the robotic design process. Failure usually refers to a condition in which a device fails to perform its specified functions under specified conditions.

Manuscript received January 23, 2019; revised April 23, 2019; accepted July 18, 2019. This work was supported in part by the National Natural Science Foundation of China under Grant 61873297, Grant 61933001, and Grant 61873024, in part by the China Post-Doctoral Science Foundation under Grant 2019T120048 and Grant 2018M630074, in part by the Joint Funds of Equipment Pre-Research and Ministry of Education of China under Grant 6141A02033339, and in part by the Fundamental Research Funds for the China Central Universities under Grant FRF-GF-18-027B. This article was recommended by Associate Editor H. Li. *(Corresponding author: Shuang Zhang.)*

The authors are with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing 100083, China, also with the Key Laboratory of Knowledge Automation for Industrial Processes, Ministry of Education, University of Science and Technology Beijing, Beijing 100083, China, and also with the Institute of Artificial Intelligence, University of Science and Technology Beijing, Beijing 100083, China (e-mail: zhangshuang.ac@gmail.com).

Color versions of one or more of the figures in this article are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TSMC.2019.2933050

Due to the large application in industry, numerous methods have been proposed for robotics [1]–[14], such as cooperative control methods [15]–[17], impedance control methods [18]–[21], fuzzy control methods [22]–[26], iterative learning control [27], admittance control method [28], model predictive control method [29], optimum control methods [30]–[34], adaptive methods [35]–[42], and NNs methods [43]–[54]. In these methods, adaptive methods and NNs methods have been widely used in FT control [55]. The former can adjust parameters online and adapt to the environment, while the latter can model the unknown robotic dynamics online. Step phenomenon occurs when the abrupt faults of the system happened. In [56], for purpose of ensuring the constraint on system states, the authors used the barrier Lyapunov function to constrain the system states, and developed an adaptive method to compensate for the external disturbance and the parameter uncertainties. For purpose of improving the dynamic performance of the uncertain nonlinear system, a new adaptive control design idea was proposed in [57]. The authors designed a constrained function that was smooth and strictly increasing. When a failure occurred, the change of parameters would be limited to a certain range, so as to meet the performance requirement. In order to decrease the errors of actuator failures system with disturbance and uncertainties, Chen *et al.* [58] took the unknown external disturbance and the unknown neural network approximation errors as a compound disturbance, then an nonlinear disturbance observer was designed to estimate the composite disturbance. The above control schemes are based on the assumption that all the states are measurable. For the case where velocity signals are immeasurable, Tong *et al.* [59] used the fuzzy logic system to observe the system state, and then used the backstepping method to design the adaptive controller. In case of failure, the system can be stabilized as soon as possible through self-tuning parameters.

In addition to the above methods, sliding mode control [60]–[64] has been widely applied to FT control and achieved satisfactory results, since it is insensitive to parameter uncertainties and external disturbance and has strong robustness. However, the sliding mode control has the defects of chattering, which should be considered when the control strategy is applied to the fault tolerance of the system. In [65], aiming at the actuator failures, the authors combined robust control strategy and adaptive control strategy with sliding mode control strategy, respectively. Then, the chattering was effectively reduced. In addition, with regard to an *n*-link robotic system model based on Euler–Lagrange equation, Van and Kang [66] designed a controller, which adopted the adaptive quasi-continuous second-order sliding mode control method to solve the system failures and uncertainties. Meanwhile, the existence of the second-order sliding mode reduced the chattering defects. Similarly, to handle the disturbance and actuator failures of the uncertain robotic system, Van *et al.* [67] used the high-order sliding mode control method based on the super-twist algorithm, which reduced the chattering at the sliding mode surface and ensured the stability of the uncertain robotic system. In addition, the terminal sliding mode has also achieved satisfactory results in FT control. Van [68] combined PID with NFTSM for the uncertain nonlinear system to accelerate the convergence speed of the system and shorten the chattering time to a certain extent.

The above methods of NNs, adaptive control and sliding mode control prompt my further research. Aiming at stabilizing the two-link manipulator system with parameter uncertainties, system failure and external disturbance, this article proposed an FT control scheme that combining NFTSM, adaptive control and radial basis function neural networks (RBFNNs). The specific contributions are as follows.

1) The NFTSM surface is used to accelerate the convergence rate and enhance the robustness of the robotic system.
2) The adaptive observer is employed to compensate for the unknown external disturbance of the robotic system. By online estimation, the observation error is reduced.
3) With the strong approximation ability of the NNs based on Gaussian kernel function, the parameter uncertainties and actuator failures of the system are estimated and compensated, and the diagnosis error is avoided.

The structure of this article is concluded as follows. The dynamics of an n-link rigid robotic system is given in Section II. Section III describes the design process of controller. In Section IV, the simulation results of three controllers are compared to show the validity and feasibility of the proposed control strategy. Section V describes the experiment setup and results.

II. Problem Formulation

A. Radial Basis Function Neural Networks

In this article, due to the system uncertainties and actuator failures, the RBFNNs are given to approximate the continuous function

$$\begin{aligned} &f_i(N) : \mathbb{R}^q \to \mathbb{R} \\ &f_i(N) = W_i^T S_i(N), \;\; i = 1, 2, \ldots, n \end{aligned} \tag{1}$$

where the weight vector $W_i \in \mathbb{R}^l$, and NNs node number $l > 1$, the input vector $N = [N_1, N_2, \ldots, N_q]^T \in \Omega_N \subset \mathbb{R}^q$, and $S_i(N) = [s_1, s_2, \ldots, s_l]^T \in \mathbb{R}^l$. By the general approximation results, we can state that, when l is chosen large enough, $W_i S_i(N)$ can approximate any continuous function $f_i(N)$, to any desired accuracy over a compact set $\Omega_N \subset \mathbb{R}^q$. This is achieved as

$$\begin{aligned} f_i(N) = W_i^{*T} S_i(N) + \varepsilon_i(N) \quad \forall N \in \Omega_Z \subset \mathbb{R}^q \\ i = 1, 2, \ldots, n \end{aligned} \tag{2}$$

where W_i^* is the ideal constant weight vector, and $\varepsilon_i(N)$ is the approximation error that is bounded, i.e., $|\varepsilon_i(N)| \leq \bar{\varepsilon}_i \; \forall N \in \Omega_N$ with $\bar{\varepsilon}_i > 0$ is an unknown constant. The constant weight vector W_i^* is an "artificial" quantity required for Lyapunov stability proofs, and it is defined as the value of W_i that minimizes $|\varepsilon_i|$ for all $N \in \Omega_N \subset \mathbb{R}^q$, i.e.,

$$W_i^* = \arg \min_{W_i \in \mathbb{R}^l} \left\{ \sup_{N \in \Omega_N} \left| f_i(N) - W_i^T S_i(N) \right| \right\}. \tag{3}$$

Then, the l Gaussian kernel function is shown as

$$s_j(N) = \exp\left[\frac{-\left(N - \mu_j\right)^T \left(N - \mu_j\right)}{\eta_j^2} \right], \; j = 1, 2, \ldots, l \tag{4}$$

where $\mu_j = [\mu_{j1}, \mu_{j2}, \ldots, \mu_{jq}]^T$ is the center of the receptive field and η_j is the width of the Gaussian function [69], [70].

Remark 1: Comparing with other NN structures, e.g., back propagation NNs, hopfield NNs, and multilayer NNs, RBFNNs belong to a class of linearly parameterized NNs whose structure are simpler. RBFNNs possess a faster learning speed and are said to approximate any nonlinear function to any accuracy [71]. This type of network structure is more suitable for the approximation of nonlinear functions and can satisfy the requirement of FT control better.

B. Mathematic Tools

Assumption 1: The disturbance $d(t)$ is continuous and upper bounded [66], its rate of change is also upper bounded [68], i.e., $\exists \bar{d} > 0$, such that $\forall t > 0$, $|d(t)| \leq \bar{d}$, $\exists b_1 > 0$, such that $\forall t > 0$, and $|(d/dt)d(t)| \leq b_1$.

Assumption 2: The actuator failure $\phi(t)$ is assumed to be upper bounded, i.e., $\exists \bar{\phi} > 0$, such that $\forall t > 0$ and $|\phi(t)| \leq \bar{\phi}$ [66].

Remark 2: $\lambda_{\min}(\bullet)$ and $\lambda_{\max}(\bullet)$ are the minimum and maximum eigenvalues of matrix $\bullet$, respectively.

Remark 3: a, b, and c are n-dimensional vectors, and $|a|^r \in \mathbb{R}^n$, $\text{sgn}(b) \in \mathbb{R}^n$ and $c^t \in \mathbb{R}^n$ are defined as

$$|a|^r = \left[|a_1|^r, |a_2|^r, \ldots, |a_n|^r \right]^T \tag{5}$$

$$\text{sgn}(b) = \left[\text{sign}(b_1), \text{sign}(b_2), \ldots, \text{sign}(b_n) \right]^T \tag{6}$$

$$c^t = \left[|c_1|^t \text{sign}(c_1), |c_2|^t \text{sign}(c_2), \ldots, |c_n|^t \text{sign}(c_n) \right]^T. \tag{7}$$

Remark 4: d and e are n-dimensional vectors, and $d \odot e$ is defined as

$$d \odot e = [d_1 e_1, d_2 e_2, \ldots, d_n e_n]^T. \tag{8}$$

Lemma 1: Consider an NFTSM surface [72]

$$s_1 = z_1 + k_1 z_1{}^{\lambda} + k_2 z_2{}^{p/q}. \tag{9}$$

If $s_1 = 0$, the convergence time T of $z_1(t)$ is given as follows:

$$\begin{aligned} T = \int_0^{|z_1(0)|} \frac{k_2{}^{q/p}}{\left(z_1(t) + k_1 z_1{}^{\lambda}\right)^{q/p}} dz_1 = \frac{\frac{p}{q} |z_1(0)|^{1 - q/p}}{k_1 \left(\frac{p}{q} - 1\right)} \\ \times F\left(\frac{q}{p}, \frac{\frac{p}{q} - 1}{(\lambda - 1)\frac{p}{q}}; 1 + \frac{\frac{p}{q} - 1}{(\lambda - 1)\frac{p}{q}}; -k_1 |z_1(0)|^{\lambda - 1} \right) \end{aligned} \tag{10}$$

where $z_1(0)$ represents the initial value of $z_1(t)$, and $F(\bullet)$ represents the Gauss' Hypergeometric function.

C. System Model

The dynamics model with the actuator failures can be described as follows:

$$M(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) = \tau + d(t) + \phi(q,\dot{q},\tau) \quad (11)$$

where $q, \dot{q}$, and $\ddot{q} \in \mathbb{R}^n$ represent the position, velocity, and acceleration, respectively; $\tau \in \mathbb{R}^n$ is the control input; $M(q) \in \mathbb{R}^{n\times n}$ denotes a positive definite symmetric inertia matrix; $C(q,\dot{q}) \in \mathbb{R}^{n\times n}$ is a centripetal and Coriolis matrix; and $G(q) \in \mathbb{R}^n$ denotes the force of gravity; $d(t) \in \mathbb{R}^n$ is a disturbance vector, and $\phi(q,\dot{q},\tau) \in \mathbb{R}^n$ is a vector that represents the actuator failures of the system dynamics. Then, we use the M, C, G, d, ϕ as the shorthand notations in the subsequent design.

Property 1: $\dot{M}(q) - 2C(q,\dot{q})$ is skew-symmetric [73].

III. Control Design

In this article, a type of actuator failures called loss-of-effectiveness is considered. It means that the performance of the controller cannot be fully exploited due to the failure. The control objective is that the system can still track a desired trajectory x_d well when the actuator failures occur. In the following, we consider two situations for the control design. First, when the robotic parameters are all known, the model-based FT controller is designed. Then, when the system parameters are unknown, the NNs-based FT controller is designed which uses NNs to estimate the system uncertainties.

A. Model-Based FT Method

Let $x_1 = q$, $x_2 = \dot{q}$, the robotic dynamics (11) is rewritten as

$$\dot{x}_1 = x_2 \quad (12)$$

$$\dot{x}_2 = M^{-1}(\tau + d + \phi - Cx_2 - G). \quad (13)$$

Then, the tracking error z_1 and the second error z_2 are defined as follows:

$$z_1 = x_1 - x_d \quad (14)$$

$$z_2 = x_2 - \alpha_1 \quad (15)$$

where $\alpha_1 = -K_1 z_1 + \dot{x}_d$ is a virtual control. Differentiating z_1, we obtain

$$\dot{z}_1 = \dot{x}_1 - \dot{x}_d = x_2 - \dot{x}_d = z_2 + \alpha_1 - \dot{x}_d = -K_1 z_1 + z_2. \quad (16)$$

The time derivative of z_2 is

$$\dot{z}_2 = M^{-1}(\tau + d + \phi - Cx_2 - G) - \dot{\alpha}_1. \quad (17)$$

The Lyapunov function candidate is proposed

$$V_1 = \frac{1}{2} z_1^T z_1. \quad (18)$$

Differentiating V_1 yields

$$\dot{V}_1 = z_1^T \dot{z}_1 = z_1^T(-K_1 z_1 + z_2) = -z_1^T K_1 z_1 + z_1^T z_2. \quad (19)$$

Choosing an NFTSM sliding surface

$$s_1 = z_1 + k_1 {z_1}^{\lambda} + k_2 {z_2}^{p/q} \quad (20)$$

where $k_1 = \mathrm{diag}(k_{11}, k_{12}, \ldots, k_{1n})$ and $k_2 = \mathrm{diag}(k_{21}, k_{22}, \ldots, k_{2n})$ are positive matrices, respectively, p and q are positive odd numbers satisfying the relations $1 < p/q < 2$ and $\lambda > p/q$. Then $\dot{s}_1$ becomes

$$\dot{s}_1 = \dot{z}_1 + k_1 \lambda \dot{z}_1 \odot |z_1|^{\lambda-1} + k_2 \frac{p}{q} \dot{z}_2 \odot |z_2|^{(p/q)-1}. \quad (21)$$

A new Lyapunov function candidate is proposed

$$V_2 = V_1 + \frac{1}{2} s_1^T M s_1. \quad (22)$$

Differentiating V_2, we have

$$\begin{aligned}
\dot{V}_2 &= \dot{V}_1 + s_1^T M \dot{s}_1 + \frac{1}{2} s_1^T \dot{M} s_1 \\
&= -z_1^T K_1 z_1 + z_1^T z_2 + \frac{1}{2} s_1^T \dot{M} s_1 \\
&\quad + s_1^T M \left(\dot{z}_1 + k_1 \lambda \dot{z}_1 \odot |z_1|^{\lambda-1} + k_2 \frac{p}{q} \dot{z}_2 \odot |z_2|^{(p/q)-1} \right) \\
&= -z_1^T K_1 z_1 + z_1^T z_2 + s_1^T \left(M\dot{z}_1 + M k_1 \lambda \dot{z}_1 \odot |z_1|^{\lambda-1} \right) \\
&\quad + s_1^T k_2 \frac{p}{q} (\tau + d + \phi - Cx_2 - G - M\dot{\alpha}_1) \odot |z_2|^{(p/q)-1} \\
&\quad + s_1^T C s_1.
\end{aligned} \quad (23)$$

A model-based FT controller can be designed as

$$\tau = \tau_1 + \tau_2 + \tau_3 \quad (24)$$

where

$$\tau_1 = Cx_2 + G + M\dot{\alpha}_1 - \bar{d}\,\mathrm{sgn}(s_1) - \bar{\phi}\,\mathrm{sgn}(s_1) \quad (25)$$

$$\tau_2 = -\frac{q}{p} {k_2}^{-1} \left(M\dot{z}_1 + Cs_1 + M k_1 \lambda \dot{z}_1 \odot |z_1|^{\lambda-1} \right) \odot |z_2|^{1-(p/q)} \quad (26)$$

$$\tau_3 = -\frac{q}{p} {k_2}^{-1} \left(\frac{s_1}{\|s_1\|^2} z_1^T z_2 + \eta |s_1| \mathrm{sgn}(s_1) \right) \odot |z_2|^{1-(p/q)} \quad (27)$$

and η is a positive constant. Substituting (24) into (23), it yields

$$\begin{aligned}
\dot{V}_2 &= -z_1^T K_1 z_1 + s_1^T \big(d - \bar{d}\,\mathrm{sgn}(s_1) + \phi - \bar{\phi}\,\mathrm{sgn}(s_1) \big) \\
&\quad - s_1^T \eta |s_1| \mathrm{sgn}(s_1) \\
&\le -z_1^T K_1 z_1 - \eta \|s_1\|^2
\end{aligned} \quad (28)$$

when $z_1 \neq 0$ or $s_1 \neq 0$, we can get $\dot{V}_2 < 0$. If and only if $z_1 = 0$ and $s_1 = 0$, $V_2 = 0$. Therefore, the stabilization and astringency of the robotic system are guaranteed.

From Lemma 1, since the NFTSM surface has the property of fast convergence, we can state that the controller can accelerate the convergence speed of the system.

In view of the assumption that the upper boundary of the disturbance can be gained, we design the above controller. When the upper boundary of external disturbance is unknown. We define the disturbance observer as $\hat{d}$. Then, the estimation error is given as

$$\tilde{d} = d - \hat{d}. \quad (29)$$

The time derivative of (29) is

$$\dot{\tilde{d}} = \dot{d} - \dot{\hat{d}}. \quad (30)$$

Consider a Lyapunov function candidate

$$V_3 = V_2 + \frac{1}{2}\tilde{d}^T \Gamma_1^{-1} \tilde{d} \tag{31}$$

where $\Gamma_1 = \mathrm{diag}(\Gamma_{11}, \Gamma_{12}, \ldots, \Gamma_{1n})$ is a positive definite matrix. Differentiating (31), we have

$$\begin{aligned}\dot{V}_3 &= \dot{V}_2 + \tilde{d}^T \Gamma_1^{-1} \dot{\tilde{d}} \\ &\le -z_1^T K_1 z_1 + z_1^T z_2 + \tilde{d}^T \Gamma_1^{-1} b_1 \\ &\quad + s_1^T k_2 \frac{p}{q}\Big(\tau + \hat{d} + \phi - Cx_2 - G - M\dot{\alpha}_1\Big) \odot |z_2|^{(p/q)-1} \\ &\quad + \sum_{i=1}^{n} \tilde{d}_i \Gamma_{1i}^{-1}\left(\Gamma_{1i} s_{1i} k_{2i} \frac{p}{q}|z_{2i}|^{(p/q)-1} - \dot{\hat{d}}_i\right) \\ &\quad + s_1^T\Big(M\dot{z}_1 + Cs_1 + Mk_1\lambda\dot{z}_1 \odot |z_1|^{\lambda-1}\Big).\end{aligned} \tag{32}$$

Therefore, the controller can be designed as

$$\tau = \tau_1 + \tau_2 + \tau_3 \tag{33}$$

where

$$\tau_1 = Cx_2 + G + M\dot{\alpha}_1 - \hat{d} - \bar{\phi}\,\mathrm{sgn}(s_1) \tag{34}$$

and τ_2 and τ_3 are designed as the same as (26) and (27), and the adaption law is given as

$$\dot{\hat{d}}_i = \Gamma_{1i}\left(s_{1i} k_{2i} \frac{p}{q}|z_{2i}|^{(p/q)-1} - \delta_{1i}\hat{d}_i\right) \tag{35}$$

where the designed parameters Γ_{1i} and δ_{1i} are two positive constants.

Substituting (33) and (35) into (32), we have

$$\begin{aligned}\dot{V}_3 &\le -z_1^T K_1 z_1 + s_1^T\big(\phi - \bar{\phi}\,\mathrm{sgn}(s_1)\big) \odot |z_2|^{(p/q)-1} - \eta\|s_1\|^2 \\ &\quad + \sum_{i=1}^{n} \tilde{d}_i \Gamma_{1i}^{-1} b_{1i} + \sum_{i=1}^{n} \tilde{d}_i \delta_{1i} \hat{d}_i \\ &\le -z_1^T K_1 z_1 - \eta\|s_1\|^2 - \frac{1}{2}\sum_{i=1}^{n}\Big(\delta_{1i} - \Gamma_{1i}^{-1}\Big)\tilde{d}_i^2 \\ &\quad + \frac{1}{2}\sum_{i=1}^{n}\Gamma_{1i}^{-1} {b_{1i}}^2 + \frac{1}{2}\sum_{i=1}^{n}\delta_{1i}\bar{d}_i^2 \\ &\le -\rho_1 V_3 + c_1\end{aligned} \tag{36}$$

where ρ_1 and c_1 are two positive constants given as

$$\rho_1 = \min\left(2\lambda_{\min}(K_1), \frac{2\eta}{\lambda_{\max}(M(x_1))}, \min_{i=1,2,\ldots,n} \frac{\delta_{1i} - \Gamma_{1i}^{-1}}{\lambda_{\max}\Big(\Gamma_1^{-1}\Big)}\right) \tag{37}$$

$$c_1 = \frac{1}{2}\sum_{i=1}^{n}\Gamma_{1i}^{-1}{b_{1i}}^2 + \frac{1}{2}\sum_{i=1}^{n}\delta_{1i}\bar{d}_i^2. \tag{38}$$

To ensure $\rho_1 > 0$, the gains δ_{1i} and Γ_{1i} are selected to fulfill

$$\min_{i=1,2,\ldots,n}\Big(\delta_{1i} - \Gamma_{1i}^{-1}\Big) > 0. \tag{39}$$

Theorem 1: Considering the manipulator system represented by (11) with uncertainties, external disturbance, and actuator failures which fulfills the Assumptions 1 and 2, under the controller (33) and the disturbance observer adaption law (35) with bounded initial conditions, the signals z_1, s_1, and $\tilde{d}$ are semiglobally bounded. In addition, the error signals z_1, s_1, as well as $\tilde{d}$ will remain within the compact sets Ω_{z_1}, Ω_{s_1}, and $\Omega_{\tilde{d}}$, respectively, defined by

$$\Omega_{z_1} = \left\{z_1 \in \mathbb{R}^n \mid \|z_1\| \le \sqrt{P}\right\} \tag{40}$$

$$\Omega_{s_1} = \left\{s_1 \in \mathbb{R}^n \mid \|s_1\| \le \sqrt{\frac{P}{\lambda_{\min}(M)}}\right\} \tag{41}$$

$$\Omega_{\tilde{d}} = \left\{\tilde{d} \in \mathbb{R}^n \mid \left\|\tilde{d}\right\| \le \sqrt{\frac{P}{\lambda_{\min}\big({\Gamma_{1i}}^{-1}\big)}}\right\} \tag{42}$$

where $P = 2(V_3(0) + c_1/\rho_1)$ with ρ_1 and c_1 given in (37) and (38).

Proof: Multiplying both sides by $e^{\rho_1 t}$ in (36), we have

$$\begin{aligned}V_3(t) &\le \big(V_3(0) - c_1/\rho_1\big)e^{-\rho_1 t} + c_1/\rho_1 \\ &\le V_3(0) + c_1/\rho_1.\end{aligned} \tag{43}$$

Define $P = 2(V_3(0) + c_1/\rho_1)$, we have

$$\begin{aligned}z_1^T z_1 &\le P \\ s_1^T M(x_1) s_1 &\le P \\ \tilde{d}^T \Gamma_1^{-1} \tilde{d} &\le P.\end{aligned} \tag{44}$$

Thus, the following inequalities hold: $\|z_1\| \le \sqrt{P}$, $\|s_1\| \le \sqrt{P/\lambda_{\min}(M)}$, and $\|\tilde{d}\| \le \sqrt{P/\lambda_{\min}({\Gamma_{1i}}^{-1})}$. From above, we know z_1, s_1, as well as the approximation errors $\tilde{d}$ are bounded. ■

B. NNs-Based FT Method

Since M, C, and G are unknown, we cannot get exact parameters in an actual platform. But we can describe them by the RBFNNs

$$\begin{aligned}W^{*T}S(N) + \varepsilon &= (-Cx_2 - G - M\dot{\alpha}_1 + \phi) \odot |z_2|^{(p/q)-1} \\ &\quad + \frac{q}{p}{k_2}^{-1}\Big(M\dot{z}_1 + Cs_1 + Mk_1\lambda\dot{z}_1 \odot |z_1|^{\lambda-1}\Big)\end{aligned} \tag{45}$$

where $N = [x_1^T, x_2^T, s_1^T, \tau^{\mathrm{T}}]^{\mathrm{T}}$ is the input of the RBFNNs. ε is the NNs approximate error satisfying $|\varepsilon(t)| \le \bar{\varepsilon}$ with $\bar{\varepsilon}$ being an unknown positive constant. The RBFNNs control is proposed as

$$\begin{aligned}\tau = &-\frac{q}{p}{k_2}^{-1}\left(\frac{s_1}{\|s_1\|^2}z_1^T z_2 + \eta|s_1|\mathrm{sgn}(s_1)\right) \odot |z_2|^{1-(p/q)} \\ &- \hat{d} - \hat{W}^T S(N) \odot |z_2|^{1-(p/q)}\end{aligned} \tag{46}$$

where $\hat{W}$ is the estimation weight. Then, the adaptive law is designed as

$$\dot{\hat{W}}_i = \Gamma_{2i}\left(s_{1i}^T k_{2i}\frac{p}{q}S_i(N) - \delta_{2i}\hat{W}_i\right) \tag{47}$$

where the controller parameters Γ_{2i} and δ_{2i} are two positive constants.

Define $\tilde{W}_i = \hat{W}_i - W_i^*$. The Lyapunov function is proposed

$$V_4 = V_3 + \frac{1}{2}\sum_{i=1}^{n}\tilde{W}_i^T \Gamma_{2i}^{-1}\tilde{W}_i. \tag{48}$$

Taking the derivative of (48) and plugging (46) and (47) into it yields

$$\begin{aligned}\dot{V}_4 &= \dot{V}_3 + \sum_{i=1}^{n} \tilde{W}_i^T \Gamma_{2i}^{-1} \dot{\hat{W}}_i \\ &\le -z_1^T K_1 z_1 - \eta \|s_1\|^2 + s_1^T k_2 \frac{p}{q} \varepsilon - \sum_{i=1}^{n} \tilde{W}_i^T \delta_{2i} \hat{W}_i \\ &\quad - \frac{1}{2}\sum_{i=1}^{n}\left(\delta_{1i} - \Gamma_{1i}^{-1}\right)\tilde{d}_i^2 + \frac{1}{2}\sum_{i=1}^{n} \Gamma_{1i}^{-1} {b_{1i}}^2 + \frac{1}{2}\sum_{i=1}^{n} \delta_{1i}\bar{d}_i^2 \\ &\le -z_1^T K_1 z_1 - {s_1}^T\left(\eta I - \frac{p}{2q}k_2\right)s_1 - \frac{1}{2}\sum_{i=1}^{n} \tilde{W}_i^T \delta_{2i} \tilde{W} \\ &\quad + \frac{p}{2q}k_2 \bar{\varepsilon}^2 + \frac{1}{2}\sum_{i=1}^{n} {W_i^*}^T \delta_{2i} W_i^* + \frac{1}{2}\sum_{i=1}^{n} \Gamma_{1i}^{-1} {b_{1i}}^2 \\ &\quad + \frac{1}{2}\sum_{i=1}^{n} \delta_{1i}\bar{d}_i^2 - \frac{1}{2}\sum_{i=1}^{n}\left(\delta_{1i} - \Gamma_{1i}^{-1}\right)\tilde{d}_i^2 \\ &\le -\rho_2 V_4 + c_2 \end{aligned} \tag{49}$$

where ρ_2 and c_2 are two positive constants given as

$$\begin{aligned}\rho_2 = \min&\Bigg(2\lambda_{\min}(K_1), \frac{2\lambda_{\min}\left(\eta I - (p/2q)k_2\right)}{\lambda_{\max}(M(x_1))}, \\ &\min_{i=1,2,\ldots,n} \frac{\delta_{1i} - \Gamma_{1i}^{-1}}{\lambda_{\max}\left(\Gamma_1^{-1}\right)}, \min_{i=1,2,\ldots,n} \frac{\delta_{2i}}{\lambda_{\max}\left(\Gamma_2^{-1}\right)}\Bigg)\end{aligned} \tag{50}$$

$$\begin{aligned}c_2 &= \frac{p}{2q}k_2 \bar{\varepsilon}^2 + \frac{1}{2}\sum_{i=1}^{n} {W_i^*}^T \delta_{2i} W_i^* + \frac{1}{2}\sum_{i=1}^{n} \Gamma_{1i}^{-1} {b_{1i}}^2 \\ &\quad + \frac{1}{2}\sum_{i=1}^{n} \delta_{1i}\bar{d}_i^2.\end{aligned} \tag{51}$$

To ensure $\rho_2 > 0$, the gains η, δ_{1i}, and Γ_{1i} are chosen to satisfy

$$\lambda_{\min}\left(\eta I - (p/2q)k_2\right) > 0, \min_{i=1,2,\ldots,n}\left(\delta_{1i} - \Gamma_{1i}^{-1}\right) > 0. \tag{52}$$

Theorem 2: Considering the manipulator system represented by (11) with uncertainties, external disturbance, and actuator failures which fulfills the Assumption 1, under the controller (46), the NNs adaption law (47) and the disturbance observer adaption law (35) with bounded initial conditions, the signals z_1, s_1, $\tilde{d}$, and $\tilde{W}$ are semiglobally bounded. In addition, the error signals z_1, s_1, $\tilde{d}$, and $\tilde{W}$ will remain within the compact sets Ψ_{z_1}, Ψ_{s_1}, $\Psi_{\tilde{d}}$, and $\Psi_{\tilde{W}}$, respectively, defined by

$$\Psi_{z_1} = \left\{z_1 \in \mathbb{R}^n \middle| \|z_1\| \le \sqrt{Q}\right\} \tag{53}$$

$$\Psi_{s_1} = \left\{s_1 \in \mathbb{R}^n \middle| \|s_1\| \le \sqrt{\frac{Q}{\lambda_{\min}(M)}}\right\} \tag{54}$$

$$\Psi_{\tilde{d}} = \left\{\tilde{d} \in \mathbb{R}^n \middle| \left\|\tilde{d}\right\| \le \sqrt{\frac{Q}{\lambda_{\min}\left({\Gamma_{1i}}^{-1}\right)}}\right\} \tag{55}$$

$$\Psi_{\tilde{W}} = \left\{\tilde{W} \in \mathbb{R}^{l\times n} \middle| |\tilde{W}_i| \le \sqrt{\frac{Q}{\lambda_{\min}\left({\Gamma_{2i}}^{-1}\right)}}, i = 1, \ldots, n\right\} \tag{56}$$

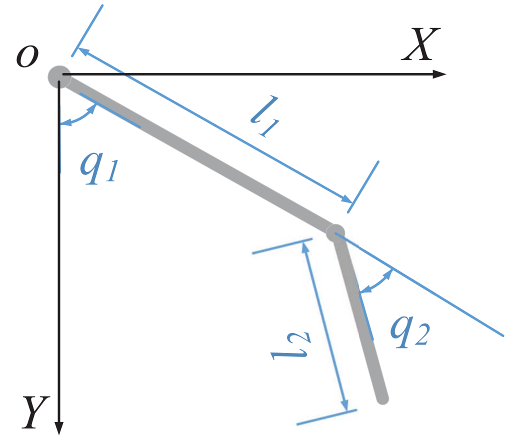


Fig. 1. 2-DOF planar manipulator.

where $Q = 2(V_4(0) + c_2/\rho_2)$ with ρ_2 and c_2 given in (50) and (51).

Proof: Multiplying both sides by $e^{\rho_2 t}$ in (49), we have

$$\begin{aligned}V_4(t) &\le \left(V_4(0) - c_2/\rho_2\right)e^{-\rho_2 t} + c_2/\rho_2 \\ &\le V_4(0) + c_2/\rho_2.\end{aligned} \tag{57}$$

Define $Q = 2(V_4(0) + c_2/\rho_2)$, we have

$$\begin{aligned} &z_1^T z_1 \le Q, s_1^T M(x_1) s_1 \le Q \\ &\tilde{d}^T \Gamma_1^{-1} \tilde{d} \le Q, \sum_{i=1}^{n} \tilde{W}_i^T \Gamma_{2i}^{-1} \tilde{W}_i \le Q.\end{aligned} \tag{58}$$

Thus, the following inequalities hold: $\|z_1\| \le \sqrt{Q}$, $\|s_1\| \le \sqrt{Q/\lambda_{\min}(M)}$, $\|\tilde{d}\| \le \sqrt{Q/\lambda_{\min}({\Gamma_{1i}}^{-1})}$, $|\tilde{W}_i| \le \sqrt{Q/\lambda_{\min}({\Gamma_{2i}}^{-1})}$. From above, we know z_1, s_1, as well as the approximation errors $\tilde{d}$, $\tilde{W}$ are bounded. ∎

IV. Simulation

In the section, for the sake of verifying the performance of the controller designed in Section III, we consider a two-link robotic manipulator shown in Fig. 1. It is assumed to move on the Cartesian space, then the position vector q can be rewritten as

$$q = \begin{bmatrix} q_1 \\ q_2 \end{bmatrix}. \tag{59}$$

For the dynamics model (11), we have

$$\begin{aligned} M(q) &= \begin{bmatrix} p_1 + p_2 + 2p_3 \cos q_2 & p_2 + p_3 \cos q_2 \\ p_2 + p_3 \cos q_2 & p_2 \end{bmatrix} \\ C(q, \dot{q}) &= \begin{bmatrix} -p_3 \dot{q}_2 \sin q_2 & -p_3(\dot{q}_1 + \dot{q}_2)\sin q_2 \\ p_3 \dot{q}_1 \sin q_2 & 0 \end{bmatrix} \\ G(q) &= \begin{bmatrix} p_4 g \cos q_1 + p_5 g \cos(q_1 + q_2) \\ p_5 g \cos(q_1 + q_2) \end{bmatrix} \end{aligned} \tag{60}$$

where

$$\begin{aligned} p_1 &= m_1 {l_{c1}}^2 + m_2 {l_1}^2 + I_1 \\ p_2 &= m_2 {l_{c2}}^2 + I_2 \\ p_3 &= m_2 l_1 l_{c2} \\ p_4 &= m_1 l_{c2} + m_2 l_1 \\ p_5 &= m_2 l_{c2} \end{aligned} \tag{61}$$

TABLE I
SYSTEM PARAMETERS

Parameter	Description	Value
m_1	Mass of link 1	2.00 kg
m_2	Mass of link 2	0.85 kg
l_1	Length of link 1	0.35 m
l_2	Length of link 2	0.31 m
I_1	Inertia of link 1	61.25×10^{-3} kgm^2
I_2	Inertia of link 2	20.42×10^{-3} kgm^2

TABLE II
DESIGN PARAMETERS

Observer	$\Gamma_1 = 0.5 I_{2\times2}, \sigma_1 = 100 I_{2\times2}$
NFTSM	$k_1 = 300 I_{2\times2}, k_2 = 30 I_{2\times2}, \lambda = 3, p = 13, q = 11$
Other Item	$\eta = 18, K_1 = 100 I_{2\times2}, \bar{\phi} = 30$

and

$$l_{c1} = \frac{1}{2} l_1, l_{c2} = \frac{1}{2} l_2. \tag{62}$$

The parameters of the robotic system are shown in Table I [73].

Then the original states of the robotic system are set as

$$q_1(0) = 0, q_2(0) = 0, \dot{q}_1(0) = 0, \dot{q}_2(0) = 0. \tag{63}$$

The desired trajectory is given as $x_d = q_d = [\sin(t) + \cos(t), \sin(t) + \cos(t)]^T$, where $t \in [0, t_f]$ and $t_f = 10$ s. The disturbance is given as $d = [\sin(t) + 1, 2\cos(t) + 0.5]^T$.

For the sake of simulating the effect of the actuator failures occurred in the system, ϕ is defined as

$$\phi = \begin{cases} [-0.5\tau(t), 0]^T, t \in [3, 5] \\ [-0.5\tau(t), 3\sin(t) + 2\cos(t) + 20]^T, t \in [6, 9] \end{cases} \tag{64}$$

where the first actuator will loss 50% its performance from 3 to 5 s as well as 6 to 9 s, and the second actuator will be affected by another type of failure $\phi_2 = 3\sin(t) + 2\cos(t) + 20$ from 6 to 9 s.

In the simulation section, we consider three cases. First, a model-based FT control (33) is taken into consideration. Second, the NNs-based FT control (46) is carried out. Third, we design a proportional differential (PD) controller. It is given to be compared with the above control effect.

A. Model-Based FT Control

The designed parameters of the controller is shown in Table II.

The position tracking performances and trajectory errors of q_1 and q_2 are shown in Figs. 2 and 3, respectively. The control input is shown in Fig. 4. It can be seen from Figs. 2 and 3, there are large fluctuations in error at $t = 3$s and $t = 6$s, since $t = 3$s and $t = 6$s are the turning points. The fluctuation is more obvious, since both of actuator failures occur at $t = 6$s.

B. NNs-Based FT Control

For the NNs-based FT control, the parameters of the observer and NFTSM and η, K_1 are same as the model-based FT control method. The NNs parameters of the controller are given as

$$\Gamma_2 = 100 I_{256\times256}, \sigma_2 = 0.2 I_{2\times2}. \tag{65}$$

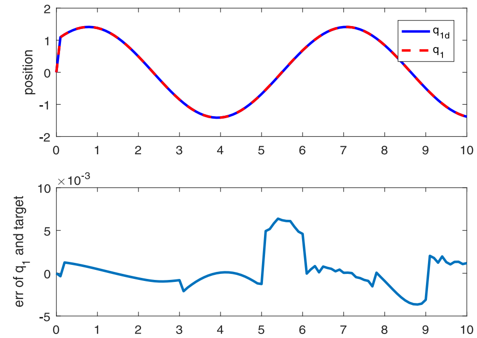


Fig. 2. Tracking performance and error of q_1 with model-based FT control.

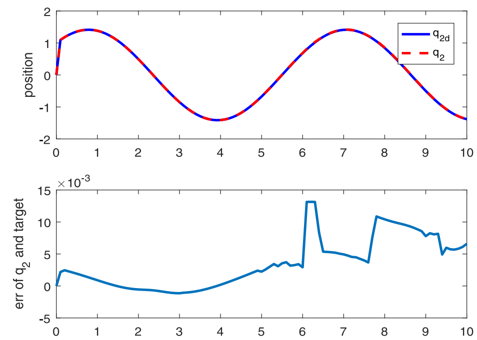


Fig. 3. Tracking performance and error of q_2 with model-based FT control.

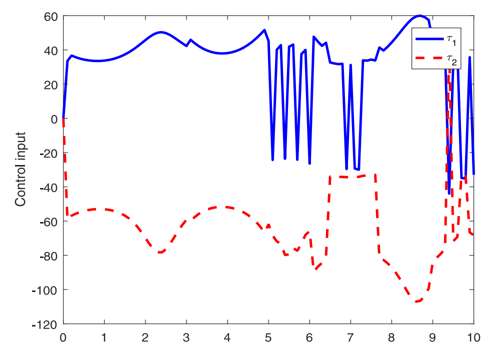


Fig. 4. Model-based FT control input.

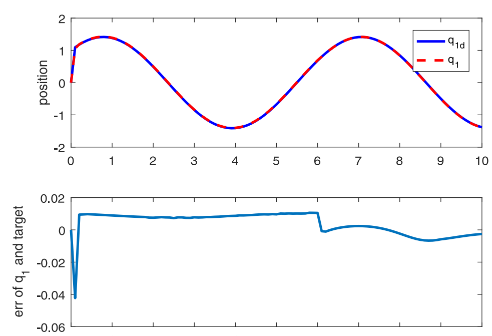


Fig. 5. Tracking performance and error of q_1 with NNs-based FT control.

This article has been accepted for inclusion in a future issue of this journal. Content is final as presented, with the exception of pagination.

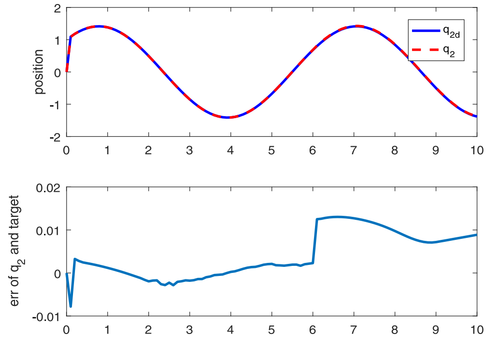


Fig. 6. Tracking performance and error of q_2 with NNs-based FT control.

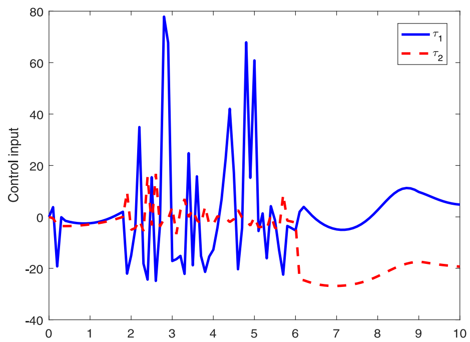


Fig. 7. NNs-based FT control input.

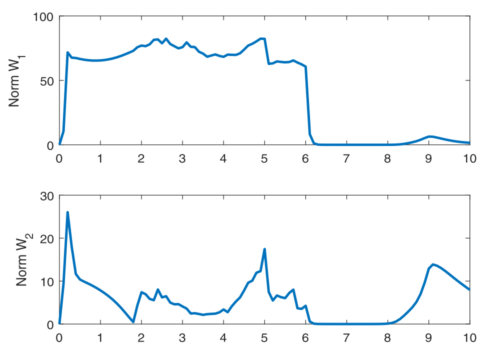


Fig. 8. Estimation weights with NNs-based FT control.

The tracking performances are shown in Figs. 5 and 6. Obviously, the error fluctuations are similar to that of the model-based FT control. However, we can see that the error fluctuations are smoother, and the NNs can reduce the chattering of sliding mode. Then, Fig. 7 shows the value entered by the controller. And Fig. 8 shows the NNs approximation weights. From Fig. 8, we can notice that the weights of the NNs converge.

We define T_a (from actuator failure occurred to tracking errors convergence to zero first) to show the recovery time, as shown in Fig. 9. And we compare the recovery time under different values of control gain η. Then, the results are given in Table III when the actuator failures occur at $t = 6s$. From the comparison, it can be seen that T_a decreases when the values of η increases. However, due to smaller tracking errors and control inputs, we still choose $\eta = 18$ as the gain value of the NNs-based FT controller.

TABLE III
VARIATION OF T_a WITH CONTROL GAIN η

η	18	20	22	24	26	28	30	32
T_a	0.315	0.298	0.284	0.244	0.251	0.233	0.192	0.175

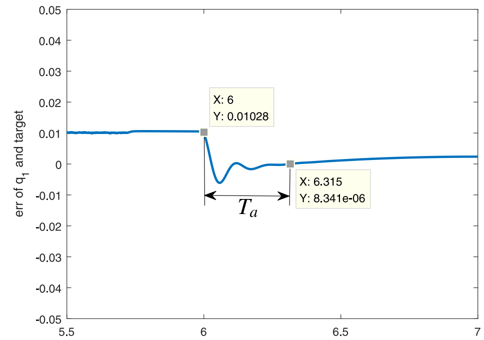


Fig. 9. Illustration of the concept of T_a (when $\eta = 18$).

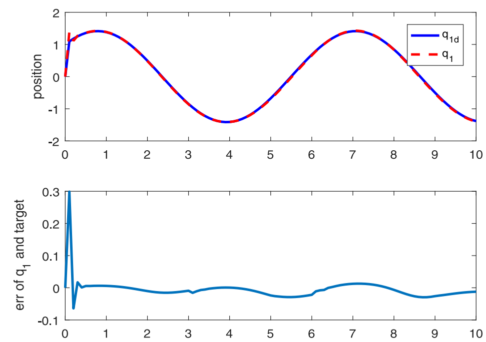


Fig. 10. Tracking performance and error of q_1 with PD control.

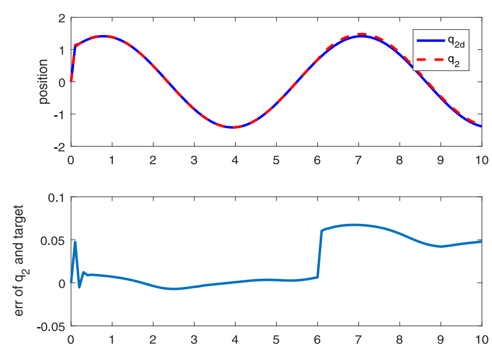


Fig. 11. Tracking performance and error of q_2 with PD control.

C. PD Control

For the PD control, the controller is devised as $\tau = -K_p(x_1 - x_d) - K_d(x_2 - \dot{x}_d)$. The parameters K_p, K_d are chosen as $K_p = \text{diag}[400, 400]$, $K_d = \text{diag}[10, 10]$. The tracking errors of q_1 and q_2 and control input are given to show the effectiveness of the PD control. Figs. 10 and 11 show the tracking performances and tracking errors. Fig. 12 shows the control input.

Compared with Figs. 5 and 6, we can see that the NNs-based FT control has a smaller overshoot than PD control. Besides, we can find that the load of the NNs-based FT control is smaller by comparing Figs. 12 with 7.

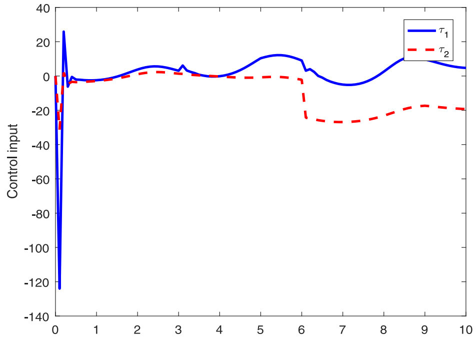


Fig. 12. PD control input.

TABLE IV
COMPARISON OF E_q AND $RMSE_q$ OF THE THREE CONTROL SCHEMES

Controller	E_{q1}	E_{q2}	$RMSE_{q1}$	$RMSE_{q2}$
Model-based	0.0055	0.0108	0.0019	0.0048
NNs-based	0.0487	0.0126	0.0086	0.0069
PD	0.2974	0.0672	0.0339	0.0362

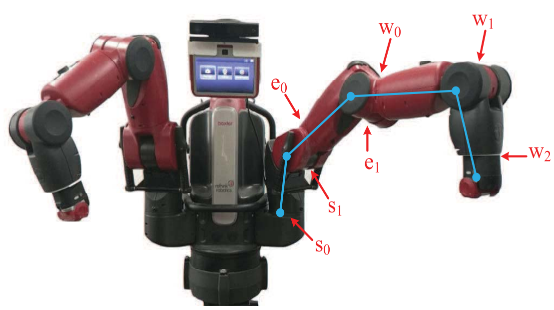


Fig. 13. Baxter robot with seven joints.

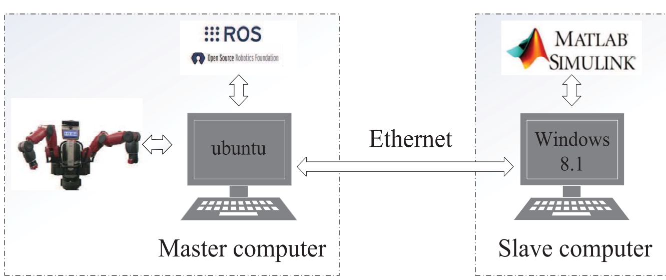


Fig. 14. Experimental schematic.

Then, we compare the maximum position tracking error (E_q) and root mean squared error ($RMSE_q$) of the three control methods. The comparison is shown in Table IV. E_q and $RMSE_q$ are defined as follows:

$$E_q = \max(\mathrm{abs}(q_i - q_d)), i = 1, 2, \ldots, n$$

$$RMSE_q = \frac{1}{10}\sqrt{\sum_{i=1}^{n}(q_i - q_d)^2}, i = 1, 2, \ldots, n. \quad (66)$$

From the comparison, the E_q and $RMSE_q$ of the NNs-based FT control method are smaller than that of PD control method.

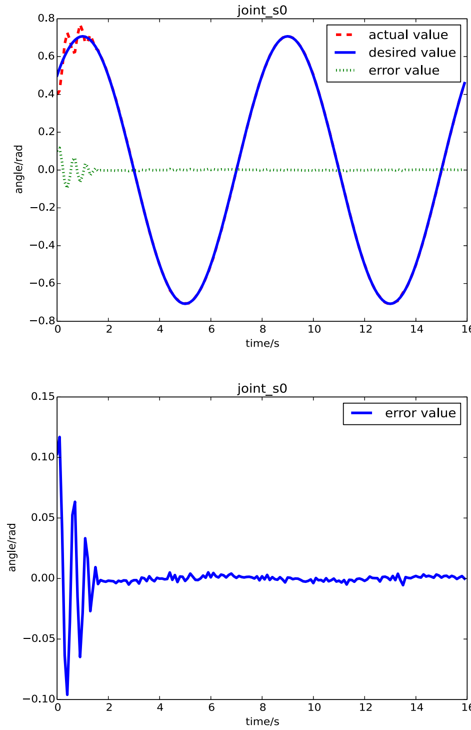


Fig. 15. Tracking performance and error of s_0 and a larger version of the error s_0.

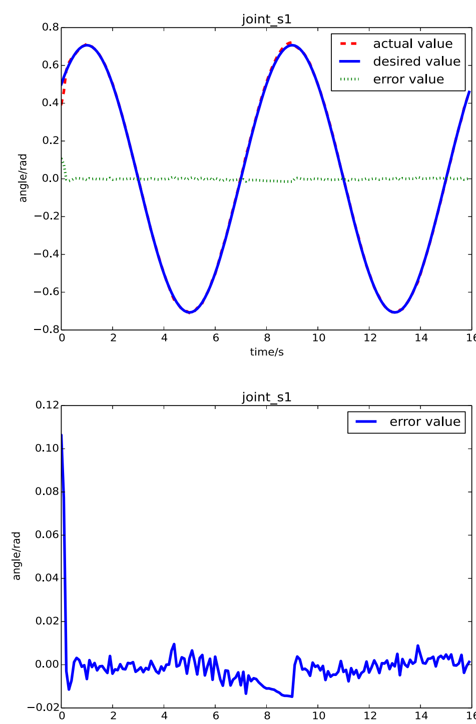


Fig. 16. Tracking performance and error of s_1 and a larger version of the error s_1.

V. EXPERIMENT

In the section, for the sake of further verifying the performance of the proposed control strategy, we test our controller on the Baxter robot with seven joints, which is shown in Fig. 13. To speed up the computing speed, we use

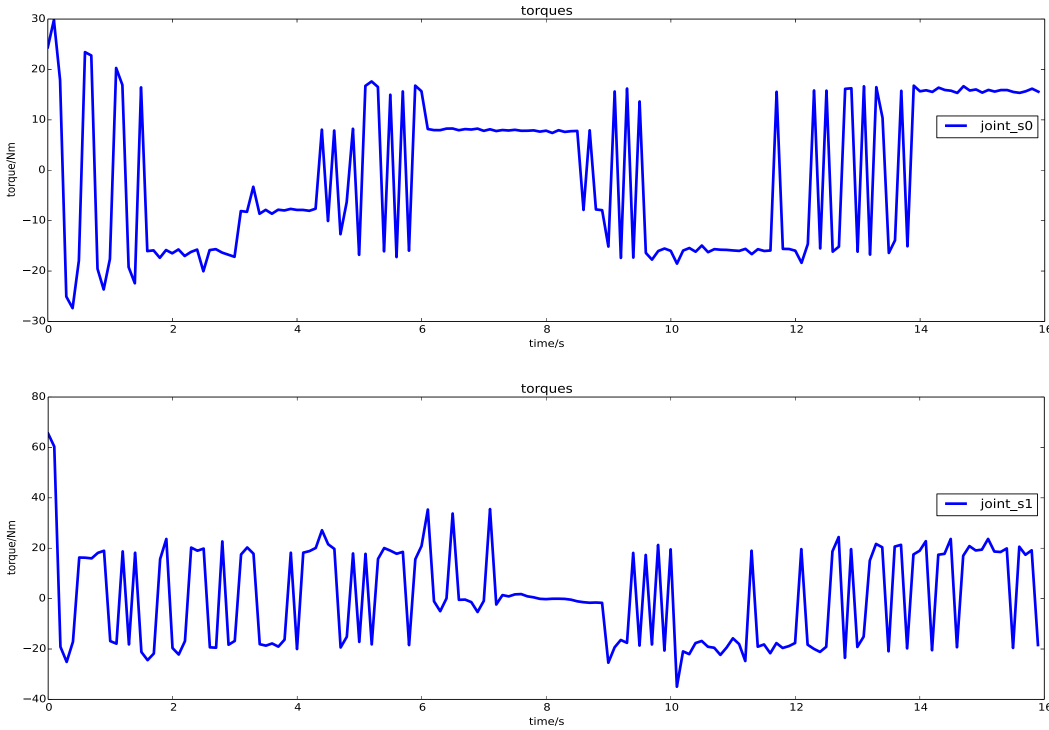


Fig. 17. Actual torques.

two computers to work together. The slave computer with Windows operating system is used to iterate the output of the NNs. The master computer with Ubuntu operating system is applied for trajectory planning and giving the control torque to Baxter robot, and sends the input signals of the NNs to the slave computer. Then, two computers communicate with each other through the Ethernet. The network diagram is shown in Fig. 14. The transmitting and receiving frequency of the master computer is set as 200 Hz. The frequency of total closed-loop is set as 200 Hz.

In this experiment, the desired trajectory is set as $x_d = q_d = 0.5[\sin(t) + \cos(t), \sin(t) + \cos(t)]^T$ for s_0 and s_1 joints. The experiment results are shown in Figs. 15–17. From Figs. 15 and 16, we can find that the trajectory error can converge to a small neighborhood around zero in a short time. From Fig. 17, we can state that the control torques are basically maintained within 40 Nm. Similar to the simulation study, the NNs-based FT controller can track the desired trajectory well.

VI. Conclusion

In this article, NNs-based FT controller has been applied to deal with robotic operating system with external disturbance, parameter uncertainties and actuator failures. In the control design, we have developed the backstepping method to deduce the controller, and the reasoning logic is more clear. The generalized errors based on the NFTSM have increased the robustness of the system, accelerated the convergence speed and shortened the convergence time. For external disturbance, adaptive disturbance observer has been designed in this article. Then, the Gaussian RBFNNs have been used to compensate for external disturbances and actuator failures. Through the simulation in Section IV and the experiment in Section V, we could state that the designed controller could deal with the actuator failures effectively. Since the abrupt fault causes the state jump, we will introduce constraints to avoid the large-scale oscillation of the manipulator caused by the state jump in the future work.

References

[1] W. He, T. Meng, X. He, and S. S. Ge, “Unified iterative learning control for flexible structures with input constraints,” *Automatica*, vol. 96, pp. 326–336, Oct. 2018.

[2] C. L. P. Chen and Z. Liu, “Broad learning system: An effective and efficient incremental learning system without the need for deep architecture,” *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 1, pp. 10–24, Jan. 2018.

[3] G. Wu, J. Sun, and J. Chen, “Optimal linear quadratic regulator of switched systems,” *IEEE Trans. Autom. Control*, vol. 64, no. 7, pp. 2898–2904, Jul. 2019. doi: 10.1109/TAC.2018.2872204.

[4] H. Liang, Z. Zhang, and C. K. Ahn, “Event-triggered fault detection and isolation of discrete-time systems based on geometric technique,” *IEEE Trans. Circuits Syst. II, Exp. Briefs*, to be published. doi: 10.1109/TCSII.2019.2907706.

[5] C. L. P. Chen, C.-E. Ren, and T. Du, “Fuzzy observed-based adaptive consensus tracking control for second-order multiagent systems with heterogeneous nonlinear dynamics,” *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 4, pp. 906–915, Aug. 2016.

[6] Z.-G. Wu, P. Shi, H. Su, and J. Chu, “Asynchronous l_2–l_∞ filtering for discrete-time stochastic Markov jump systems with randomly occurred sensor nonlinearities,” *Automatica*, vol. 50, no. 1, pp. 180–186, 2014.

[7] M. Van, S. S. Ge, and H. Ren, “Finite time fault tolerant control for robot manipulators using time delay estimation and continuous nonsingular fast terminal sliding mode control,” *IEEE Trans. Cybern.*, vol. 47, no. 7, pp. 1681–1693, Jul. 2017.

[8] Q. Shen, B. Jiang, and V. Cocquempot, “Adaptive fuzzy observer-based active fault-tolerant dynamic surface control for a class of nonlinear systems with actuator faults,” *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 2, pp. 338–349, Apr. 2014.

[9] C. L. P. Chen, G.-X. Wen, Y.-J. Liu, and Z. Liu, “Observer-based adaptive backstepping consensus tracking control for high-order nonlinear semi-strict-feedback multiagent systems,” *IEEE Trans. Cybern.*, vol. 46, no. 7, pp. 1591–1601, Jul. 2016.

[10] B. Zhao, D. Liu, and Y. Li, "Observer based adaptive dynamic programming for fault tolerant control of a class of nonlinear systems," *Inf. Sci.*, vol. 384, pp. 21–33, Apr. 2017.

[11] Y. S. Hagh, R. M. Asl, and V. Cocquempot, "A hybrid robust fault tolerant control based on adaptive joint unscented Kalman filter," *ISA Trans.*, vol. 66, pp. 262–274, Jan. 2017.

[12] H. Li, Y. Gao, P. Shi, and H.-K. Lam, "Observer-based fault detection for nonlinear systems with sensor fault and limited communication capacity," *IEEE Trans. Autom. Control*, vol. 61, no. 9, pp. 2745–2751, Sep. 2016.

[13] W. He, X. He, M. Zou, and H. Li, "PDE model-based boundary control design for a flexible robotic manipulator with input backlash," *IEEE Trans. Control Syst. Technol.*, vol. 27, no. 2, pp. 790–797, Mar. 2019.

[14] C. Yang, C. Zeng, P. Liang, Z. Li, R. Li, and C. Su, "Interface design of a physical human–robot interaction system for human impedance adaptive skill transfer," *IEEE Trans. Autom. Sci. Eng.*, vol. 15, no. 1, pp. 329–340, Jan. 2018.

[15] H. Liang, Y. Zhang, T. Huang, and H. Ma, "Prescribed performance cooperative control for multiagent systems with input quantization," *IEEE Trans. Cybern.*, to be published. doi: 10.1109/TCYB.2019.2893645.

[16] W. He and S. S. Ge, "Cooperative control of a nonuniform gantry crane with constrained tension," *Automatica*, vol. 66, pp. 146–154, Apr. 2016.

[17] H. Liang, Y. Zhou, H. Ma, and Q. Zhou, "Adaptive distributed observer approach for cooperative containment control of nonidentical networks," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 49, no. 2, pp. 299–307, Feb. 2019.

[18] W. He and Y. Dong, "Adaptive fuzzy neural network control for a constrained robot using impedance learning," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 4, pp. 1174–1186, Apr. 2018.

[19] Z. Li, B. Huang, A. Ajoudani, C. Yang, C. Su, and A. Bicchi, "Asymmetric bimanual control of dual-arm exoskeletons for human-cooperative manipulations," *IEEE Trans. Robot.*, vol. 34, no. 1, pp. 264–271, Feb. 2018.

[20] Y. Dong and B. Ren, "UDE-based variable impedance control of uncertain robot systems," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published. doi: 10.1109/TSMC.2017.2767566.

[21] L. Kong, W. He, C. Yang, Z. Li, and C. Sun, "Adaptive fuzzy control for coordinated multiple robots with constraint using impedance learning," *IEEE Trans. Cybern.*, vol. 49, no. 8, pp. 3052–3063, Aug. 2019.

[22] H. Li, L. Bai, Q. Zhou, R. Lu, and L. Wang, "Adaptive fuzzy control of stochastic nonstrict-feedback nonlinear systems with input saturation," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 8, pp. 2185–2197, Aug. 2017.

[23] H.-N. Wu, H.-D. Wang, and L. Guo, "Disturbance rejection fuzzy control for nonlinear parabolic PDE systems via multiple observers," *IEEE Trans. Fuzzy Syst.*, vol. 24, no. 6, pp. 1334–1348, Dec. 2016.

[24] H.-N. Wu and H.-Y. Zhu, "Guaranteed cost fuzzy state observer design for semilinear parabolic PDE systems under pointwise measurements," *Automatica*, vol. 85, pp. 53–60, Nov. 2017.

[25] Q. Zhou, H. Li, L. Wang, and R. Lu, "Prescribed performance observer-based adaptive fuzzy control for nonstrict-feedback stochastic nonlinear systems," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 48, no. 10, pp. 1747–1758, Oct. 2018.

[26] Q. Zhou, L. Wang, C. Wu, H. Li, and H. Du, "Adaptive fuzzy control for nonstrict-feedback systems with input saturation and output constraint," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 1, pp. 1–12, Jan. 2017.

[27] W. He, T. Meng, X. He, and C. Sun, "Iterative learning control for a flapping wing micro aerial vehicle under distributed disturbances," *IEEE Trans. Cybern.*, vol. 49, no. 4, pp. 1524–1535, Apr. 2019.

[28] Z. Li, B. Huang, Z. Ye, M. Deng, and C. Yang, "Physical human–robot interaction of a robotic exoskeleton by admittance control," *IEEE Trans. Ind. Electron.*, vol. 65, no. 12, pp. 9614–9624, Dec. 2018.

[29] Z. Li, W. Yuan, Y. Chen, F. Ke, X. Chu, and C. L. P. Chen, "Neural-dynamic optimization-based model predictive control for tracking and formation of nonholonomic multirobot systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 12, pp. 6113–6122, Dec. 2018.

[30] C. L. P. Chen, T. Zhang, L. Chen, and S. C. Tam, "I-Ching divination evolutionary algorithm and its convergence analysis," *IEEE Trans. Cybern.*, vol. 47, no. 1, pp. 2–13, Jan. 2017.

[31] T. Zhang, C. L. P. Chen, L. Chen, X. Xu, and B. Hu, "Design of highly nonlinear substitution boxes based on I-Ching operators," *IEEE Trans. Cybern.*, vol. 48, no. 12, pp. 3349–3358, Dec. 2018.

[32] C. Sun and Y. Xia, "An analysis of a neural dynamical approach to solving optimization problems," *IEEE Trans. Autom. Control*, vol. 54, no. 8, pp. 1972–1977, Aug. 2009.

[33] K. Peng, K. Zhang, J. Dong, and B. You, "Quality-relevant fault detection and diagnosis for hot strip mill process with multi-specification and multi-batch measurements," *J. Frankl. Inst.*, vol. 352, no. 3, pp. 987–1006, 2015.

[34] K. Peng, K. Zhang, B. You, J. Dong, and Z. Wang, "A quality-based nonlinear fault diagnosis framework focusing on industrial multimode batch processes," *IEEE Trans. Ind. Electron.*, vol. 63, no. 4, pp. 2615–2624, Apr. 2016.

[35] J.-X. Zhang and G.-H. Yang, "Robust adaptive fault-tolerant control for a class of unknown nonlinear systems," *IEEE Trans. Ind. Electron.*, vol. 64, no. 1, pp. 585–594, Jan. 2017.

[36] H. Li, S. Zhao, W. He, and R. Lu, "Adaptive finite-time tracking control of full state constrained nonlinear systems with dead-zone," *Automatica*, vol. 100, pp. 99–107, Feb. 2019.

[37] H. Li, L. Wang, H. Du, and A. Boulkroune, "Adaptive fuzzy backstepping tracking control for strict-feedback systems with input delay," *IEEE Trans. Fuzzy Syst.*, vol. 25, no. 3, pp. 642–652, Jun. 2017.

[38] C. Mu, Z. Ni, C. Sun, and H. He, "Data-driven tracking control with adaptive dynamic programming for a class of continuous-time nonlinear systems," *IEEE Trans. Cybern.*, vol. 47, no. 6, pp. 1460–1470, Jun. 2017.

[39] S.-L. Dai, S. He, H. Lin, and C. Wang, "Platoon formation control with prescribed performance guarantees for USVs," *IEEE Trans. Ind. Electron.*, vol. 65, no. 5, pp. 4237–4246, May 2018.

[40] Y.-J. Liu and S. Tong, "Barrier Lyapunov functions for Nussbaum gain adaptive control of full state constrained nonlinear systems," *Automatica*, vol. 76, pp. 143–152, Feb. 2017.

[41] Q. Zhou, P. Shi, Y. Tian, and M. Wang, "Approximation-based adaptive tracking control for MIMO nonlinear systems with input saturation," *IEEE Trans. Cybern.*, vol. 45, no. 10, pp. 2119–2128, Oct. 2015.

[42] G. Xie, L. Sun, T. Wen, X. Hei, and F. Qian, "Adaptive transition probability matrix-based parallel IMM algorithm," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published. doi: 10.1109/TSMC.2019.2922305.

[43] C. Sun, W. He, and J. Hong, "Neural network control of a flexible robotic manipulator using the lumped spring-mass model," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 8, pp. 1863–1874, Aug. 2017.

[44] Z.-G. Wu, J. Lam, H. Su, and J. Chu, "Stability and dissipativity analysis of static neural networks with time delay," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 2, pp. 199–210, Feb. 2012.

[45] Z.-G. Wu, P. Shi, H. Su, and J. Chu, "Stochastic synchronization of Markovian jump neural networks with time-varying delay using sampled data," *IEEE Trans. Cybern.*, vol. 43, no. 6, pp. 1796–1806, Dec. 2013.

[46] H. Gao, Y. Song, and C. Wen, "Backstepping design of adaptive neural fault-tolerant control for MIMO nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 28, no. 11, pp. 2605–2613, Nov. 2017.

[47] S.-L. Dai, M. Wang, and C. Wang, "Neural learning control of marine surface vessels with guaranteed transient tracking performance," *IEEE Trans. Ind. Electron.*, vol. 63, no. 3, pp. 1717–1727, Mar. 2016.

[48] W. He, Y. Dong, and C. Sun, "Adaptive neural impedance control of a robotic manipulator with input saturation," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 3, pp. 334–344, Mar. 2016.

[49] Y.-J. Liu, S. Tong, C. L. P. Chen, and D.-J. Li, "Neural controller design-based adaptive control for nonlinear MIMO systems with unknown hysteresis inputs," *IEEE Trans. Cybern.*, vol. 46, no. 1, pp. 9–19, Jan. 2016.

[50] S.-L. Dai, S. He, M. Wang, and C. Yuan, "Adaptive neural control of underactuated surface vessels with prescribed performance guarantees," *IEEE Trans. Neural Netw. Learn. Syst.*, to be published. doi: 10.1109/TNNLS.2018.2876685.

[51] L. Kong, W. He, Y. Dong, L. Cheng, C. Yang, and Z. Li, "Asymmetric bounded neural control for an uncertain robot by state feedback and output feedback," *IEEE Trans. Syst., Man, Cybern., Syst.*, to be published. doi: 10.1109/TSMC.2019.2901277.

[52] M. Chen, S.-Y. Shao, and B. Jiang, "Adaptive neural control of uncertain nonlinear systems using disturbance observer," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3110–3123, Oct. 2017.

[53] W. He, Z. Yan, C. Sun, and Y. Chen, "Adaptive neural network control of a flapping wing micro aerial vehicle with disturbance observer," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3452–3465, Oct. 2017.

[54] C. Yang, X. Wang, L. Cheng, and H. Ma, "Neural-learning-based telerobot control with guaranteed performance," *IEEE Trans. Cybern.*, vol. 47, no. 10, pp. 3148–3159, Oct. 2017.

[55] M. Chen and G. Tao, "Adaptive fault-tolerant control of uncertain nonlinear large-scale systems with unknown dead zone," *IEEE Trans. Cybern.*, vol. 46, no. 8, pp. 1851–1862, Aug. 2016.

[56] X. Jin, "Adaptive fault tolerant control for a class of input and state constrained MIMO nonlinear systems," *Int. J. Robust Nonlin. Control*, vol. 26, no. 2, pp. 286–302, 2015.

[57] W. Wang and C. Wen, "Adaptive actuator failure compensation control of uncertain nonlinear systems with guaranteed transient performance," *Automatica*, vol. 46, no. 12, pp. 2082–2091, 2010.

[58] M. Chen, P. Shi, and C.-C. Lim, "Adaptive neural fault-tolerant control of a 3-DOF model helicopter system," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 6, pp. 260–270, Feb. 2016.

[59] S. Tong, B. Huo, and Y. Li, "Observer-based adaptive decentralized fuzzy fault-tolerant control of nonlinear large-scale systems with actuator failures," *IEEE Trans. Fuzzy Syst.*, vol. 22, no. 1, pp. 1–15, Feb. 2014.

[60] B. Xiao, S. Yin, and H. Gao, "Reconfigurable tolerant control of uncertain mechanical systems with actuator faults: A sliding mode observer-based approach," *IEEE Trans. Control Syst. Technol.*, vol. 26, no. 4, pp. 1249–1258, Jul. 2018.

[61] H. Li, P. Shi, and D. Yao, "Adaptive sliding-mode control of Markov jump nonlinear systems with actuator faults," *IEEE Trans. Autom. Control*, vol. 62, no. 4, pp. 1933–1939, Apr. 2017.

[62] Y. Gao, L. Wu, P. Shi, and H. Li, "Sliding mode fault-tolerant control of uncertain system: A delta operator approach," *Int. J. Robust Nonlin. Control*, vol. 27, no. 18, pp. 4173–4187, Feb. 2017.

[63] S. Mobayen, D. Baleanu, and F. Tchier, "Second-order fast terminal sliding mode control design based on LMI for a class of non-linear uncertain systems and its application to chaotic systems," *J. Vib. Control*, vol. 23, no. 18, pp. 2912–2925, 2017.

[64] R. M. Asl, Y. S. Hagh, and R. Palm, "Robust control by adaptive non-singular terminal sliding mode," *Eng. Appl. Artif. Intell.*, vol. 59, pp. 205–217, Mar. 2017.

[65] Q. Meng, T. Zhang, X. Gao, and J.-Y. Song, "Adaptive sliding mode fault-tolerant control of the uncertain stewart platform based on offline multibody dynamics," *IEEE/ASME Trans. Mechatronics*, vol. 19, no. 3, pp. 882–894, Jun. 2014.

[66] M. Van and H.-J. Kang, "Robust fault-tolerant control for uncertain robot manipulators based on adaptive quasi-continuous high-order sliding mode and neural network," *Proc. Inst. Mech. Eng. C J. Mech. Eng. Sci.*, vol. 229, no. 8, pp. 1425–1446, 2015.

[67] M. Van, P. Franciosa, and D. Ceglarek, "Fault diagnosis and fault-tolerant control of uncertain robot manipulators using high-order sliding mode," *Math. Problems Eng.*, vol. 2016, p. 14, Jul. 2016.

[68] M. Van, "An enhanced robust fault tolerant control based on an adaptive fuzzy PID-nonsingular fast terminal sliding mode control for uncertain nonlinear systems," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 3, pp. 1362–1371, Jun. 2018.

[69] R. M. Sanner and J.-J. E. Slotine, "Gaussian networks for direct adaptive control," *IEEE Trans. Neural Netw.*, vol. 3, no. 6, pp. 837–863, Nov. 1992.

[70] S.-L. Dai, C. Wang, and M. Wang, "Dynamic learning from adaptive neural network control of a class of nonaffine nonlinear systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 1, pp. 111–123, Jan. 2014.

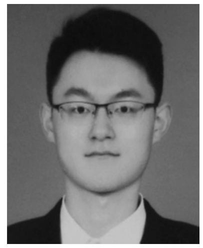
[71] W. He, Z. Li, Y. Dong, and T. Zhao, "Design and adaptive control for an upper limb robotic exoskeleton in presence of input saturation," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 1, pp. 97–108, Jan. 2019.

[72] L. Yang and J. Yang, "Nonsingular fast terminal sliding-mode control for nonlinear dynamical systems," *Int. J. Robust Nonlin. Control*, vol. 21, no. 16, pp. 1865–1879, 2011.

[73] S. Zhang, Y. Dong, Y. Ouyang, Z. Yin, and K. Peng, "Adaptive neural control for robotic manipulators with output constraints and uncertainties," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 29, no. 11, pp. 5554–5564, Nov. 2018.

Shuang Zhang (S'11–M'15) received the M.Eng. degree in automatic control from the School of Automation Science and Engineering, South China University of Technology, Guangzhou, China, in 2009, and the Ph.D. degree from the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, in 2012.

She is currently an Associate Professor with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing, China. Her current research interests include robotics, adaptive controls, and vibration controls.

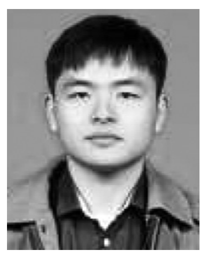


Pengxin Yang received the B.Eng. degree in automation from the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing, China, in 2018, where he is currently pursuing the M.Eng. degree in control engineering with the School of Automation and Electrical Engineering.

His current research interests include robotics, neural network control, and fault tolerant control.

Linghuan Kong received the B.Eng. degree in automation from the College of Engineering, Qufu Normal University, Rizhao, China, in 2016, the M.Eng. degree in control engineering from the School of Automation Engineering, University of Electronic Science and Technology of China, Chengdu, China, in 2019. He is currently pursuing the Ph.D. degree in control science and engineering with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing, China.

His current research interests include robotics, neural network control, and adaptive control.



Wenshi Chen received the B.Eng. degree in intelligent science and technology from the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing, China, in 2018, where he is currently pursuing the M.Eng. degree in control engineering.

His current research interests include robotics, neural network control, and dynamic movement primitives.

Qiang Fu received the B.S. degree in thermal energy and power engineering from Beijing Jiaotong University, Beijing, China, in 2009, and the Ph.D. degree in control science and engineering from Beihang University (formerly Beijing University of Aeronautics and Astronautics), Beijing, in 2016.

He is currently a Lecturer with the School of Automation and Electrical Engineering, University of Science and Technology Beijing, Beijing. His current research interests include vision-based navigation and 3-D vision.

Kaixiang Peng (M'15) received the B.E. degree in automation and the M.E. and Ph.D. degrees from the Research Institute of Automatic Control, University of Science and Technology, Beijing, China, in 1995, 2002, and 2007, respectively.

He is currently a Professor with the School of Automation and Electrical Engineering, University of Science and Technology. His current research interests include fault diagnosis, prognosis, and maintenance of complex industrial processes, modeling and control for complex industrial processes, and control system design for the rolling process.