Composite-Learning-Based Adaptive Neural Control for Dual-Arm Robots With Relative Motion

Yiming Jiang, *Member, IEEE*, Yaonan Wang, Zhiqiang Miao, *Member, IEEE*, Jing Na, *Member, IEEE*, Zhijia Zhao, *Member, IEEE*, and Chenguang Yang, *Senior Member, IEEE*

***Abstract*— This article presents an adaptive control method for dual-arm robot systems to perform bimanual tasks under modeling uncertainties. Different from the traditional symmetric bimanual robot control, we study the dual-arm robot control with relative motions between robotic arms and a grasped object. The robot system is first divided into two subsystems: a settled manipulator system and a tool-used manipulator system. Then, a command filtered control technique is developed for trajectory tracking and contact force control. In addition, to deal with the inevitable dynamic uncertainties, a radial basis function neural network (RBFNN) is employed for the robot, with a novel composite learning law to update the NN weights. The composite learning is mainly based on an integration of the historic data of NN regression such that information of the estimate error can be utilized to improve the convergence. Moreover, a partial persistent excitation condition is employed to ensure estimation convergence. The stability analysis is performed by using the Lyapunov theorem. Numerical simulation results demonstrate the validity of the proposed control and learning algorithm.**

***Index Terms*— Adaptive robot control, bimanual robot, composite learning, neural network, relative motion.**

I. Introduction

RECENTLY, coordination control of dual-arm robots has received increasing attention due to its superiority compared with traditional single-arm robot systems, including stronger payload capability, larger workspace, and more flexibility. Thus, the dual-arm robots have been involved in many high technology applications, such as intelligent assembly, out-space repairing, and elderly people assistance [1]–[3]. However, controlling the dual-arm robots is challenging due to the increase of complexity in motion control and path planning. Therefore, advanced control technologies have been extensively studied for dual-arm robots in past decades [4]–[11].

An adaptive decentralized control scheme was proposed to address the object handling problem of a cooperative robot, where an implicit force control scheme was employed to simultaneously regulate the force and position [9]. In [10], a decentralized control structure for multiple mobile manipulators was developed, where the internal forces were constrained by employing an augmented object model for the multiple systems with a virtual linkage. In [11], the loading problem for multiple manipulators was addressed by analyzing the grasp space of the robot. Note that the abovementioned controllers were developed under the assumption that the object is firmly held by the robotic arms such that no relative motion occurred between the arms and the objects. However, in practical applications, such as polishing, grinding, and welding, the robot end-effectors need to operate along the object's surface, where sliding movements usually happened between the robotic arm and the object [12]–[14].

In this respect, the coordination control of dual-arm robots with relative motion deserves further investigation. The relative motion is also known as the asymmetric bimanual task. In [15], a relative impedance controller was developed by using a relative Jacobian method such that the dual-arm system can be treated as a single-arm robotic system. In [16], a brain-actuated control architecture was proposed for dual-arm robots to perform the asymmetric bimanual task, where electroencephalogram signals and visual stimulation were employed to send control command through a brain–machine interface. In these works, however, the controllers were designed under the assumption that the robot dynamics are fully available, while the stability analysis of the contact force between the robotic arm and the object was not given.

The dynamic model of the robot system is of great importance in the controller design [17]–[22], but it is often unavailable in practice. For example, in carrying tasks, the dynamics of the grasped object is hard to obtain in advance. Without a precise dynamics model, the model-based control method became invalid and may cause degeneration of the control performance. Hence, advanced control strategies have been presented to compensate for the model uncertainties. Neural network (NN) is well known by its advantages in alleviating

Manuscript received November 13, 2019; revised June 13, 2020; accepted November 2, 2020. This work was supported in part by the National Natural Science Foundation of China under Grant 62003136, Grant 61733004, Grant 61803109, Grant 61903135, Grant 61922037, and Grant 62073131; in part by the Science Technology Project of Hunan Province under Grant 2017XK2102, Grant 2018GK2022, and Grant 2018JJ3079; in part by the Special Funding Support for the Construction of Innovative Provinces in Hunan Province under Grant 2019GK1010; in part by the Natural Science Foundation of Hunan Province under Grant 2020JJ5090; and in part by the Fellowship of the China Postdoctoral Science Foundation under Grant 2020M672486 and Grant 2020M682554. *(Corresponding author: Yaonan Wang.)*

Yiming Jiang, Yaonan Wang, and Zhiqiang Miao are with the National Engineering Laboratory for Robot Visual Perception and Control, Hunan University, Changsha 410082, China (e-mail: ymjiang@hnu.edu.cn; yaonan@hnu.edu.cn; miaozhiqiang@hnu.edu.cn).

Jing Na is with the Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming 650093, China (e-mail: najing25@163.com).

Zhijia Zhao is with the School of Mechanical and Electrical Engineering, Guangzhou University, Guangzhou 510006, China (e-mail: zhjzhaoscut@163.com).

Chenguang Yang is with the Bristol Robotics Laboratory, University of the West of England, Bristol BS16 1QY, U.K. (e-mail: cyang@ieee.org).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TNNLS.2020.3037795.

Digital Object Identifier 10.1109/TNNLS.2020.3037795

modeling difficulties of nonlinear systems due to the powerful approximation ability [23]. Thus, NN control synthesizes have been widely implemented in developing controllers for nonlinear robotic systems [24]–[32].

A fuzzy neural network control approach was presented for pure-feedback stochastic systems by using a semi-Nussbaum function [33]. In [34], an adaptive NN control strategy was proposed for an uncertain robot to ensure the state not to violate the prescribed constraints. Recently, a sensorless admittance controller was designed to solve the unknown environments' interaction by using the NN technique [35]. Significant works have been done in [36]–[38] to make a complex topic understandable about modeling and control of flapping-wing flying robots to the average reader. While the NN controllers have been successfully developed in the abovementioned work, a major limitation for existing adaptive NN control schemes lies in that only convergence of the tracking errors can be achieved, instead of the convergence of NN weights to their ideal values. Without the convergence of NN weights, the NN compensation can be hardly accomplished, and the system performance may be degraded and, eventually, became unstable. In this respect, developing a novel control scheme with guaranteed NN convergence is of great significance.

In our recent work [39], a filtered operation was presented to control the robotic arm with finite-time convergence under a linear-in-parameter (LIP) robotic dynamic model. Nevertheless, the guaranteed convergence of the NN weights is more difficult. It is well known that the persistent excitation (PE) condition is important to guarantee the estimation convergence [40]. However, in practice, it is very stringent to ensure the PE condition of neural networks due to the sparse characteristics of the NN regressor vector. Recent research of neural networks in [41] presented a partial persistent excitation (PPE) condition instead of the traditional PE condition. It has been proven that, for the radial basis function neural network (RBFNN) defined in a regular lattice, neural nodes could be partially activated for any recurrent NN inputs trajectory remained in this local region [41]. In the subsequent work [42], this idea was employed for the control design of nonlinear strict-feedback systems to guarantee the system stability and accurate NN approximation. However, the NN inputs still need to satisfy the condition of recurrent trajectory, and a small input excitation strength may lead to slow learning speed.

The work in [43] indicates that parameter convergence can be improved if certain information of the estimation error can be integrated into the adaptation. In [44], a novel parameter estimation law was proposed for a robotic system with unknown dynamics by using a sliding mode technique and a finite-time estimator. In [45], the estimation error was integrated into the adaptation scheme of a class of nonlinear systems to achieve the convergence of NN weights. Motivated by the abovementioned idea, in this article, we develop a composite learning controller for the dual-arm robot to perform bimanual relative motion tasks. To the best of our knowledge, few studies have investigated the learning control in the frame of the dual-arm robot systems subject to relative motion and unknown dynamics. Moreover, different from the work in [46], a PPE condition is also introduced in the estimation scheme to achieve a relaxation of the requirement of the PE condition. In comparison to the method in [45], the estimation error of the NN weights is properly expressed and employed to enhance the approximation of the neural network.

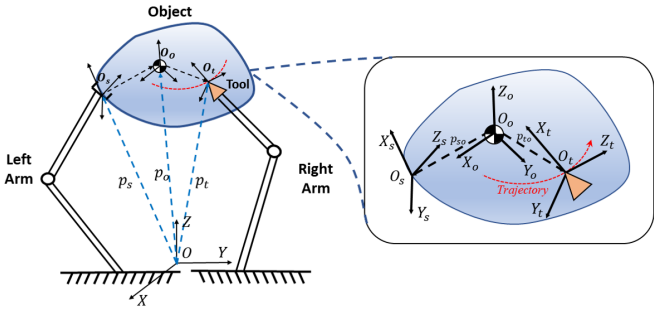

Fig. 1. Overlook of the two-robotic-arm coordination with relative motion.

The objective of this article is to develop a control framework for dual-arm robot tracking control under relative motion. The main contributions of this article can be summarized as follows.

1) A novel neural control framework is developed for dual-arm robot systems to perform asymmetric bimanual tasks with no prior knowledge of the dynamics.
2) A novel composite learning algorithm is designed for NN weights adaptation such that information of the estimate errors could be appropriately integrated into the adaptation law to improve the estimation performance.
3) A partial persistent condition is introduced for the adaptation of NN weights such that the requirement of conventional PE condition can be greatly relaxed.

In the following sections, the system modeling and control design procedures are detailed. Section II discusses the system modeling of the dual-arm robot in addition to some preliminaries. Section III presents the design of the composite learning control algorithm by utilizing a command filtered backstepping technique with stability analysis. Section IV demonstrates the simulation results. A brief conclusion is given in Section V.

II. System Description

A. Kinematics Modeling

The system studied in this article is shown in Fig. 1, where a rigid object is tightly caught by a settled manipulator (SM), while a tool-used manipulator (TM) holds an implement to follow a given trajectory on the surface of the object. As depicted in Fig. 1, the world coordinate is described by $OXYZ$. Besides, three coordinates are attached to the settled manipulator end-effector (SME) $(O_sX_sY_sZ_s)$, the tool-used manipulator end-effector (TME) $(O_tX_tY_tZ_t)$, and the grasped object $(O_oX_oY_oZ_o)$, respectively. The SME carries an object to track the desired trajectory during the manipulation task. The TME follows a trajectory $\rho(p_{to}) = 0$ defined on a manifold of the object surface, and relative motion is, therefore, generated between the object and robotic end-effector. Table I presents some notations used in this article.

To facilitate the modeling procedure, the following assumptions are adopted.

TABLE I
NOMENCLATURE OF THE COORDINATION SYSTEM

O_o	Centre of mass (COM) of object i
O_s	Contact point where the TME holds the object
O_t	Contact point between the tool manipulator and object
$p_o = [r_o, \theta_o]$	Configuration of $O_oX_oY_oZ_o$ in world frame with r_o being the position and θ_o being the orientation of $O_oX_oY_oZ_o$
$p_s = [r_s, \theta_s]$	Configuration of $O_sX_sY_sZ_s$ in the world frame with r_s being the position and θ_s is the orientation
$p_t = [r_t, \theta_t]$	Configuration of $O_tX_tY_tZ_t$ in world frame with r_t being the position and θ_t being the orientation
$p_{so} = [r_{so}^T, \theta_{so}^T]^T$	Vector from the origin of the $O_oX_oY_oZ_o$ to $O_sX_sY_sZ_s$, r_{so} and θ_{so} are the translation vector and rotation vector
$p_{to} = [r_{to}^T, \theta_{to}^T]^T$	Vector from the origin of $O_oX_oY_oZ_o$ to $O_tX_tY_tZ_t$, r_{to} and θ_{to} are the translation vector and rotation vector
q_s	Joint variable of the settled manipulator
q_t	Joint variable of the tool manipulator
$\rho(p_{to}) = 0$	Manifold the TME followed

Assumption 1: The object is firmly caught by the end-effector of the SM, and no sliding movement has occurred between SME and the object.

Assumption 2: The TME is always in contact with the object surface.

Assumption 3: The inverse kinematics of the dual-arm robot are fully available, and singularity points are properly avoided.

In terms of the notations in Table I, the kinematics of the dual-arm coordination system can be described as follows [47]:

$$r_t = r_o + R_o(\theta_o) r_{to} \tag{1}$$
$$r_s = r_o + R_o(\theta_o) r_{so} \tag{2}$$
$$R_t = R_o(\theta_o) R_{to}(\theta_{to}) \tag{3}$$
$$R_s = R_o(\theta_o) \tag{4}$$

where θ_o is the rotation angle of the object's CoM, θ_{to} is the rotation from the object's CoM to the TME, and $R_o(\theta_o) \in \mathbb{R}^{l_r \times l_r}$ and $R_{to}(\theta_{to}) \in \mathbb{R}^{l_r \times l_r}$ are the rotation matrices of θ_o and θ_{to}, respectively, with l_r being the dimension of rotation. Then, R_t and R_s represent the rotation matrices of the SME and TME, respectively.

Taking the derivative of (1) with respect to time yields

$$\dot{r}_t = \dot{r}_o + R_o(\theta_o)\dot{r}_{to} - E(R_o(\theta_o) r_{to})\phi_o \tag{5}$$
$$\dot{r}_s = \dot{r}_o - E(R_o(\theta_o) r_{so})\phi_o \tag{6}$$
$$\phi_t = \phi_o + R_o(\theta_o)\phi_{to} \tag{7}$$
$$\phi_s = \phi_o \tag{8}$$

and

$$E(a) = \begin{bmatrix} 0 & -a_3 & a_2 \\ a_3 & 0 & -a_1 \\ -a_2 & a_1 & 0 \end{bmatrix} \tag{9}$$

for any given vector $a = [a_1, a_2, a_3]^T$, where ϕ_o, ϕ_{to}, and ϕ_{so} are the vectors of angular velocities. Note that, in the above derivation, due to the factor that the workpiece is tightly caught by the TME, $\phi_{so} = 0$ and $\dot{\phi}_{so} = 0$ are omitted in (5)–(8).

Equations (5)–(8) can be rewritten in a compact form as

$$\upsilon_t = D_t \upsilon_o + R_\upsilon \upsilon_{to} \tag{10}$$
$$\upsilon_s = D_s \upsilon_o \tag{11}$$

where $\upsilon_t = [\dot{r}_t^T \ \phi_t^T]^T$, $\upsilon_s = [\dot{r}_s^T \ \phi_s^T]^T$, $\upsilon_o = [\dot{r}_o^T \ \phi_o^T]^T$, $\upsilon_{to} = [\dot{r}_{to}^T \ \phi_{to}^T]^T$, and

$$R_\upsilon = \begin{bmatrix} R_o(\theta_o) & 0 \\ 0 & R_o(\theta_o) \end{bmatrix} \tag{12}$$

$$D_t = \begin{bmatrix} I & -E(R_o(\theta_o) r_{to}) \\ 0 & I \end{bmatrix} \tag{13}$$

$$D_s = \begin{bmatrix} I & -E(R_o(\theta_o) r_{so}) \\ 0 & I \end{bmatrix} \tag{14}$$

with I being an unit matrix with proper dimension. Note that the TME follows a reference trajectory $\rho(p_{to}) = 0$ on the rigid object surface, and the constraint force f_t vertical to object surface can be expressed as follows [48]:

$$f_t = \eta_t \xi \tag{15}$$
$$\eta_t = \frac{R_\upsilon (\partial\rho/\partial p_{to})^T}{||(\partial\rho/\partial p_{to})^T||} \tag{16}$$

where η_t is a vector representing the direction of the constraint force and ξ is a Lagrange multiplier that denotes the force magnitude. Then, through the transformation matrix D_t, the contact force applied on the object can be obtained as

$$f_o = -D_t^T f_t = -D_t^T \eta_t \xi. \tag{17}$$

B. Dynamics Modeling

In order to build the dynamic model of the dual-arm robot system, we first build the dynamic model of the SME and the TME, respectively. Remind in Assumption 1 that the object is rigidly caught by the SME, and no motion is generated between the SME and object. Hence, the object and the SME can be treated as a lump dynamic system, and its dynamics is obtained by using the Lagrange–Euler method as follows:

$$M_s(q_s)\ddot{q}_s + C_s(q_s, \dot{q}_s)\dot{q}_s + G_s(q_s) = \tau_s + J_s^T(q_s) f_o \tag{18}$$

where M_s is the inertia matrix of the arm-object system, C_s is the Coriolis/centrifugal force matrix, G_s is the gravity vector, τ_s is the torque of the joints, and J_s is the Jacobian matrix. On the other hand, the dynamics of TME can be described as follows:

$$M_t(q_t)\ddot{q}_t + C_t(q_t, \dot{q}_t)\dot{q}_t + G_t(q_t) = \tau_t + J_t^T(q_t) f_t \tag{19}$$

where M_t, C_t, and G_t are the matrices of inertia, Coriolis/centrifugal force, and the vector of the gravitational force, respectively, τ_t is the joints torque of TM, f_t is the contact force defined in (15), and J_t is the corresponding Jacobian matrix.

Combining the dynamics of robot systems (17)–(19), we can obtain the robot dynamics of the dual-arm robot system as

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) = \tau + J^T(q)\eta_t \xi \tag{20}$$

where

$$M(q)=\begin{bmatrix} M_s(q_s) & 0 \\ 0 & M_t(q_t)\end{bmatrix}$$
$$C(q,\dot{q})=\begin{bmatrix} C_s(q_s,\dot{q}_s) & 0 \\ 0 & C_t(q_t,\dot{q}_t)\end{bmatrix}$$
$$G(q)=\begin{bmatrix} G_s(q_s) \\ G_t(q_t)\end{bmatrix},\ J(t)=\begin{bmatrix} J_s(q_s) \\ J_t(q_t)\end{bmatrix},$$
$$q=\begin{bmatrix} q_s \\ q_t\end{bmatrix},\quad \tau=\begin{bmatrix} \tau_s \\ \tau_t\end{bmatrix}. \tag{21}$$

Note that the dual-arm robot is constrained by a holonomic kinematic constraints represented by (10) and (11); thus, some degrees of freedom of the robotic system will be lost in the movement. Let us select a set of n independent variables $q_c = [q_{c1}, q_{c2}, \ldots, q_{cn}]^T$ from the joints variable $q = [q_s^T, q_t^T]^T$, and q can be represented by a function of $d(q_c)$ as

$$q = d(q_c). \tag{22}$$

Differentiating (22) with respect to time yields

$$\begin{aligned}\dot{q} &= N(q_c)\dot{q}_c \\ \ddot{q} &= N(q_c)\ddot{q}_c + \dot{N}(q_c)\dot{q}_c\end{aligned} \tag{23}$$

where $N(q_c) = \partial q/\partial q_c$ with N being a full column rank matrix.

Note that $v_{to} = R_v^{-1}(v_t - D_t v_o) = R_v^T(v_t - D_t v_o)$, and $v_t = J_t(q_t)\dot{q}_t$ and $v_o = J_o(q_t)\dot{q}_o$ hold in terms of the kinematics. Also, $v_{to} = R_v^T J_c N_c \dot{q}_c$ can be derived from (22) and (23). Since v_{to} and η_t are orthogonal to each other, we can obtain $\eta_t^T v_{to} = 0$. From the above derivation, we can derive $\eta_t^T J_c N_c \dot{q}_c = 0$. Since q_c is a variable, we obtain that $N^T(q_c)J_c^T(q_c)\eta_t = 0$.

The combination of (20), (22), and (23) yields

$$M_c(q_c)\ddot{q}_c + C_c(q_c,\dot{q}_c)\dot{q}_c + G_c(q_c) = \tau + J_c^T(q_c)\eta_t\xi \tag{24}$$

where $M_c(q_c) = M(q_c)N(q_c)$, $C_c(q_c,\dot{q}_c) = M(q_c)(\dot{N}(q_c) + C(q_c,\dot{q}_c)N(q_c))$, $G_c = G(d(q_c))$, and $J_c(q_c) = J(d(q_c))$.

Several useful properties for the system (24) is given as follows.

Property 1 [47]: The matrix $\mathcal{M}(q_c) = N^T M_c(q_c)$ is a bounded symmetric positive definite matrix.

Property 2 [47]: Define $\mathcal{C}(q_c,\dot{q}_c) = N^T C_c(q_c,\dot{q}_c)$. Then, the matrix $\dot{\mathcal{M}}(q_c) - 2\mathcal{C}(q_c,\dot{q}_c)$ is a skew-symmetric matrix such that

$$v^T(\dot{\mathcal{M}} - 2\mathcal{C})v = 0 \quad \forall v \in \mathbb{R}^n. \tag{25}$$

Property 3 [48]: For the tool-use manipulator, the following relationship is satisfied:

$$N^T(q_c)J_c^T(q_c)\eta_t = 0. \tag{26}$$

It should be emphasized that (24) gives the constraint dynamics of the dual-arm robot, where the last term on the right-hand side represents the contact force applied on the robotic joint. To address the effect of the contact force, let us consider a controller τ_c as follows:

$$\tau_c = \tau_d - J_c(q_c)\eta_t\left(\xi_d + k_\xi\int_0^t e_\xi d\tau\right) \tag{27}$$

where τ_d is a feedback control term to be designed later, and $e_\xi = \xi - \xi_d$. Note that, in (27), τ_d is designed for the trajectory following, while the later term is developed to deal with the contact force between the TME and the grasped object.

Substituting (27) into (24), we have

$$\begin{aligned}&M_c(q_c)\ddot{q}_c + C_c(q_c,\dot{q}_c)\dot{q}_c + G_c(q_c) \\ &\quad= \tau_d - J_c(q_c)\eta_t\left(\xi_d + k_\xi\int_0^t e_\xi d\tau\right) + J_c^T(q_c)\eta_t\xi \\ &\quad= \tau_d - J_c(q_c)\eta_t\left(e_\xi + k_\xi\int_0^t e_\xi d\tau\right).\end{aligned} \tag{28}$$

Premultiplying on both sides of (28) with N^T and considering $N^T(q_c)J_c^T(q_c)\eta_t = 0$ in Property 3, we have

$$\mathcal{M}(q_c)\ddot{q}_c + \mathcal{C}(q_c,\dot{q}_c)\dot{q}_c + \mathcal{G}(q_c) = N^T\tau_d \tag{29}$$

where $\mathcal{M}(q_c)$ with $\mathcal{C}(q_c,\dot{q}_c)$ are defined in Properties 1 and 2, respectively, and $\mathcal{G}(q_c) = N^T G_c(q_c)$.

Remark 1: Equation (29) represents a subspace of the robot dynamics, which is orthogonal to the contact constraints. That is, (29) gives a description of the error dynamics of the robot tracking, while (28) represents the error dynamics of the contact force e_ξ. Now, the dynamic behavior of the dual-arm robot system with relative motion and contact constraint is available for the controller design.

C. Preliminaries

A number of useful definitions and lemmas used in this article are introduced as follows.

Definition 1 (PE [44]): A variable or vector φ is termed PE if there exist positive constants t_a and ϱ such that $\int_{t-t_a}^t \varphi^T(\tau)\varphi(\tau)d\tau \geq \varrho I\ \forall t \geq 0$.

Lemma 1 (RBFNN [49]): The RBFNN is of the capability to approximate any given continuous function $F(Z) = [f_1(Z), f_2(Z), \ldots, f_n(Z)]^T \in \mathbb{R}^n$ with

$$F(Z) = W^{*T}S(x) + \varepsilon \tag{30}$$

where W^* is the optimal NN weights, and $S(x) = [s_1, s_2, \ldots, s_l]^T \in \mathbb{R}^l$ is the NN regressor vector with s_i being

$$s_i(||x - \mu_i||) = \exp\left[\frac{-(x-\mu_i)^T(x-\mu_i)}{\vartheta_i^2}\right]$$

where $\mu_i (i = 1, \ldots, l)$ denote the centers of NN nodes. ε is the NN approximation error.

Lemma 2 [49]: The regressor vectors of the RBFNN satisfy that $||S(x)|| \leq \bar{S}$ for any given constant vector x, where $\bar{S}$ is a positive constant denoting the boundary of S.

Lemma 3 (Spatially Localized Approximation [41]): For any given trajectory, a function can be approximated by the RBFNN with a limited number of NN nodes located in a local region along $x(t)$ such that

$$f(x) = S_\zeta^T W_\zeta^* + \varepsilon_\zeta \tag{31}$$

where $W_\zeta^* = [w_{l_1}^*, w_{l_2}^*, \ldots, w_{l_{N_\zeta}}^*]$ and $S_\zeta = [s_{l_1}, s_{l_2}, \ldots, s_{l_{N_\zeta}}]^T$ are the elements from W^* and S, receptively, satisfying that $s_{li} > \delta_l$, $N_\zeta \leq N$, where δ_l is small positive constant, and ε_ζ is the corresponding NN approximate error.

Lemma 4 (Partial PE Condition [41]): Consider the RBFNN whose centers are placed on a regular lattice, and we can obtain that the regressor S_ζ is PPE for any given recurrent trajectory $x(t)$.

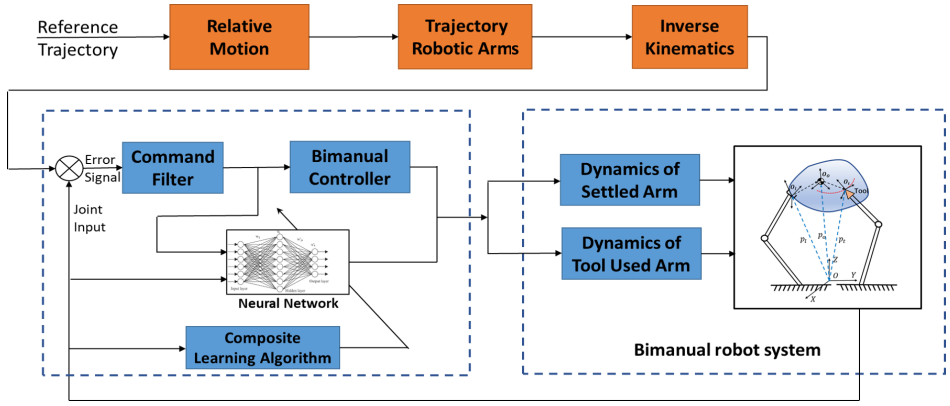

Fig. 2. Overall framework of the proposed bimanual control system.

Remark 2: For most existing estimation schemes, the PE condition is required for the convergence of estimation. However, the PE condition in Definition 1 is very stringent for neural networks due to the requirement of excitation of S for all time. The PPE condition presented in Lemma 4 indicates that only partial regressors S_ζ need to be activated. Thus, the required condition is claimed to be relaxed. This condition has been applied in a wide range of implementations of neural control, such as robotics and unmanned underwater vehicles [50].

III. Controller Design

Revisiting the objective of the controller design for the dual-arm robot system, we aim to control the TME to follow a reference path on the surface of the grasped object while maintaining the contact force at the desired level. Defining p_{od} as the desired trajectory of the object, p_{tod} is the desired path for the TM, while ξ_d is the desired constraint force. To facilitate the theoretical analysis, the Cartesian space variables, p_{od} and p_{tod}, are transformed to desired joint variables q_{cd} by using the inverse kinematics. Then, the control objective can be reformulated to follow the reference trajectory q_{cd} in the joint space. The proposed control system is depicted as shown in Fig. 2.

A. Command Filtered Control

Let $x_1 = q_c$, $x_2 = \dot{q}_c$, and the robotic dynamics in (29) can be rewritten as

$$\begin{aligned}\dot{x}_1 &= x_2\\ \dot{x}_2 &= \mathcal{M}(q_c)^{-1}\Big(N^T\tau_d - \mathcal{C}(q_c,\dot{q}_c)x_2 - \mathcal{G}(q_c)\Big).\end{aligned}\tag{32}$$

Then, we define the tracking errors as follows:

$$\begin{aligned}e &= x_1 - x_{1d}\\ z &= x_2 - \alpha^c\end{aligned}\tag{33}$$

where $x_{1d} = q_{cd}$, and α^c is the filtered virtual control term to be specified later.

The controller is designed with the following steps. Consider a Lyapunov candidate V_1 with respect to the tracking error e as follows:

$$V_1 = \frac{1}{2}e^T e.\tag{34}$$

Taking time derivative of (34), we have

$$\dot{V}_1 = e^T\dot{e}.\tag{35}$$

Substituting (33) into (35) yields

$$\dot{V}_1 = e^T(\dot{x}_1 - \dot{x}_{1d}).\tag{36}$$

For the conventional backstepping technique, a virtual control term $\alpha = \dot{x}_{1d} - \Lambda e$ is designed with $\Lambda = \text{diag}[k_{11}, k_{12}, \ldots, k_{1n}]$ being a positive definite matrix. Here, we employ a filtered virtual controller α^c to replace α as

$$\begin{cases}\dot{\upsilon}_1 = \upsilon_2\\ \dot{\upsilon}_2 = -2\kappa\sigma\upsilon_2 + \sigma^2(\alpha - \upsilon_1)\end{cases}\tag{37}$$

where $\upsilon_1 = \alpha^c$, $\upsilon_2 = \dot{\alpha}^c$ with $\upsilon_1(0) = \alpha(0)$, $\upsilon_2(0) = 0$, σ is a positive constant denoting the natural frequency, and κ is a positive damping ratio.

Lemma 5: [43] For the command filtered defined in (37), there exists a small positive constant d_α such that $|\tilde{\alpha}| \le d_\alpha$ with a sufficiently large frequency σ.

Remark 3: Note that the filtered error $\tilde{\alpha} = \alpha^c - \alpha$ is bounded in terms of Lemma 5. The command filter here can be used to simplify the process of calculating the command derivative signals.

According to the definition of α^c, (36) can be rewritten as follows:

$$\dot{V}_1 = e^T(z - \Lambda e + \tilde{\alpha}).\tag{38}$$

Considering the system state z, let us define a Lyapunov candidate V_2 as follows:

$$V_2 = \frac{1}{2}z^T\mathcal{M}z.\tag{39}$$

The time derivative of (39) gives

$$\dot{V}_2 = z^T\mathcal{M}\dot{z} + \frac{1}{2}z^T\dot{\mathcal{M}}z.\tag{40}$$

Combining (32) and (33) and substituting them into (40), we can obtain that

$$\begin{aligned}\dot{V}_2 &= z^T\Big(N^T\tau_d - \mathcal{C}(q_c,\dot{q}_c)x_2 - \mathcal{G}(q_c)\Big)\\ &\quad -\frac{1}{2}z^T\mathcal{M}\dot{\alpha}^c + \frac{1}{2}z^T\dot{\mathcal{M}}z\\ &= z^T\Big(N^T\tau_d - \mathcal{M}\dot{\alpha}^c - \mathcal{C}(q_c,\dot{q}_c)\alpha^c - \mathcal{G}(q_c)\Big)\\ &\quad -z^T\mathcal{C}z + \frac{1}{2}z^T\dot{\mathcal{M}}z.\end{aligned}\tag{41}$$

Note that the property in (25) holds, and (41) can be rewritten as

$$\dot{V}_2 = z^T\Big(N^T\tau_d + f(\dot{\alpha}^c, \alpha^c, \dot{q}_c, q_c)\Big) \tag{42}$$

where $f(x_c) = -\mathcal{M}\dot{\alpha}^c - \mathcal{C}(q_c, \dot{q}_c)\alpha^c - \mathcal{G}(q_c)$ and $x_c = [\dot{\alpha}^c, \alpha^c, \dot{q}_c, q_c]$.

B. Composite Learning

It should be emphasized that the robot dynamics $f(x_c)$ is strongly nonlinear and often not available in practice. Therefore, an RBFNN is introduced to address the nonlinearity. The RBFNN is constructed to approximate $f(x_c)$ as

$$f(x_c) = \hat{W}^T S \tag{43}$$

where S is a vector of the radial basis function, and $\hat{W}$ is the estimation of W^*.

To improve the learning performance of the neural weights, the following auxiliary term is introduced to design the learning law:

$$\begin{aligned} P &= \int_0^t e^{-\gamma(t-\tau)} S(x_c(\tau))S(x_c(\tau))^T d\tau \\ Q &= \int_0^t e^{-\gamma(t-\tau)} S(x_c(\tau))\Big(S(x_c(\tau))^T W^* + \varepsilon\Big)d\tau \end{aligned} \tag{44}$$

where γ is a forgetting constant.

Remark 4: $e^{-\gamma(t-\tau)}$ in (44) is introduced to remain the boundedness of the auxiliary terms: P and Q. This is because noises and external disturbances may be accumulated over time in the integration, which leads to the oversize of P, Q. The forgetting factor γ can help to maintain the amplitude of P and Q by reducing the influence of the past time variables and, thus, help to improve the convergence speed.

Then, let us define a novel predication error as

$$\epsilon = Q(t) - P(t)\hat{W}. \tag{45}$$

Note that Q can be calculated from (32) and (43) as $Q = \int_0^t e^{-\gamma(t-\tau)} S(S^T W^* + \varepsilon)d\tau = \int_0^t e^{-\gamma(t-\tau)} S\tau' d\tau$, where $\tau' = N^T\tau_d$.

The combination of (44) and (45) yields

$$\epsilon = P(t)\tilde{W} + \varepsilon_a \tag{46}$$

where $\varepsilon_a = \int_0^t e^{-\gamma(t-\tau)} S(x_c(\tau))\varepsilon(\tau)d\tau$ and $\tilde{W} = W^* - \hat{W}$. According to Lemma 3, $f(x_c)$ can be expressed by partial NN regression vector as

$$f(x_c) = S_\zeta^T W_\zeta^* + \varepsilon_\zeta \tag{47}$$

where ε_ζ satisfies that $|\varepsilon_\zeta| < d_{\varepsilon_\zeta}$, and $d_{\varepsilon_\zeta} \in \mathbb{R}^+$ represents the bound of ε_ζ. Here, a projection $\Theta(\cdot)$ is defined to choose the excitation elements such that $S_\zeta = \Theta(S)$. Then, P can be reformulated by $P_\zeta = \int_0^t e^{-\gamma(t-\tau)} S_\zeta(\tau)S_\zeta^T(\tau)d\tau$, and we can rewrite (46) by $\epsilon_\zeta = P_\zeta\tilde{W}_\zeta + \varepsilon_{a\zeta}$.

The neural weight adaptation law is designed as follows:

$$\dot{\hat{W}}_\zeta = \beta\Gamma(S_\zeta(x_c)z^T N^T + \lambda_\omega\epsilon_\zeta) \tag{48}$$

where β and λ_ω are the positive gain matrices, and Γ is a projection operator defined as

$$\Gamma(u) = \begin{cases} u, & \text{if } ||\hat{W}|| \le \sigma_\omega \text{ or } (||\hat{W}|| = \sigma_\omega \,\&\, \hat{W}^T u \le 0) \\ u - \dfrac{\hat{W}^T\hat{W}}{||\hat{W}||}u, & \text{otherwise} \end{cases} \tag{49}$$

where σ_ω is a positive constant representing the boundary of NN weights.

Remark 5: The projection operator Γ is defined for the adaptation of $\hat{W}$ such that $\hat{W}$ is kept as its original form if $\hat{W}$ is inside the compact set $\Omega_{\sigma\omega}$ or on the boundary but going into the compact set. Otherwise, $\hat{W}$ is projected to the boundary of $\Omega_{\sigma\omega}$.

Then, the controller can be designed as

$$\tau_d = (N^T)^{-1}(-K_2 z - e) - \hat{W}_\zeta^T S_\zeta \tag{50}$$

where $K_2 = \text{diag}\{k_{21}, k_{22}, \ldots, k_{2n}\}$ is a positive definite gain matrix.

C. Stability Analysis

Theorem 1: Consider the closed-loop dual-arm robot dynamic system described by (29), using the control law (27) and (50), and the adaptation law (48), under the assumption that the PPE condition is satisfied; then, the following results hold: 1) the tracking errors e and z of the system converge to a small neighborhood containing the origin; 2) the estimation errors $\tilde{W}_\xi$ converge to a small neighborhood containing the origin; and 3) the error of the contact force e_ξ is bounded.

Proof: Let us consider a Lyapunov candidate V_3 as follows:

$$V_3 = \frac{1}{2\beta}\tilde{W}_\zeta{}^T\tilde{W}_\zeta. \tag{51}$$

Combining (34), (39), and (51), we can define $V = V_1 + V_2 + V_3$. Taking the derivative of V with respect to time yields

$$\dot{V} = \dot{V}_1 + \dot{V}_2 + \dot{V}_3. \tag{52}$$

Then, substituting (38) and (42) into (52), we can obtain that

$$\dot{V} = e^T(z - \Lambda e + \tilde{\alpha}) + z^T(N^T\tau_d + f(x_c)) - \frac{1}{\beta}\tilde{W}_\zeta^T\dot{\hat{W}}_\zeta. \tag{53}$$

Substituting the designed controller (50) to (53), we have

$$\begin{aligned} \dot{V} &= e^T(z - \Lambda e + \tilde{\alpha}) + z^T\Big(-K_2 z - e\Big) \\ &\quad + z^T f(x_c) - z^T N^T\hat{W}_\zeta^T S_\zeta - \frac{1}{\beta}\tilde{W}_\zeta^T\dot{\hat{W}}_\zeta \\ &= -e^T\Lambda e - z^T K_2\, z + e^T\tilde{\alpha} \\ &\quad + z^T(f(x_c) - N^T\hat{W}_\zeta^T S_\zeta) - \frac{1}{\beta}\tilde{W}_\zeta^T\dot{\hat{W}}_\zeta. \end{aligned} \tag{54}$$

Then, substituting the adaptation law (48) into (54) yields

$$\begin{aligned} \dot{V} &= -e^T\Lambda e - z^T K_2\, z + e^T\tilde{\alpha} \\ &\quad + z^T(f(x_c) - N^T\hat{W}_\zeta^T S_\zeta) - \frac{1}{\beta}\tilde{W}_\zeta^T\dot{\hat{W}}_\zeta \\ &= -e^T\Lambda e - z^T K_2\, z + e^T\tilde{\alpha} + z^T N^T\tilde{W}_\zeta^T S_\zeta \\ &\quad + z^T\varepsilon_\zeta - \tilde{W}_\zeta^T\Gamma(S_\zeta(Nz)^T + \lambda_\omega\epsilon_\zeta). \end{aligned} \tag{55}$$

From the projection operation $\Gamma(\cdot)$ defined in (49), $\hat{W}_\zeta(t)$ is remained in $\hat{W}_\zeta(t) \in \Omega_{\sigma\omega}$ by choosing proper initial condition $\hat{W}_\zeta(0) \in \Omega_{\sigma\omega}$. Since the initial condition is easy to guarantee, we can obtain $\tilde{W}_\zeta^T S_\zeta(x_c)(Nz)^T - \tilde{W}_\zeta^T \Gamma(S_\zeta(x_c)(Nz)^T + \lambda_\omega \epsilon_\zeta) \le -\lambda_\omega \tilde{W}_\zeta^T \epsilon_\zeta$.

Then, (55) can be rewritten as

$$\dot{V} \le -e^T \Lambda e - z^T K_2\, z + e^T \tilde{\alpha} + z^T \varepsilon_\zeta - \lambda_\omega \tilde{W}_\zeta^T \epsilon_\zeta. \quad (56)$$

Reminding the definition of ϵ in (46), we can obtain that

$$\begin{aligned}\dot{V} &\le -e^T \Lambda e - z^T K_2\, z + e^T \tilde{\alpha} + z^T \varepsilon_\zeta \\ &\quad -\lambda_\omega \tilde{W}_\zeta^T (P_\zeta \tilde{W}_\zeta + \varepsilon_{a\zeta}) \\ &\le -e^T \Lambda e - z^T K_2\, z + e^T \tilde{\alpha} - \lambda_\omega \tilde{W}_\zeta^T P_\zeta \tilde{W}_\zeta \\ &\quad + z^T \varepsilon_\zeta - \lambda_\omega \tilde{W}_\zeta^T \varepsilon_{a\zeta}. \end{aligned} \quad (57)$$

Note that $\tilde{\alpha}$ is bounded by $|\tilde{\alpha}| \le d_\alpha$, and ε is bounded by $|\varepsilon| \le d_\varepsilon$. Employing Young's inequality $-a^2 + 2ab - b^2 \le 0$ here, we can obtain

$$\dot{V} \le -\frac{1}{2} e^T \Lambda e - \frac{1}{2} z^T K_2\, z - \lambda_\omega \tilde{W}_\zeta^T P \tilde{W}_\zeta + \mu - \lambda_\omega \tilde{W}_\zeta^T \varepsilon_{a\zeta} \quad (58)$$

where $\mu = (d_\alpha)^2/(2\min(k_{1i})) + (d_\varepsilon)^2/(2\min(k_{2i}))$.

Then, according to the assumption that PPE condition is guaranteed, we can obtain that P_ζ is positive definite and $\lambda(P_\zeta) \ge \sigma_P I$ [51], where σ_P is a small constant satisfying $\sigma_P \le \lambda_{\min}(P_\zeta)$, and I is an unit matrix with proper dimension. Therefore, (58) can be rewritten as

$$\begin{aligned}\dot{V} &\le -\frac{1}{2} e^T \Lambda e - \frac{1}{2} z^T K_2\, z - \lambda_\omega \sigma_P \tilde{W}_\zeta^T \tilde{W}_\zeta \\ &\quad + \mu - \lambda_\omega \tilde{W}_\zeta^T \varepsilon_{a\zeta} \\ &\le -\frac{1}{2} e^T \Lambda e - \frac{1}{2} z^T K_2\, z - \frac{1}{2} \lambda_\omega \sigma_P \tilde{W}_\zeta^T \tilde{W}_\zeta \\ &\quad + \mu - \lambda_\omega \tilde{W}_\zeta^T \varepsilon_{a\zeta} - \frac{1}{2} \lambda_\omega \sigma_P \tilde{W}_\zeta^T \tilde{W}_\zeta \\ &\le -\frac{1}{2} e^T \Lambda e - \frac{1}{2} z^T K_2\, z - \frac{1}{2} \lambda_\omega \sigma_P \tilde{W}_\zeta^T \tilde{W}_\zeta + \mu + \varpi \end{aligned} \quad (59)$$

where

$$\varpi = \lambda_\omega \frac{||\bar{S} d_\varepsilon||^2}{2\sigma_P}$$

and again Young's inequality

$$-\frac{1}{2} \lambda_\omega \sigma_P (W_\zeta^T \tilde{W}_\zeta + 2 \frac{1}{\sigma_P} \tilde{W}_\zeta^T \varepsilon_{a_\zeta}) \le \lambda_\omega \frac{||\varepsilon_{a\zeta}||^2}{2\sigma_P}$$

is employed with $||\varepsilon_{a\zeta}|| \le ||\bar{S} d_{\varepsilon_\zeta}||$.

Comparing with the elements of V defined in (34), (39), and (51), we can obtain that

$$\dot{V} \le -\vartheta_v V + \delta \quad (60)$$

where $\vartheta_v = \min\{\lambda_{\min}(\Lambda), \lambda_{\min}(K_2)/\lambda_{\max}(\mathcal{M}), \lambda_\omega \sigma_P/\beta\}$, and $\delta = \mu + \varpi$.

Integrating on both sides of (60) to solve the differential inequality equation yields

$$V(t) \le V(0) e^{-\vartheta_v t} + \delta/\vartheta_v. \quad (61)$$

Therefore, we can derive that there exists a compact set $\{\Omega : V \le \delta/\vartheta_v\}$ such that the states outside Ω could enter into it and remain in Ω in the future time. In addition, the tracking errors e, z, and $\tilde{W}_\zeta$ converge to a small neighborhood around zero, and the size of the compact set can be diminished by increasing k_{1i}, k_{2i}, and λ_ω.

Then, let us revisit the stability of contact force. Substituting the error dynamics (50) into (28), we have

$$(N^T)^{-1}(-K_2 z - e + N^T \tilde{W}^T S) = J_c(q_c)\eta_t \left(e_\xi + k_\xi \int_0^t e_\xi d\tau\right). \quad (62)$$

According to the results of Theorem 1, as z, e, $\tilde{W}$, and ε converge to a small neighborhood around zero and N and J_c are also full column rank matrices, we can obtain

$$e_\xi + k_\xi \int_0^t e_\xi d\tau \le U \quad (63)$$

where $U = \Psi(\max\{k_{2i}\}|\sigma_z| + |\sigma_e| + |\sigma_w|\bar{S} + |d_\varepsilon|)$, with σ_z, σ_e, and $\sigma_{\tilde{W}}$ being the boundary of the compact set of the tracking errors, and $\Psi = |\lambda_{\max}((N^T J_c)^{-1}/\eta_t)|$ is a bounded parameter. Since e_ξ is continuous and the inequality in (63) holds, we can conclude that the error of the contact force e_ξ is bounded in terms of the boundedness of U, while the magnitude of e_ξ can be diminished through the increases of k_ξ and η_t.

This completes the proof.

Remark 6: Different from the traditional NN learning algorithms, a leakage term $\lambda_\omega \epsilon_\zeta$ is designed in (48). As discussed in Section III-B, ϵ_ζ can be described by a function of the estimation error of the NN weights. Therefore, integrating ϵ_ζ in the adaptation law will result in a quadratic term $\lambda_\omega \tilde{W}_\zeta^T P_\zeta \tilde{W}_\zeta$ in the Lyapunov function (57). Thus, the estimation error and the tracking error are claimed to be converged simultaneously. In comparison to the widely used NN adaptation algorithm in [31] and [41], the proposed composite learning algorithm is clearly different by using the prediction error for ensuring the convergence. More detailed illustrations will be given in Section IV with numerical examples.

IV. Simulation

To verify the validity of the proposed bimanual control algorithm, we employ a bimanual robot consists of a manipulator with two revolute joints and an arm with one translate joint, to rigidly grasp a rectangular object, as shown in Fig. 3. The object coordinate frame $X_O O_O Y_O$ is attached to the object mass center O_o. The joint variables of the SM and TM are set to $q_s = x$ and $q_t = [q_1, q_2]$, with base displacements of $b = 0.5$ and $a = 0.3$. From Fig. 3, we can derive the position vectors of the object coordinate and the tool coordinate as $p_o = [q_s, b]^T$, $p_c = [l_1 \cos(q_1) + l_2 \cos(q_1+q_2)+a,\ l_1 \sin(q_1)+l_2 \sin(q_1+q_2)]^T$. We assume that the tool-used manipulator tracks a straight line on the object as $\rho(p_{to}) = x_{to} - y_{to} = 0$. The joint coordinate is chosen to be $d(q) = [q_{t1}, q_{t2}]^T$ and $q_s(q) = l_1(\cos(q_1) - \sin(q_1)) + l_2(\cos(q_1 + q_2) - \sin(q_1 + q_2)) + a$.

Then, the kinematic relationship of the bimanual robot can be derived by

$$J_t = \begin{bmatrix} -l_1 \sin(q_1) - l_2 \sin(q_1 + q_2) & -l_2 \sin(q_1 + q_2) \\ -l_1 \cos(q_1) + l_2 \cos(q_1 + q_2) & -l_2 \cos(q_1 + q_2) \end{bmatrix}$$

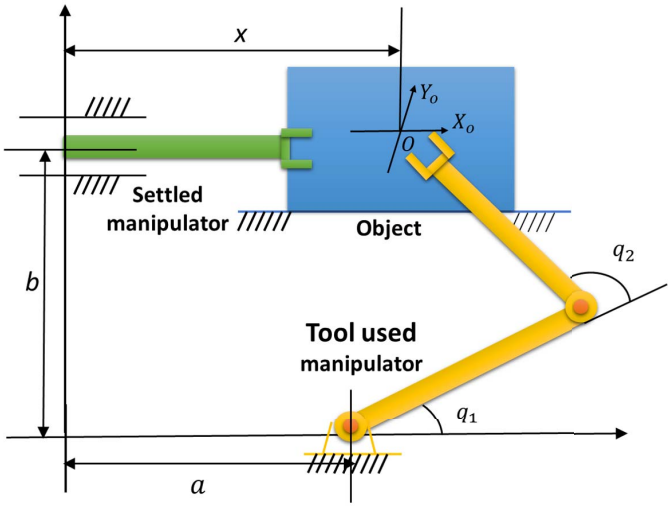

Fig. 3. Simulated bimanual robot system.

and

$$J_s = \begin{bmatrix} 1 \\ 0 \end{bmatrix}$$

and

$$N^T = \begin{bmatrix} 1 & 0 & -l_1(s_1 + c_1) - l_2(s_{12} + c_{12}) \\ 0 & 1 & -l_2(s_{12} + c_{12}) \end{bmatrix}$$
$$\dot{N}^T = \begin{bmatrix} 0 & 0 & l_1(s_1 - c_1)\dot{q}_1 - l_2(s_{12} - c_{12})(\dot{q}_1 + \dot{q}_2) \\ 0 & 1 & l_2(s_{12} - c_{12})(\dot{q}_1 + \dot{q}_2) \end{bmatrix}.$$

The TM robot is chosen as a two-link manipulator, and its dynamics is given by

$$M_t(q_t)\ddot{q}_t + C_t(q_t, \dot{q}_t)\dot{q}_t + G_t(q_t) = \tau_t + J_t^T(q_t) f_t \quad (64)$$

where

$$M_t = \begin{bmatrix} M_{11} & M_{12} \\ M_{21} & M_{22} \end{bmatrix}$$

with

$$M_{11} = (m_1 + m_2)l_1^2 + m_2 l_2^2 + 2m_2 l_1 l_2 \cos(q_2) + I_1 + I_2; M_{12} = m_2 l_2^2 + m_2 l_1 l_2 \cos(q_2) + I_2;$$
$$M_{22} = m_2 l_2^2 + I_2$$

$$C_t = \begin{bmatrix} -c\dot{q}_2 & -c(\dot{q}_1 + \dot{q}_2) \\ c\dot{q}_1 & 0 \end{bmatrix}$$

with $c = m_2 l_1 l_2 \sin(q_2)$; and $G_t = [v_1 g, v_2 g]^T$, with $v_1 = (m_1 + m_2)l_1 \cos(q_2) + m_2 l_2 \cos(q_1 + q_2)$ and $v_2 = m_2 l_2 \cos(q_1 + q_2)$. The SM is a one-link translate manipulator, and its dynamics can be described as

$$M_s \ddot{q}_s = \tau_s - J_s(q_s) D_t^T \eta_t \xi \quad (65)$$

where $m_s = 0.1$, $D_t = I$, and $\eta_t = [1/\sqrt{2}, -1/\sqrt{2}]^T$. The parameters of the links are given in Table II.

The tracking trajectory of the object is given as $x_{\text{od}} = 0.15(1 - \sin(t + 12))$, and the desired trajectory on the object surface is defined as $x_{\text{tod}} = -1.5/12\cos(t + 2)$ and $y_{\text{tod}} = -1.5/12\cos(t + 2)$. The desired contact force is set to $\xi_d = 2N$. The initial conditions of the robot joints are set to $q_1 = 0$, $q_2 = 2$, $\dot{q}_1 = 0$, and $\dot{q}_2 = 0$. The initial object position is set

TABLE II
NOMENCLATURE OF THE COORDINATION SYSTEM

Parameter	Value
link lengths	$l_1 = l_2 = 0.5$m
link offset	$a = 0.2$m, $b = 0.5$m
link mass	$m_1 = m_2 = 0.5$ kg, $m_s = 0.1$kg
link inertia	$I_1 = I_2 = 0.3$ kg
gravity constant	$g = 9.8$ m/s^2

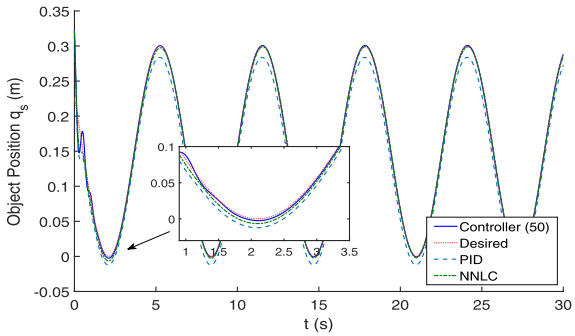

Fig. 4. Tracking performance of the object position x_{od}.

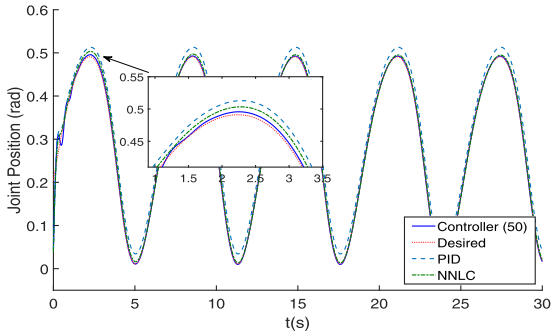

Fig. 5. Tracking performance of the joint position q_1.

to $x_{\text{od}} =$ and $\dot{x}_{\text{od}} = 0$. The control gains are selected as $K_2 = \text{diag}(15, 10)$ and $\Lambda = \text{diag}(15, 20)$, with the command filtered parameters being $\kappa = 1$ and $\sigma = 50$. Assume that the robot dynamics M_t, M_s, C_t, and G_t are unknown to users, and an RBFNN with $2^8 = 256$ nodes is employed to cancel the effect of dynamic uncertainties, with the centers placing on a regular lattice as $[-1, 1] \times [-1.5, 1.5] \times [-0.5, 0.5] \times [-0.5, 0.5] \times [-1, 1] \times [-1, 1] \times [-1, 1] \times [-1, 1]$. The gains of the NN are selected as $\beta = 15$ and $\lambda_\omega = 1$. The NN is initialized with $W(0) = 0$ and $S(0) = 0$. To further verify the superiority of the proposed bimanual control algorithm, comparative studies have also been conducted using the traditional PID controller and the NN learning control (NNLC) in [41].

Next, we analyze the tracking performance of the proposed control scheme. Figs. 4–6 show the typical performance of q_s, q_1, and q_2 under the proposed controller, a well-turned PID controller, and an NNLC controller. We can see from the figures that the proposed composite learning control scheme follows the reference trajectory very well with high tracking precision. On the other hand, the tracking of NNLC is

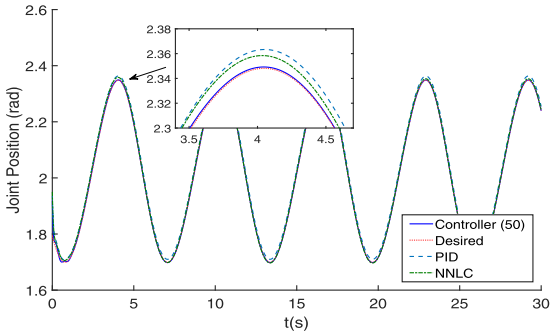

Fig. 6. Tracking performance of the joint position q_2.

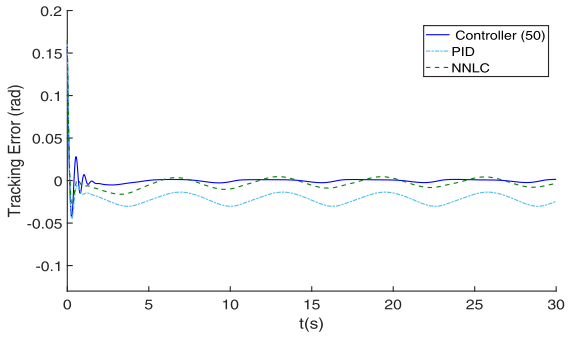

Fig. 7. Tracking error of joint angles e_1.

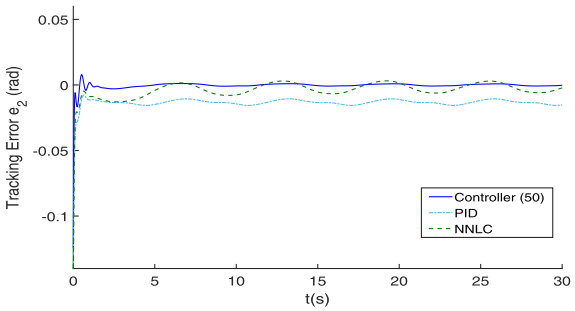

Fig. 8. Tracking error of joint angles e_2.

better than the PID controller; however, steady-state position tracking errors still can be observed. The tracking errors of the controllers were depicted, as shown in Figs. 7 and 8. We can see from the figures that the tracking errors of the proposed controller are less than 0.01 rad in a few seconds with fast convergence. As a comparison, the PID controller demonstrates large tracking errors in the steady state. Although the NNLC achieves adequate control precision, the setting time is less than the proposed controller, and also small tracking errors existed. The peak of the tracking error of the proposed controller is nearly 70% lower than the case of the NNLC controller. Therefore, the proposed controller demonstrates better control performance than the other two controllers. Fig. 9 shows the filter error of $\tilde{\alpha}_c$. We can see that the filter error is less than 0.001, which implies an accurate approximation of the command derivative.

The NN learning performance and weight convergence are depicted, as shown in Figs. 10–16, where the proposed composite learning algorithm and the NNLC are compared. We can

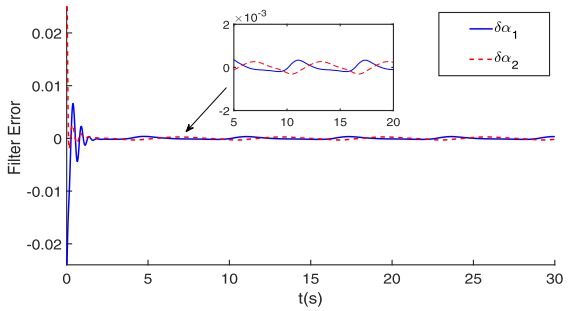

Fig. 9. Filter error of the command filter $\tilde{\alpha}$.

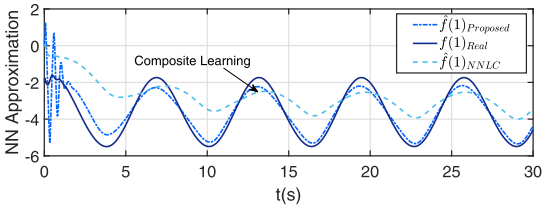

Fig. 10. Approximation performance of the NN with and without composite learning.

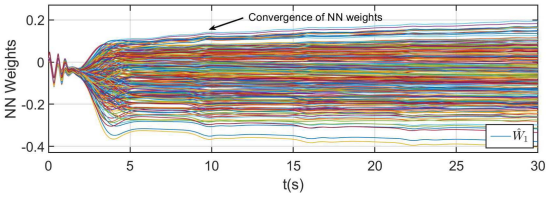

Fig. 11. Converge of NN weights W_1 with composite learning.

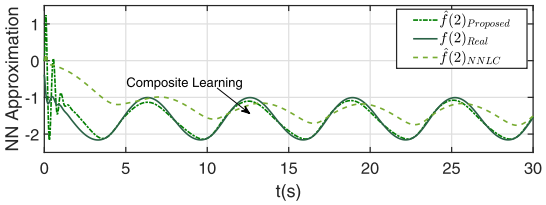

Fig. 12. Approximation performance of the NN with and without composite learning.

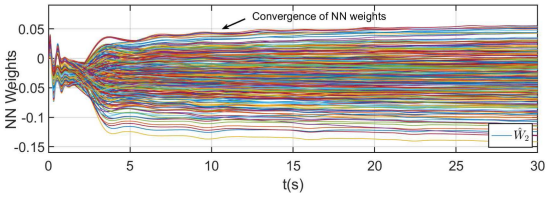

Fig. 13. Converge of NN weights W_2 with composite learning.

see from the figures that the approximation of the unknown function $f(x_c)$ is successful (dotted line ".-") by the proposed composite learning method. For the NNLC, the approximation of $f(x_c)$ is not achieved (dotted line "–"). Figs. 11–15 show that the NN weight parameters converge to constant values in about 10 s, a time interval after which the overall tracking performance remains substantially unvaried. The convergence of the NN norm is depicted in Fig. 16, where the convergence

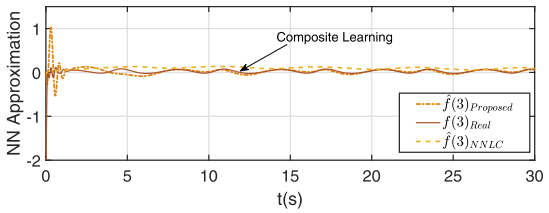

Fig. 14. Approximation performance of the NN with and without composite learning.

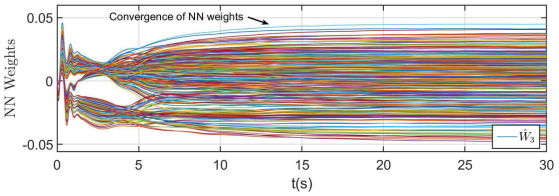

Fig. 15. Converge of NN weights W_3 with composite learning.

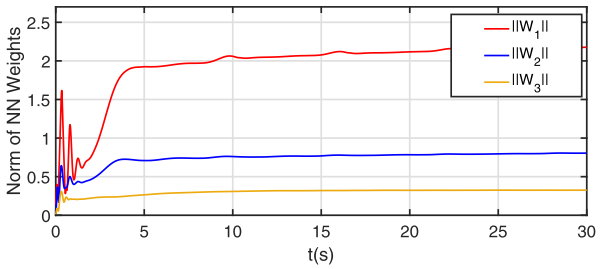

Fig. 16. Norm of the NN weights.

of NN weight can be observed in a more direct manner. Thus, the effectiveness of the proposed composite learning algorithm is demonstrated.

V. Conclusion

In this article, we investigate the control design for bimanual robots to perform the relative motion. The dynamic modeling of the bimanual robot system is conducted using a holonomic kinematic constraint. The command filtered backstepping method is employed for the controller design. Moreover, a composite learning algorithm is developed for the adaptation of RBFNN by using the integration of learning errors. The stability analysis is conducted through the Lyapunov theory. Finally, the effectiveness and superiority of the proposed composite learning algorithm are demonstrated through comparative simulation studies. Future work includes the asymmetric bimanual control of dual-arm robot with the nonrigid surface by using the reinforcement learning method.

References

[1] M. Fuchs *et al.*, "Rollin' justin–design considerations and realization of a mobile platform for a humanoid upper body," in *Proc. IEEE Int. Conf. Robot. Autom.*, May 2009, pp. 4131–4137.

[2] R. Liza Ahmad Shauri and K. Nonami, "Assembly manipulation of small objects by dual-arm manipulator," *Assem. Autom.*, vol. 31, no. 3, pp. 263–274, Aug. 2011.

[3] R. Bogue, "Robots to aid the disabled and the elderly," *Ind. Robot, Int. J.*, vol. 40, no. 6, pp. 519–524, Oct. 2013.

[4] Y.-H. Liu, Y. Xu, and M. Bergerman, "Cooperation control of multiple manipulators with passive joints," *IEEE Trans. Robot. Autom.*, vol. 15, no. 2, pp. 258–267, Apr. 1999.

[5] M. Li, H. Yin, K. Tahara, and A. Billard, "Learning object-level impedance control for robust grasping and dexterous manipulation," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 6784–6791.

[6] C. Yang, C. Chen, W. He, R. Cui, and Z. Li, "Robot learning system based on adaptive neural control and dynamic movement primitives," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 3, pp. 777–787, Mar. 2019.

[7] N. Sommer, M. Li, and A. Billard, "Bimanual compliant tactile exploration for grasping unknown objects," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, May 2014, pp. 6400–6407.

[8] C. Yang, C. Zeng, C. Fang, W. He, and Z. Li, "A DMPs-based framework for robot learning and generalization of humanlike variable impedance skills," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 3, pp. 1193–1203, Jun. 2018.

[9] Y.-H. Liu and S. Arimoto, "Decentralized adaptive and nonadaptive position/force controllers for redundant manipulators in cooperations," *Int. J. Robot. Res.*, vol. 17, no. 3, pp. 232–247, Mar. 1998.

[10] O. Khatib, K. Yokoi, K. Chang, D. Ruspini, R. Holmberg, and A. Casal, "Vehicle/arm coordination and multiple mobile manipulator decentralized cooperation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst. (IROS)*, vol. 2, Nov. 1996, pp. 546–553.

[11] J. H. Chung, B.-J. Yi, and W. K. Kim, "Analysis of internal loading at multiple robotic systems," *J. Mech. Sci. Technol.*, vol. 19, no. 8, pp. 1554–1567, Aug. 2005.

[12] L. Feng-yun and L. Tian-sheng, "Development of a robot system for complex surfaces polishing based on CL data," *Int. J. Adv. Manuf. Technol.*, vol. 26, nos. 9–10, pp. 1132–1137, Oct. 2005.

[13] L. Wu, K. Cui, and S. B. Chen, "Redundancy coordination of multiple robotic devices for welding through genetic algorithm," *Robotica*, vol. 18, no. 6, pp. 669–676, Nov. 2000.

[14] C. Zeng, C. Yang, Z. Chen, and S.-L. Dai, "Robot learning human stiffness regulation for hybrid manufacture," *Assem. Autom.*, vol. 38, no. 5, pp. 539–547, Nov. 2018, doi: 10.1108/aa-02-2018-019.

[15] J. Lee, P. H. Chang, and R. S. Jamisola, "Relative impedance control for dual-arm robots performing asymmetric bimanual tasks," *IEEE Trans. Ind. Electron.*, vol. 61, no. 7, pp. 3786–3796, Jul. 2014.

[16] Z. Li, W. Yuan, S. Zhao, Z. Yu, Y. Kang, and C. L. P. Chen, "Brain-actuated control of dual-arm robot manipulation with relative motion," *IEEE Trans. Cognit. Develop. Syst.*, vol. 11, no. 1, pp. 51–62, Mar. 2019.

[17] Y.-J. Liu, S. Lu, and S. Tong, "Neural network controller design for an uncertain robot with time-varying output constraint," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 47, no. 8, pp. 2060–2068, Aug. 2017.

[18] J. Na, B. Jing, Y. Huang, G. Gao, and C. Zhang, "Unknown system dynamics estimator for motion control of nonlinear robotic systems," *IEEE Trans. Ind. Electron.*, vol. 67, no. 5, pp. 3850–3859, May 2020.

[19] C. Yang, C. Chen, N. Wang, Z. Ju, J. Fu, and M. Wang, "Biologically inspired motion modeling and neural control for robot learning from demonstrations," *IEEE Trans. Cogn. Develop. Syst.*, vol. 11, no. 2, pp. 281–291, Jun. 2019.

[20] Z. Zhao, C. K. Ahn, and H.-X. Li, "Dead zone compensation and adaptive vibration control of uncertain spatial flexible riser systems," *IEEE/ASME Trans. Mechatronics*, vol. 25, no. 3, pp. 1398–1408, Jun. 2020.

[21] Y. Zhu and W. X. Zheng, "Observer-based control for cyber-physical systems with periodic DoS attacks via a cyclic switching strategy," *IEEE Trans. Autom. Control*, vol. 65, no. 8, pp. 3714–3721, Aug. 2020.

[22] Z. Zhao, C. K. Ahn, and H.-X. Li, "Boundary antidisturbance control of a spatially nonlinear flexible string system," *IEEE Trans. Ind. Electron.*, vol. 67, no. 6, pp. 4846–4856, Jun. 2020.

[23] M. Wang, Y. Zhang, and C. Wang, "Learning from neural control for non-affine systems with full state constraints using command filtering," *Int. J. Control*, vol. 93, no. 10, pp. 2392–2406, 2020.

[24] J. Na, S. Wang, Y.-J. Liu, Y. Huang, and X. Ren, "Finite-time convergence adaptive neural network control for nonlinear servo systems," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2568–2579, Jun. 2020.

[25] L. Liu, Y. Liu, D. Li, S. Tong, and Z. Wang, "Barrier Lyapunov function-based adaptive fuzzy FTC for switched systems and its applications to resistance–inductance–capacitance circuit system," *IEEE Trans. Cybern.*, vol. 50, no. 8, pp. 3491–3502, Aug. 2020.

[26] F. Luan, J. Na, Y. Huang, and G. Gao, "Adaptive neural network control for robotic manipulators with guaranteed finite-time convergence," *Neurocomputing*, vol. 337, pp. 153–164, Apr. 2019.

[27] C. Liu, G. Wen, Z. Zhao, and R. Sedaghati, "Neural-network-based sliding-mode control of an uncertain robot using dynamic model approximated switching gain," *IEEE Trans. Cybern.*, early access, Mar. 18, 2020, doi: 10.1109/tcyb.2020.2978003.

[28] L. Liu, Y.-J. Liu, A. Chen, S. Tong, and C. L. P. Chen, "Integral barrier Lyapunov function-based adaptive control for switched nonlinear systems," *Sci. China Inf. Sci.*, vol. 63, no. 3, pp. 1–14, Mar. 2020.

[29] X. Li, C. K. Ahn, D. Lu, and S. Guo, "Robust simultaneous fault estimation and nonfragile output feedback fault-tolerant control for Markovian jump systems," *IEEE Trans. Syst., Man, Cybern. Syst.*, vol. 49, no. 9, pp. 1769–1776, Sep. 2019.
[30] L. Liu, Y.-J. Liu, and S. Tong, "Neural networks-based adaptive finite-time fault-tolerant control for a class of strict-feedback switched nonlinear systems," *IEEE Trans. Cybern.*, vol. 49, no. 7, pp. 2536–2545, Jul. 2019.
[31] Y.-J. Liu, L. Tang, S. Tong, and C. L. P. Chen, "Adaptive NN controller design for a class of nonlinear MIMO discrete-time systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 26, no. 5, pp. 1007–1018, May 2015.
[32] L. Liu, Z. Wang, X. Yao, and H. Zhang, "Echo state networks based data-driven adaptive fault tolerant control with its application to electro-mechanical system," *IEEE/ASME Trans. Mechatronics*, vol. 23, no. 3, pp. 1372–1382, Jun. 2018.
[33] C. Chen, Z. Liu, K. Xie, Y. Zhang, and C. L. P. Chen, "Asymptotic fuzzy neural network control for pure-feedback stochastic systems based on a semi-Nussbaum function technique," *IEEE Trans. Cybern.*, vol. 47, no. 9, pp. 2448–2459, Sep. 2017.
[34] W. He, Y. Chen, and Z. Yin, "Adaptive neural network control of an uncertain robot with full-state constraints," *IEEE Trans. Cybern.*, vol. 46, no. 3, pp. 620–629, Mar. 2016.
[35] C. Yang, G. Peng, Y. Li, R. Cui, L. Cheng, and Z. Li, "Neural networks enhanced adaptive admittance control of optimized robot–environment interaction," *IEEE Trans. Cybern.*, vol. 49, no. 7, pp. 2568–2579, Jul. 2019.
[36] 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.
[37] W. He, X. Mu, L. Zhang, and Y. Zou, "Modeling and trajectory tracking control for flapping-wing micro aerial vehicles," *IEEE/CAA J. Automatica Sinica*, early access, Sep. 24, 2020, doi: 10.1109/JAS.2020.1003417.
[38] W. He, T. Wang, X. He, L.-J. Yang, and O. Kaynak, "Dynamical modeling and boundary vibration control of a rigid-flexible wing system," *IEEE/ASME Trans. Mechatronics*, early access, Apr. 21, 2020, doi: 10.1109/tmech.2020.2987963.
[39] C. Yang, Y. Jiang, W. He, J. Na, Z. Li, and B. Xu, "Adaptive parameter estimation and control design for robot manipulators with finite-time convergence," *IEEE Trans. Ind. Electron.*, vol. 65, no. 10, pp. 8112–8123, Oct. 2018.
[40] Z. Miao, Y.-H. Liu, Y. Wang, H. Chen, H. Zhong, and R. Fierro, "Consensus with persistently exciting couplings and its application to vision-based estimation," *IEEE Trans. Cybern.*, early access, Jun. 10, 2019, doi: 10.1109/tcyb.2019.2918796.
[41] C. Wang and D. J. Hill, "Learning from neural control," *IEEE Trans. Neural Netw.*, vol. 17, no. 1, pp. 130–146, Jan. 2006.
[42] C. Wang, M. Wang, T. Liu, and D. J. Hill, "Learning from ISS-modular adaptive NN control of nonlinear strict-feedback systems," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 23, no. 10, pp. 1539–1550, Oct. 2012.
[43] P. M. Patre, W. MacKunis, M. Johnson, and W. E. Dixon, "Composite adaptive control for Euler–Lagrange systems with additive disturbances," *Automatica*, vol. 46, no. 1, pp. 140–147, Jan. 2010.
[44] J. Na, M. N. Mahyuddin, G. Herrmann, X. Ren, and P. Barber, "Robust adaptive finite-time parameter estimation and control for robotic systems," *Int. J. Robust Nonlinear Control*, vol. 25, no. 16, pp. 3045–3071, Nov. 2015.
[45] Y. Pan, T. Sun, Y. Liu, and H. Yu, "Composite learning from adaptive backstepping neural network control," *Neural Netw.*, vol. 95, pp. 134–142, Nov. 2017.
[46] C. Yang, Y. Jiang, J. Na, Z. Li, L. Cheng, and C.-Y. Su, "Finite-time convergence adaptive fuzzy control for dual-arm robot with unknown kinematics and dynamics," *IEEE Trans. Fuzzy Syst.*, vol. 27, no. 3, pp. 574–588, Mar. 2019.
[47] Z. Li, P. Yuen Tao, S. Sam Ge, M. Adams, and W. S. Wijesoma, "Robust adaptive control of cooperating mobile manipulators with relative motion," *IEEE Trans. Syst., Man, Cybern., B, Cybern.*, vol. 39, no. 1, pp. 103–116, Feb. 2009.
[48] S. S. Ge, L. Huang, and T. H. Lee, "Model-based and neural-network-based adaptive control of two robotic arms manipulating an object with relative motion," *Int. J. Syst. Sci.*, vol. 32, no. 1, pp. 9–23, Jan. 2001.
[49] C. Yang, J. Na, G. Li, Y. Li, and J. Zhong, "Neural network for complex systems: Theory and applications," *Complexity*, vol. 2018, pp. 1–2, May 2018.
[50] C. Wang and D. J. Hill, *Deterministic Learning Theory for Identification, Recognition, and Control*. Boca Raton, FL, USA: CRC Press, 2018.
[51] J. Na, G. Herrmann, and K. Zhang, "Improving transient performance of adaptive control via a modified reference model and novel adaptation," *Int. J. Robust Nonlinear Control*, vol. 27, no. 8, pp. 1351–1372, May 2017.

Yiming Jiang (Member, IEEE) received the B.S. degree in automation from Hunan University, Changsha, China, in 2011, the M.S. degree in control theory and engineering from the School of Automation, Guangdong University of Technology, Guangzhou, China, in 2015, and the Ph.D. degree from the School of Control Science and Engineering, South China University of Technology, Guangzhou, in 2019.

He was a Visiting Scholar with the University of Portsmouth, Portsmouth, U.K., from 2017 to 2018. He is currently a Post-Doctoral Fellow with the College of Electrical and Information Engineering, Hunan University, Changsha, China. His research interests include robotics, intelligent control, and human–robot interaction.

Yaonan Wang received the B.S. degree in computer engineering from East China Science and Technology University (ECSTU), Fuzhou, China, in 1981, and the M.S. and Ph.D. degrees in electrical engineering from Hunan University, Changsha, China, in 1990 and 1994, respectively.

From 1994 to 1995, he was a Post-Doctoral Research Fellow with the National University of Defence Technology, Changsha. From 1981 to 1994, he was with ECSTU. From 1998 to 2000, he was a Senior Humboldt Fellow in Germany. From 2001 to 2004, he was a Visiting Professor with the University of Bremen, Bremen, Germany. He has been a Professor with Hunan University since 1995. He is an Academician of the Chinese Academy of Engineering. His current research interests include robotics, intelligent perception and control, and computer vision for industrial applications.

Zhiqiang Miao (Member, IEEE) received the B.S. and Ph.D. degrees in electrical and information engineering from Hunan University, Changsha, China, in 2010 and 2016, respectively.

From 2014 to 2015, he was a Visiting Scholar with the University of New Mexico, Albuquerque, NM, USA. From 2016 to 2018, he was a Post-Doctoral Fellow with the Department of Mechanical and Automation Engineering, The Chinese University of Hong Kong, Hong Kong. He is currently an Associate Professor with the College of Electrical and Information Engineering, Hunan University. His current research interests include multirobot systems, visual servoing, motion planning, and nonlinear control.

Jing Na (Member, IEEE) received the B.Eng. degree in automation and the Ph.D. degree in control science and engineering from the School of Automation, Beijing Institute of Technology, Beijing, China, in 2004 and 2010, respectively.

From 2011 to 2013, he was a Monaco/ITER Postdoctoral Fellow at ITER Organization, Saint-Paul-les, Durance, France. From 2015 to 2017, he was a Marie Curie Fellow with the Department of Mechanical Engineering, University of Bristol, Bristol, U.K. Since 2010, he has been with the Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming, China, where he became a Professor in 2013. He has coauthored one monograph and more than 100 international journal and conference papers. His research interests include intelligent control, adaptive parameter estimation, nonlinear control and applications for robotics, vehicle systems, and wave energy convertors.

Dr. Na was a recipient of the Best Application Paper Award of IFAC ICONS 2013 and the 2017 Hsue-shen Tsien Paper Award. He is currently an Associate Editor of the IEEE TRANSACTIONS ON INDUSTRIAL ELECTRONICS and *Neurocomputing*, and has served as the Organization Committee Chair of DDCLS 2019 and the International Program Committee Chair of ICMIC 2017.

Zhijia Zhao (Member, IEEE) received the B.Eng. degree from the North China University of Water Resources and Electric Power, Zhengzhou, China, in 2010, and the M.Eng. and Ph.D. degrees from the South China University of Technology, Guangzhou, China, in 2013 and 2017, respectively, all in automatic control.

He is currently a Lecturer of mechanical and electrical engineering with Guangzhou University, Guangzhou. His research interests include flexible mechanical systems, ocean cybernetics, and robotics.

Chenguang Yang (Senior Member, IEEE) received the Ph.D. degree in control engineering from the National University of Singapore, Singapore, in 2010, and Post-Doctoral Training in human robotics from Imperial College London, London, U.K.

His research interest lies in human–robot interaction and intelligent system design.

Dr. Yang has been awarded the EU Marie Curie International Incoming Fellowship, the U.K. EPSRC UKRI Innovation Fellowship, and the Best Paper Award of the IEEE TRANSACTIONS ON ROBOTICS as well as over ten international conference best paper awards. He is the Co-Chair of the Technical Committee on Bio-mechatronics and Bio-robotics Systems (B2S), IEEE Systems, Man, and Cybernetics Society, and the Co-Chair of the Technical Committee on Collaborative Automation for Flexible Manufacturing (CAFM), IEEE Robotics and Automation Society. He serves as an Associate Editor for a number of IEEE Transactions and other international leading journals.