Force Sensorless Admittance Control for Teleoperation of Uncertain Robot Manipulator Using Neural Networks

Chenguang Yang, *Senior Member, IEEE*, Guangzhu Peng, Long Cheng, *Senior Member, IEEE*, Jing Na, *Member, IEEE*, and Zhijun Li, *Senior Member, IEEE*

***Abstract*—In this paper, a force sensorless control scheme based on neural networks (NNs) is developed for interaction between robot manipulators and human arms in physical collision. In this scheme, the trajectory is generated by using geometry vector method with Kinect sensor. To comply with the external torque from the environment, this paper presents a sensorless admittance control approach in joint space based on an observer approach, which is used to estimate external torques applied by the operator. To deal with the tracking problem of the uncertain manipulator, an adaptive controller combined with the radial basis function NN (RBFNN) is designed. The RBFNN is used to compensate for uncertainties in the system. In order to achieve the prescribed tracking precision, an error transformation algorithm is integrated into the controller. The Lyapunov functions are used to analyze the stability of the control system. The experiments on the Baxter robot are carried out to demonstrate the effectiveness and correctness of the proposed control scheme.**

***Index Terms*—Admittance control, error transformation, force observer, Kinect, neural adaptive control, neural networks (NNs), robot.**

I. Introduction

IN THE last few decades, robots have become widely used in various fields, such as industry, service, and medical [1]–[5]. The robot can not only improve the quality of life but also can improve work efficiency and complete work that operators cannot finish under certain condition. However, traditional operating methods of robot usually need to use the external devices and softwares, and will bring inconvenience to the operator and reduce the production rates. An alternative method to make the robot interact with the human directly is letting robot learn human skills.

Traditional motion capture methods require operators to fix sensors on each joint of human body, but this will bring a lot of inconvenience [6]. In recent years, the vision-based motion capture scheme for motion recognition provides us another idea to achieve this goal [7], [8]. Because of its convenience and accuracy, this vision-based scheme has been widely adopted in robotics [9]. This control scheme uses a camera to capture human motion, which can avoid operators wearing a large number of wearable accessories. In this paper, the camera used for motion capture is Kinect (version 2.0) developed by Microsoft Company [10], [11]. Due to an RGB camera and depth sensor embedded in Kinect sensor, we can get three-dimensional (3-D) coordinates of each joint of human body. Based on this, we used a geometry vector-based method proposed in [12] to calculate each joint angle of human arm and generate a desired trajectory.

In practical teleoperation control system, robots may encounter external force from the environment. One approach to achieve compliant behavior is impedance control. The concept of impedance control in physical human robot interactions was introduced by Hogan [13]. Nowadays, this approach has become a classical control approach in robotics. The core idea of the impedance control methodology is to map generalized positions and velocities to generalized force. When controlling the impedance of a mechanism, we are controlling the force of resistance to external motions that are imposed by the environment. From a practical point of view, we usually view the behavior of the robot as the pose of the end-effector, which is defined in the Cartesian coordinates. Typically, the Cartesian position and velocity are the input of the controller and the motor torque is the output. Another approach is an admittance control, which is widely used in industrial robots. As shown in Fig. 1, admittance control is the inverse of impedance: it defines motions that result from a force input. An admittance control architecture is able to receive external force in each joint as inputs and generate the new motion. Therefore, force sensors which are applied to receive external force have been

Manuscript received July 12, 2018; revised March 24, 2019; accepted May 25, 2019. Date of publication June 28, 2019; date of current version April 15, 2021. This work was supported in part by the Engineering and Physical Sciences Research Council under Grant EP/S001913, in part by the National Natural Science Foundation of China under Grant 61873268 and Grant 61633016, and in part by the Beijing Municipal Natural Science Foundation under Grant L182060. This paper was recommended by Associate Editor S. Nahavandi. *(Corresponding author: Chenguang Yang.)*

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

G. Peng is with the Department of Computer and Information Science, Faculty of Science and Technology, University of Macau, Macau 999078, China (e-mail: gz.peng@qq.com).

L. Cheng is with the State Key Laboratory of Management and Control for Complex Systems, Institute of Automation, Chinese Academy of Sciences, Beijing 100190, China, and also with the School of Artificial Intelligence, University of Chinese Academy of Sciences, Beijing 100049, China (e-mail: long.cheng@ia.ac.cn).

J. Na is with the Faculty of Mechanical and Electrical Engineering, Kunming University of Science and Technology, Kunming 650500, China (e-mail: najing25@163.com).

Z. Li is with the Department of Automation, University of Science and Technology of China, Hefei 230026, China (e-mail: zjli@ieee.org).

Color versions of one or more figures in this article are available at https://doi.org/10.1109/TSMC.2019.2920870.

Digital Object Identifier 10.1109/TSMC.2019.2920870

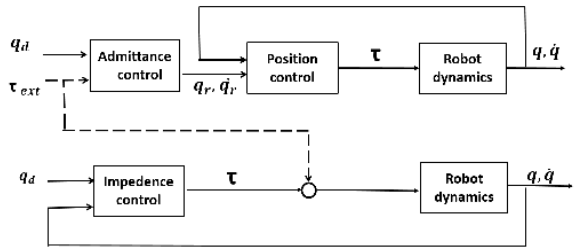


Fig. 1. Diagram of impedance and admittance control.

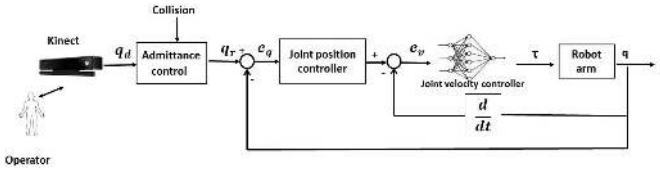


Fig. 2. Diagram of the control system.

widely used in admittance control systems. The general idea of measuring the external force is to fix force sensors on manipulators. However, these sensors added to the system are fragile and costly. For these reasons, the related techniques of estimating external force have received great attention and various schemes have been proposed. In [14], early estimation methods for robot application have been presented. In [15], disturbance observer approaches based on motor torques, joint angles, and velocities have been analyzed. In [16], a sensorless robot collision detection approach based on generalized momentum has been introduced.

Under an admittance control, with the measurement of external force, a desired trajectory will be modified. Then, a modified desired trajectory is obtained and tracked. In teleoperation control systems, tracking precision is of great importance for robotic manipulation. Model-free control and model-based control are the two main categories of controlling a robot manipulator. Compared with the model-free control methods, the model-based control methods usually have better control performance [17]. However, due to existence of uncertainties, it is hard for us to obtain an accurate dynamic model of a robot [18], [19]. How to deal with uncertainties has become a core issue in control design [20]. Generally, one of the most commonly used methods is adaptive control without prior information of system parameters. In [21], adaptation laws are designed to handle parametric uncertainties of the system.

In recent years, with the development of the neural networks (NNs) technology, adaptive control schemes with NNs have been widely employed in many systems [22]–[25]. In [26], NNs are integrated into control design to solve control problem in discrete-time systems with dead zone. In [27], an adaptive neural control is used to achieve a good result with unknown prior knowledge of system dynamics. In [28], a novel adaptive control scheme is presented for an autonomous helicopter and an NNs mechanism is employed into system to identify the unknown inertial matrix. NNs have a variety of models, one of the widely used network models is radial basis function NN (RBFNN), which has a good generalization ability and fast learning convergence speed. In [29], RBFNN is used to estimate unknown functions in WMR system. In [30] and [31], NN has been applied to handle the system uncertainties to get a desired result. In [32], RBFNN is to approximate unknown dynamics in the robot system. In [33], RBFNN is employed to approximate unknown functions in nonlinear systems. NNs are also used in other areas, such as image processing [34], function approximation [35], [36], and optimization [37]. The system uncertainties can also be estimated by other intelligent tools, such as fuzzy logic system, etc. [38], [39]. In practice, the rigorous precision requires that both the transient and steady performance should be taken into account. However, most general adaptive control methods can only guarantee the steady performance, while difficult to solve the transient problem [40]. For this purpose, we use error transformation technique proposed in [41] and [42] to govern the tracking errors into a desired level.

The contributions of this paper are presented as follows.

1) Combination of the admittance control and the force observer shows an effective way to make the robot have a compliant behavior subject to the external force.
2) Kinect sensor is used to generate trajectory to teleoperate the robot. The error transformation technique and NN are used in teleoperation system so that both transient and stable tracking performance are guaranteed.
3) Analysis of signals in the admittance control system are given to prove that all signals are bounded.

The rest of this paper is structured as follows. After giving the preliminaries of the system in Section II. Section III gives the design and analysis of the control design. The experimental results are given in Section IV, before a conclusion is drawn in Section V.

II. Preliminaries

A. System Configuration

The teleoperation control system is shown in Fig. 2. Using the Kinect sensor, a desired trajectory will be generated.

Kinect V2 is a human–machine interaction device launched by Microsoft. It contains an RGB camera and depth sensor, which are based on IR emitters. The RGB camera is used to shoot color images within the scope of view and the depth sensor can obtain and analyze spectra and create depth images of the human body.

Without external torque from the environment, the robot will follow the trajectory of the operator. If external torque exists, the desired trajectory will be modified. The robot will track the modified trajectory affected by the external torque from the environment.

B. Human Arm Geometry Vector Approach

Most geometry approaches are based on locations of the movements. With Kinect sensor, each joint of human body is represented by 3-D point in the Kinect coordinate frame, which follows the right-hand rule, as shown in Fig. 4. The Kinect sensor is regarded as the origin of the coordinate

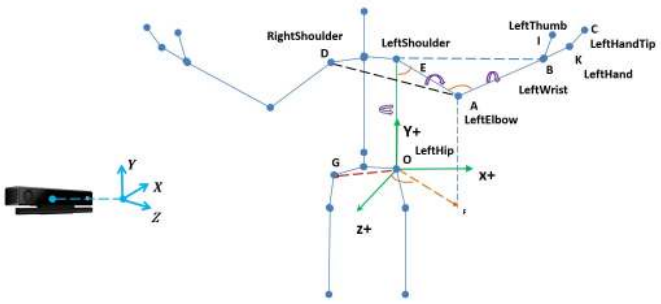


Fig. 3. Geometry model of human left arm.

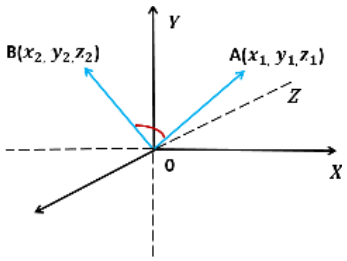


Fig. 4. Points representation in the Kinect coordinate frame modified from [50].

frame and the z-axis is consistent with the direction of Kinect induction. The geometry model of human left arm is built in Fig. 3.

Since the vector method is not applicable to the Kinect coordinate frame, we should map the coordinate frame of Kinect to the mathematical coordinate frame.

From the skeleton data, we can transform two different points into a vector, which is in the mathematical coordinate frame. The transformation can be provided as

$$\overrightarrow{AB} = (x_2 - x_1,\ y_2 - y_1,\ z_2 - z_1)^T \tag{1}$$

where $A(x_1,\ y_1,\ z_1)$ and $B(x_2,\ y_2,\ z_2)$ are the two different points in the coordinate frame of Kinect.

After the vector of the mathematical coordinate frame is obtained, based on the geometry vector approach [12], we can calculate desired angle values using the vector angle formula

$$\cos\left\langle \overrightarrow{V_1}, \overrightarrow{V_2} \right\rangle = \frac{\overrightarrow{V_1} \cdot \overrightarrow{V_2}}{|\overrightarrow{V_1}| \cdot |\overrightarrow{V_2}|}. \tag{2}$$

1) Calculation of Shoulder Angle: As shown in Fig. 3, the shoulder yaw ($\angle$DEA) can be obtained by calculating angle between plane OEA and OED. The shoulder roll is the angle of plane OEA and EAB.

The shoulder pitch angle ($\angle$OEA) is the angle between vector EO and EA which can be calculated by passing EO and EA into (2).

2) Calculation of Elbow Angle: There are two angles related with the elbow joint. Elbow pitch ($\angle$EAB) and elbow roll is the angle between plane EAB and ABI.

3) Calculation of Wrist Angle: Now, we are coming to solve angle of wrist. The wrist yaw angle is the angle between lower arm and hand plane. The angle of wrist pitch can be viewed as the angle between vector $\overrightarrow{X_5}$ and $\overrightarrow{Y_7}$, we can calculate it by employing following equations:

$$\overrightarrow{Z_7} = \overrightarrow{Y_7} \times \overrightarrow{X_7},\ \overrightarrow{X_7} = \frac{\overrightarrow{BK}}{|\overrightarrow{BK}|}, \overrightarrow{Z_7} = \frac{\overrightarrow{BK} \times \overrightarrow{BI}}{|\overrightarrow{BK}| \times |\overrightarrow{BI}|} \tag{3a}$$

$$\overrightarrow{X_5} = k_1 \cdot \overrightarrow{BI} + k_2 \cdot \overrightarrow{BK} \tag{3b}$$

$$\left(k_1 \cdot \overrightarrow{BI} + k_2 \cdot \overrightarrow{BK} \cdot\right)\overrightarrow{AB} = 0 \tag{3c}$$

$$\left|k_1 \cdot \overrightarrow{BI} + k_2 \cdot \overrightarrow{BK}\right| = 1. \tag{3d}$$

Until now, we get all seven joint angles. They are shoulder yaw, shoulder pitch, shoulder roll, elbow pitch, elbow roll, wrist yaw, and wrist pitch, which can be defined as $q_{d1}, q_{d2}, q_{d3}, q_{d4}, q_{d5}, q_{d6}$, and q_{d7}.

C. External Torque Estimation: Observer Approach

In this section, the way to estimate an external torque in joint space is using a force observer based on the generalized momentum approach. Compared with the traditional methods requiring computation of joint accelerations or the inversion of the inertia matrix [43], this observer avoid reduce the computing burden and noise with the acceleration of joint angle.

The system dynamics can be described by

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + \tau_{\text{ext}} = \tau \tag{4}$$

where $q \in R^n$ and $\dot{q} \in R^n$ denote the joint angle and velocity vector, $C \in R^{n\times n}$, $M \in R^{n\times n}$, and $G \in R^n$ are the systematic dynamics, representing Coriolis matrix, inertia matrix, and gravity load, respectively. $\tau_{\text{ext}} \in R^n$ is the external torque on joints, and τ is the joint torque on robotic arms. In [43], the generalized momentum is expressed as

$$p = M(q)\dot{q}. \tag{5}$$

Its time derivative form

$$\dot{p} = \dot{M}\dot{q} + M\ddot{q}. \tag{6}$$

Substituting (6) into (4), we have

$$\dot{p} = \dot{M}(q, \dot{q})\dot{q} + \tau - C(q, \dot{q})\dot{q} - G(q) - \tau_{\text{ext}}. \tag{7}$$

Then, the inertia matrix M and can be written as [44]

$$\dot{M} = C + C^T. \tag{8}$$

Substituting (8) into (7) results in

$$\dot{p} = C^T(q, \dot{q})\dot{q} + \tau - G(q) - \tau_{\text{ext}}. \tag{9}$$

The advantage of this method is that (9) based on the generalized momentum does not involve joint angle accelerations $\ddot{q}$. In the end, the external torque can be modeled as

$$\dot{\tau}_{\text{ext}} = A_\tau \tau_{\text{ext}} + w_\tau \tag{10}$$

where w_τ is the uncertainty, $w_\tau \sim N(0, Q_\tau)$. Usually, A_τ is defined as $A_\tau = 0_{n\times n}$. However, a negative diagonal matrix can reduce the offset of the estimation of disturbances. Then, (9) can be reformulated as

$$\dot{p} = u - \tau_{\text{ext}} \tag{11}$$

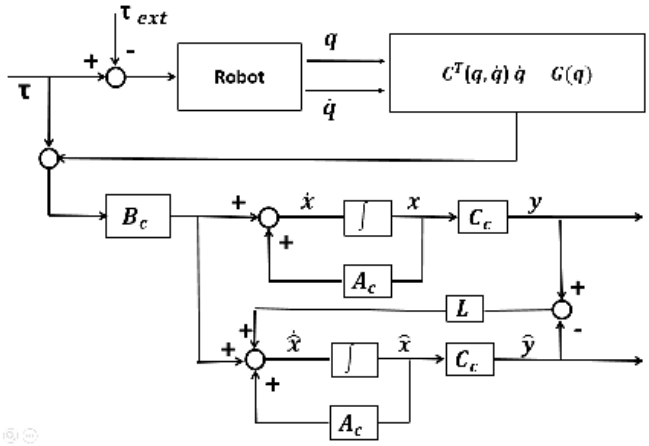


Fig. 5. Overview of force observer based on generalized momentum approach.

where

$$u = \tau + C^T(q, \dot{q})\dot{q} - G(q). \tag{12}$$

The above equations can be combined and reformulated in the state-space form

$$\underbrace{\begin{bmatrix} \dot{p} \\ \dot{\tau}_{\text{ext}} \end{bmatrix}}_{\dot{x}} = \underbrace{\begin{bmatrix} 0_n & -I_n \\ 0_n & A_\tau \end{bmatrix}}_{A_c} \underbrace{\begin{bmatrix} p \\ \tau_{\text{ext}} \end{bmatrix}}_{x} + \underbrace{\begin{bmatrix} I_n \\ 0_n \end{bmatrix}}_{B_c} u + \underbrace{\begin{bmatrix} 0 \\ w_\tau \end{bmatrix}}_{w}$$
$$y = \underbrace{\begin{bmatrix} I_n & 0_n \end{bmatrix}}_{C_c} \begin{bmatrix} p \\ \tau_{\text{ext}} \end{bmatrix} + v \tag{13}$$

where v is the measurement noise $v \sim N(0, R_c)$. It can be easily proved that this system is observable. Since q and $\dot{q}$ are able to be measured, the generalized momentum defined in (5) can be regarded as a measurement. Then, a state observer is designed

$$\begin{cases} \dot{\hat{x}} = A_c\hat{x} + B_c u + L\big(y - \hat{y}\big) \\ \hat{y} = C_c\hat{x}. \end{cases} \tag{14}$$

Solving the L is to design a gain matrix for the system, and L can be calculated as

$$L = PC_c{}^T R_c^{-1} \tag{15}$$

where the matrix P can be calculated by the algebraic Riccati equation (ARE) [45]

$$A_c P + PA_c{}^T - PC_c{}^T R_c^{-1} C_c P + Q_c = 0 \tag{16}$$

where Q_c is the uncertainty of the state, written as

$$Q_c = \text{diag}([0, Q_\tau]). \tag{17}$$

A schematic overview of the force observer is shown in Fig. 5. As shown in Fig. 5, the output $y = C_c x(t)$ is compared with $C_c\hat{x}(t)$. Their difference, passing through the gain matrix L, is used as a correcting term. If the gain matrix L is properly designed, the difference will drive the estimated state to actual state. From the above analysis, we can see that the estimation of states can be obtained from observer, which can be written as

$$\hat{\tau}_{\text{ext}} = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}\hat{x}$$
$$\hat{p} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \end{bmatrix}\hat{x}. \tag{18}$$

D. Admittance Control

In this section, an admittance control method using the estimated external torque is presented. We assume that the manipulator will modify its desired trajectory when the external torque is imposed on the robot. In this case, we use an admittance control to receive the external torque. Based on the measurements of the external torque $\hat{\tau}_{\text{ext}}$ and the initial desired trajectory q_d obtained from the Kinect, a modified trajectory q_r is generated. Therefore, the controller has the causality of mapping $\hat{\tau}_{\text{ext}}$ to q_r. Generally, an admittance model can be described as

$$\hat{\tau}_{\text{ext}} = f(q_r, q_d) \tag{19}$$

where $q_d \in R^7$ is the vector of joint angles obtained from Kinect and $q_r \in R^7$ is the vector of joint angles affected by external torque; and $f(\cdot)$ is the mapping function. A simple admittance model is $K_d(q_r - q_d) = \hat{\tau}_{\text{ext}}$, where K_d is a positive constant.

E. RBFNN

RBFNN is an artificial NN and has been widely used as function approximators in control engineering. It is proved that any smooth function can be approximated by the RBFNN within a compact set Ω [46]. It can be expressed as follows:

$$\phi(Z_{\text{NN}}) = W^T S(Z_{\text{NN}}) + \varepsilon(Z_{\text{NN}}) \tag{20}$$

where $Z_{\text{NN}} \in \Omega \subset R^m$ is the input vector, W is the weight matrix, and l represents the number of neurons. $S(Z_{\text{NN}}) = [s_1(Z_{\text{NN}}), s_2(Z_{\text{NN}}), \ldots, s_1(Z_{\text{NN}})]^T$ is the basis function of RBFNN, and $s_i(Z_{\text{NN}})$ is commonly chosen as the Gaussian function with

$$s_i(Z_{\text{NN}}) = \exp\left[\frac{-(Z_{\text{NN}} - u_i)^T(Z_{\text{NN}} - u_i)}{\sigma_i^2}\right], \quad i = 1, \ldots, l \tag{21}$$

where u_i is a center of the node and σ_i denotes the variance.

If the number of neurons l is sufficiently large, there is a weight matrix W^* and an approximation error $\varepsilon^*(Z_{\text{NN}})$

$$\phi(Z_{\text{NN}}) = W^{*T} S(Z_{\text{NN}}) + \varepsilon^*(Z_{\text{NN}}). \tag{22}$$

If the center of the node is chosen appropriately, the approximation error $\varepsilon^*(Z_{\text{NN}})$ is bounded and could be minimized

$$W^* = \arg\min_{W \subset R^m} \left\{\sup\left|\phi(Z_{\text{NN}}) - W^T S(Z_{\text{NN}})\right|\right\}. \tag{23}$$

The ideal weight matrix W is unknown. In the practical system, the weight matrix W is replaced by the estimation $\hat{W}$. Thus, (20) can be described

$$\hat{\phi}(Z_{\text{NN}}) = \hat{W}^T S(Z_{\text{NN}}) + \varepsilon(Z_{\text{NN}}). \tag{24}$$

The weight estimation errors are $\tilde{W} = W^* - \hat{W}$.

III. Controller Design

The controller is designed to make the robot can follow the desired trajectory in the joint space generated by the Kinect, as shown in Fig. 2. The NN is used to estimate uncertainties of the model and ensure the steady state of the system.

A. Error Transformation

We define tracking errors of the manipulator

$$\begin{aligned} e_q &= q - q_d \\ e_v &= \dot{q} - v^d \end{aligned} \tag{25}$$

where v^d will be defined later. The objective is to make the actual joint trajectory q track the desired joint trajectory q_d effectively. At first, we define a smooth and bounded performance function

$$\rho(t) = (\rho_0 - \rho_\infty)e^{-pt} + \rho_\infty \tag{26}$$

where the parameters of ρ_0, ρ_∞, and p are the positive constants. To guarantee that the tracking error can meet the transient performance, we introduce the following transformation functions:

$$e_{qi}(t) = \rho(t)R_i\left(P_i\left(\frac{e_{qi}(t)}{\rho(t)}\right)\right) \tag{27a}$$

$$R_i(t) = \begin{cases} \frac{\exp(t)-\sigma}{1+\exp(t)}, & \text{if } e_{qi}(0) > 0 \\ \frac{\sigma\exp(t)-1}{1+\exp(t)}, & \text{if } e_{qi}(0) < 0. \end{cases} \tag{27b}$$

$R_i(\cdot)$ is the inverse function of $P_i(\cdot)$

$$P_i(t) = \begin{cases} \ln\frac{t+\sigma}{1-t}, & \text{if } e_{qi}(0) > 0 \\ \ln\frac{t+1}{\sigma-t}, & \text{if } e_{qi}(0) < 0 \end{cases} \tag{28}$$

where σ is a positive constant. According to the function $R_i(\cdot)$, if the $\eta_i(t)$ is bounded, the bounds of the tracking error $e_q(t)$ can be defined: $-\sigma\rho(t) < e_q(t) < \rho(t)$ with $e_q(t) > 0$ and $-\rho(t) < e_q(t) < \sigma\rho(t)$ with $e_q(t) < 0$. Therefore, the overshoot Δ in transient phase is bounded by

$$\begin{aligned} -\sigma\rho(0) < \Delta < \rho(0) \text{ if } e_q(0) > 0 \\ -\rho(0) < \Delta < \sigma\rho(0) \text{ if } e_q(0) < 0 \end{aligned} \tag{29}$$

and the amplitude of tracking errors in stable state will be within in $\max[\rho_\infty, \sigma\rho_\infty]$ and the maximum overshoot and undershoot of transient performance are bounded in $[\sigma\rho_{0i}, -\sigma\rho_{0i}]$. Usually, the settling time is the shortest time that the system achieve and maintain the steady state error within the $100\% \pm 5\%$ range, then the settling time is less than $(\max(1,\sigma)/p)\ln(\rho_0 - \rho_\infty/1.05\rho_\infty)$. Therefore, we can control the transient and stable state of the system by setting proper parameters. From (27a), we define

$$\eta_i(t) = P_i\left(\frac{e_{qi}(t)}{\rho(t)}\right). \tag{30}$$

Then, the desired joint velocity $v_i^d(t)$ is designed as

$$v_i^d(t) = -k_1\rho(t)\eta_i(t) + \dot{q}_i^d(t) + \frac{\dot{\rho}(t)}{\rho(t)}e_{qi}(t) \tag{31}$$

where k_1 is a positive constant.

We define a Lyapunov function $V_1 = (1/2)\eta^T(t)\eta(t)$. Its differential form is

$$\dot{V}_1 = \frac{\eta^T(t)\dot{P}(\eta(t))e_v(t)}{\rho(t)} - k_1\eta^T(t)\dot{P}(\eta(t))\eta_i(t) \tag{32}$$

where

$$\dot{P}(\eta(t)) = \text{diag}\left(\dot{P}_1(R_1(\eta_1(t))), \ldots, \dot{P}_n(R_n(\eta_n(t)))\right) \tag{33a}$$

$$v^d = \left[v_1^d, v_2^d, \ldots, v_n^d\right]. \tag{33b}$$

B. Neural Network and Joint Velocity Control

The goal of joint velocity control is to make the velocity error $e_v(t)$ as small as possible. Substituting the differentiation of $\eta_i(t)$ into (4), we can obtain that

$$M(q)\dot{e}_v + C(q,\dot{q})e_v + G'(q) = \tau + M(q)\dot{v}^d + C(q,\dot{q})v^d \tag{34}$$

where $G'(q) = G(q) + ([\dot{P}(\eta(t))\eta(t)]/\rho(t))$.

Design the control torque

$$\tau = -k_2 e_v - \hat{M}(q)\dot{v}^d - \hat{C}(q,\dot{q})v^d - \hat{G}'(q) + \hat{\tau}_{\text{ext}}. \tag{35}$$

Applying the NN approximation technique, we have

$$\begin{aligned} M(q) &= W_M^{*T}S_M(q) + \varepsilon_M \\ C(q,\ \dot{q}) &= W_C^{*T}S_C(q,\dot{q}) + \varepsilon_C \\ G(q) &= W_G^{*T}S_G(q) + \varepsilon_G \end{aligned} \tag{36}$$

where W_M^{*T}, W_C^{*T}, and W_G^{*T} are the ideal weight matrix. The estimation of $M(q)$, $C(q,\dot{q})$, and $G(q)$ are based on RBFNN can be written as

$$\begin{aligned} \hat{M}(q) &= \hat{W}_M^T S_M(q) \\ \hat{C}(q,\ \dot{q}) &= \hat{W}_C^T S_C(q,\dot{q}) \\ \hat{G}(q) &= \hat{W}_G^T S_G(q). \end{aligned} \tag{37}$$

Then, the dynamics can be rewritten as

$$\begin{aligned} &M(q)\dot{e}_v + C(q,\dot{q})e_v + k_2 e_v + \frac{\dot{P}(\eta(t))\eta(t)}{\rho(t)} \\ &\quad = (M(q) - \hat{M}(q))\dot{v}^d + \left(C(q,\dot{q}) - \hat{C}(q,\dot{q})\right)v^d \\ &\qquad + \left(G(q) - \hat{G}(q)\right) + \left(\hat{\tau}_{\text{ext}} - \tau_{\text{ext}}\right) \end{aligned} \tag{38}$$

where $\hat{M}(q)$, $\hat{C}(q,\dot{q})$, and $\hat{G}(q)$ are the estimation matrix. The right-hand side of the equation can be expressed as $\tilde{W}_{(\cdot)}S_{(\cdot)}$, where $\tilde{W}_{(\cdot)} = W_{(\cdot)} - \hat{W}_{(\cdot)}$.

Considering the Lyapunov function $V_2 = (1/2)(e_v)^T(t)M(q)e_v$, its differential form with respect to time is

$$\begin{aligned} \dot{V}_2 &= (1/2)(e_v)^T\dot{M}(q)e_v + (e_v)^T M(q)\dot{e}_v \\ &= -k_2\|e_v\|_2^2 - (e_v)^T e_f + (e_v)^T\frac{\dot{P}(\eta(t))\eta(t)}{\rho(t)} \\ &\quad + (e_v)^T\tilde{W}_M^T S_M\dot{v}^d + (e_v)^T\tilde{W}_C^T S_C v^d + (e_v)^T\tilde{W}_G^T S_G \end{aligned} \tag{39}$$

where $e_f = (\tau_{\text{ext}} - \hat{\tau}_{\text{ext}})$.

The updating law of the weight matrix $\hat{W}$ is

$$\begin{aligned}\dot{\hat{W}}_M &= \Theta_M\left(S_M\dot{v}^d(e_v)^T - \gamma_M\hat{W}_M\right)\\ \dot{\hat{W}}_C &= \Theta_C\left(S_Cv^d(e_v)^T - \gamma_C\hat{W}_C\right)\\ \dot{\hat{W}}_G &= \Theta_G\left(S_G(e_v)^T - \gamma_G\hat{W}_G\right)\end{aligned} \tag{40}$$

where Θ and γ are the positive constant specified by the designer.

C. Stability Analysis

Let us construct the overall Lyapunov function

$$\begin{aligned}V = V_1 + V_2 &+ \frac{1}{2}\mathrm{tr}\left(\tilde{W}_M^T\Theta_M^{-1}\tilde{W}_M\right) + \frac{1}{2}\mathrm{tr}\left(\tilde{W}_C^T\Theta_C^{-1}\tilde{W}_C\right)\\ &+ \frac{1}{2}\mathrm{tr}\left(\tilde{W}_G^T\Theta_G^{-1}\tilde{W}_G\right).\end{aligned} \tag{41}$$

The derivative of V is calculated by

$$\begin{aligned}\dot{V} &= -k_1\eta^T(t)\dot{P}(\eta(t))\eta(t) - k_2\|e_v\|_2^2 - (e_v)^Te_f\\ &\quad + (e_v)^T\tilde{W}_M^TS_M + (e_v)^T\tilde{W}_C^TS_C + (e_v)^T\tilde{W}_G^TS_G\\ &\quad - \mathrm{tr}\left(\tilde{W}_M^T\Theta_M^{-1}\dot{\hat{W}}_M\right) - \mathrm{tr}\left(\tilde{W}_C^T\Theta_C^{-1}\dot{\hat{W}}_C\right) - \mathrm{tr}\left(\tilde{W}_G^T\Theta_G^{-1}\dot{\hat{W}}_G\right)\\ &\le -k_1\eta^T(t)\dot{P}(\eta(t))\eta(t) - k_2\|e_v\|_2^2 - (e_v)^Te_f\\ &\quad - \gamma_M\mathrm{tr}\left(\tilde{W}_M^T\hat{W}_M\right) - \gamma_C\mathrm{tr}\left(\tilde{W}_C^T\hat{W}_C\right) - \gamma_G\mathrm{tr}\left(\tilde{W}_G^T\hat{W}_G\right).\end{aligned} \tag{42}$$

According to the definition of function $\dot{P}(\eta(t))$, we can obtain $\eta^T(t)\dot{P}(\eta(t))\eta(t) \ge 2/(1+\sigma)\|\eta(t)\|^2$. Considering the Young's inequality [47]

$$\begin{aligned}\tilde{W}^T\left(W^* - \tilde{W}\right) &\le -\frac{1}{2}\|\tilde{W}\|^2 + \frac{1}{2}\|W^*\|^2\\ -(e_v)^Te_f &\le \frac{1}{2}\|e_v\|^2 + \frac{1}{2}\|e_f\|^2.\end{aligned} \tag{43}$$

Then, (42) can be derived

$$\begin{aligned}\dot{V} &\le -2k_1/(1+\sigma)\|\eta(t)\|^2 - \left(k_2 - \frac{1}{2}\right)\|e_v\|^2 + \frac{1}{2}\|e_f\|^2\\ &\quad - \frac{1}{2}\gamma\mathrm{tr}\left(W_M^{*T}W_M^{*T} + W_C^{*T}W_C^{*T} + W_G^{*T}W_G^{*T}\right)\\ &\quad - \frac{1}{2}\gamma\mathrm{tr}\left(\tilde{W}_M^T\tilde{W}_M + \tilde{W}_C^T\tilde{W}_C + \tilde{W}_G^T\tilde{W}_G\right).\end{aligned} \tag{44}$$

For $k_2 > (1/2)$, if the inequality satisfies the following requirements:

$$\begin{aligned}\kappa &\le -2k_1/(1+\sigma)\|\eta(t)\|^2 - \left(k_2 - \frac{1}{2}\right)\|e_v\|^2\\ &\quad + \frac{1}{2}\gamma\mathrm{tr}\left(\tilde{W}_M^T\tilde{W}_M + \tilde{W}_C^T\tilde{W}_C + \tilde{W}_G^T\tilde{W}_G\right)\end{aligned} \tag{45}$$

where $\kappa = (1/2)\|e_f\|^2 - (1/2)\gamma\mathrm{tr}(W_{(\cdot)}^{*T}W_{(\cdot)}^*)$. Then, we have $\dot{V} \le 0$.

We define the state variable ξ composed of $\eta(t)$, e_v, and $\tilde{W}_{(\cdot)}$, and it can be expressed as

$$\dot{V}(\xi) < 0 \quad \forall\|\xi\| > \varrho \tag{46}$$

where ϱ is a positive constant. Conversely, $\dot{V}(\xi) > 0 \ \forall\|\xi\| > \varrho$.

Let us choose $0 < V(\xi) < \beta < c$, where β and c are the positive constants. Define that $\Omega_b = \{V(\xi) \le \beta\}$ and $\Omega_c = \{V(\xi) \le c\}$, and we have

$$\Xi = \{\beta \le V(\xi) \le c\} = \Omega_c - \Omega_b. \tag{47}$$

We see that $V(\xi)$ with respect to time is negative over Ξ, that is

$$\dot{V}(\xi) < 0 \quad \forall\xi \in \Xi. \tag{48}$$

In other words, the state variable $\xi(t)$ that outside the set Ω_b will enter into Ω_b within a period of time, and cannot escape Ω_b because $\dot{V}$ is negative on/outside its boundary $V(\xi) = \beta$, as shown in Fig. 6.

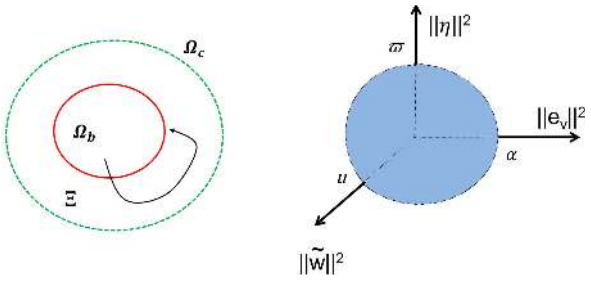


Fig. 6. Trajectory of ξ and the set Ω_b defined in (49).

Theorem 1: Using the uniformly ultimately bounded (UUB), errors $\eta(t)$, e_v and $\tilde{W}_{(\cdot)}$ will fall into the set Ω_b, which is defined as

$$\begin{aligned}\Omega_b = \Bigg\{ &\left(\|\tilde{W}_M\|, \|\tilde{W}_C\|, \|\tilde{W}_G\|, \|e_v\|, \|\eta(t)\|\right), \Bigg|\\ &\frac{\gamma_M\|\tilde{W}_M\|^2}{2\kappa} + \frac{\gamma_C\|\tilde{W}_C\|^2}{2\kappa} + \frac{\gamma_G\|\tilde{W}_G\|^2}{2\kappa}\\ &+ \frac{\left(k_2 - \frac{1}{2}\right)}{\kappa}\|e_v\|^2 + \frac{2k_1}{(1+\sigma)\kappa}\|\eta(t)\|^2 \le 1\Bigg\}.\end{aligned} \tag{49}$$

As shown in Fig. 6, points on each axis of the coordinate are defined as

$$\begin{aligned}\frac{2k_1}{(1+\sigma)}\|\eta(t)\|^2 = \kappa,\ \eta(t) = \varpi\\ \left(k_2 - \frac{1}{2}\right)\|e_v\|^2 = \kappa,\ e_v = \alpha\\ \gamma\left\|\tilde{W}_{(\cdot)}\right\|^2 = 2\kappa\ \tilde{W}_{(\cdot)} = u.\end{aligned} \tag{50}$$

From the above analysis, we can conclude that the $\|\eta(t)\|$, $\|W\|_F$, and e_v are bounded. According to (25) and (26a), we can obtain the tracking errors e_q can be bounded, which can guarantee the transient performance. Then, the $q = e_q + q_d$ is bounded.

IV. Experiment Studies

In this section, experiments studies are given to demonstrate the effectiveness and correctness of the proposed control scheme. The experiment is based on the Baxter Research robot by Rethink Robotics, as shown in Fig. 7. The Baxter robot is a two-armed robot with 7 degrees of freedom $(s_0, s_1, e_0, e_1, w_0, w_1, w_2)$. Each joint is driven by a series elastic actuator (SEA), which enable the robot have human-like

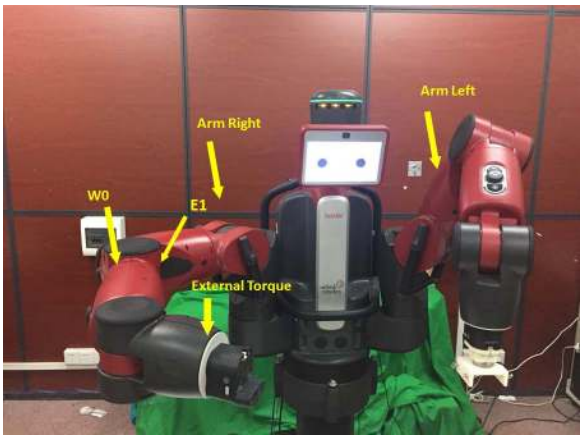


Fig. 7. Experimental description.

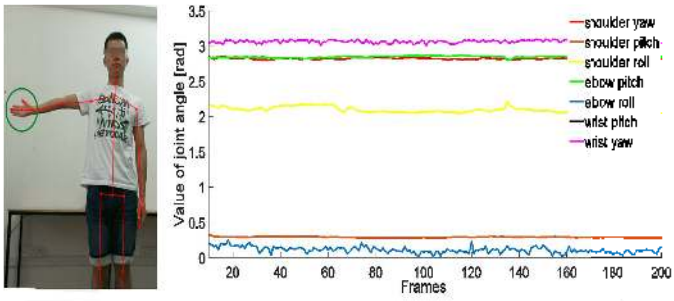


Fig. 8. Operator is in a static position and keep his right arm in the horizontal state. The right figure presents the variation of joint angle in successive frames.

behaviors. The robot is controlled and linked to a computer and runs on the robot operating system (ROS).

In the experiment, the robot is interacting with the environment and the external torque is applied at the end-effector. For the right arm of the robot, we initialized it in horizontal posture. Considering simplicity and generality, we use two joints (e_1, w_0) and positions of other joints are locked in the experiment. The desired trajectory generated by using the Kinect sensor is the input signal of the control system and will be modified by the external torque.

A. Test of Geometry Vector Performance

Two kinds of experiments are primarily implemented to test the performance of kinematics geometry vector-based approach. In the course of the experiments, only the operator stands in front of the Kinect about 3 m. The first experiment needs the operator to keep in a static position in front of the Kinect sensor and his left arm in the horizontal state. The experimental result is shown in Fig. 8, it is clear that the variation of the seven joint angles are smooth and accurate. Although there are some small fluctuations, the fluctuations are so small that can be ignored. The reason for fluctuations is that the points of the joints detected by the Kinect are not absolutely stable and the operator cannot ensure that the arm is completely stationary in the course of the experiment.

In the second experiment, the operator is in a dynamic action and reciprocate rotation of his elbow from the origin position to final position with a low speed. Using the same method, the data is sent to MATLAB for processing.

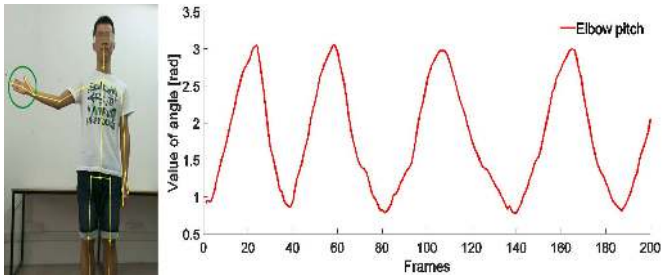


Fig. 9. Operator rotate his arm circularly with a low speed from origin position to final position.

As shown in Fig. 9, it is obvious to find that the variation of elbow pitch angle is periodic and regular in line with the movement of the arm. Due to the jitter in the process of elbow movement, there will be some tiny fluctuations in the curve. The overall trend of angle is correct and satisfactory. According to the above two kinds of experimental results, the overall performance is consistent with our expectation and satisfactory, which verify the correctness of our proposed method.

B. Test of Neural-Learning Tracking Performance

This set of experiments are mainly to demonstrate the effectiveness of the neural adaptive controller. The desired trajectory is obtained from the Kinect sensor. The desired trajectories are elbow pitch joint and wrist roll joint, respectively, where $t \in [0, t_s]$ and $t_s = 20$ s, as shown in Fig. 10. The initial values of joint angle are set to be: $q_1 = 0$ rad and $q_2 = 0.1$ rad, and the initial values of joint velocity are set to be: $\dot{q}_1 = 0$ rad/s and $\dot{q}_2 = 0$ rad/s. To guarantee the transient performance, the parameters of the performance functions are set to be: $\rho_0 = 0.2$ and $\rho_\infty = 0.03$; and $\sigma = 5$. Therefore, the error is bounded in $[-\sigma\rho(t),\ \sigma\rho(t)]$. The control gains are selected as $k_1 = [12, 1]$ and $k_2 = [15, 1]$. The initial weight matrices are: $\hat{W}_M^T(0) = 0 \in R^{nl\times n}$, $\hat{W}_C^T(0) = 0 \in R^{2nl\times n}$, and $\hat{W}_{G'}^T = 0 \in R^{nl\times n}$.

Comparative experiments are carried to test the tracking performance with three different methods. The experimental results are shown in Figs. 10 and 11. As shown in Figs. 10(a) and (b) and 11(a) and (b), the actual trajectory can follow the desired trajectory well and the tracking errors can converge to the prescribed bounded defined in (26) in both transient and stable phase. From Fig. 11(a) and (b), we can see that there is no overshot of each joint under the proposed controller. Fig. 12 shows the convergence of NN weight norm of each joint and the control inputs presented in Fig. 13 are bounded. For the purpose of comparison, we carry out two different controller proposed in [48] and [49], respectively. Fig. 10(c)–(f) presents the tracking performance and the tracking errors are in Fig. 11(c)–(f). From Fig. 11(c) and (d), under the controller in [48], the tracking errors violate the prescribed bounds and errors in stable phase are relatively larger than the proposed controller. From Fig. 11(e) and (f), under the controller in [49], we can observe that without transient constraint control, the values of overshot are about 8.7% and 10.9% and the values of settling time are 2.2 and 1.47 s for each joint, respectively.

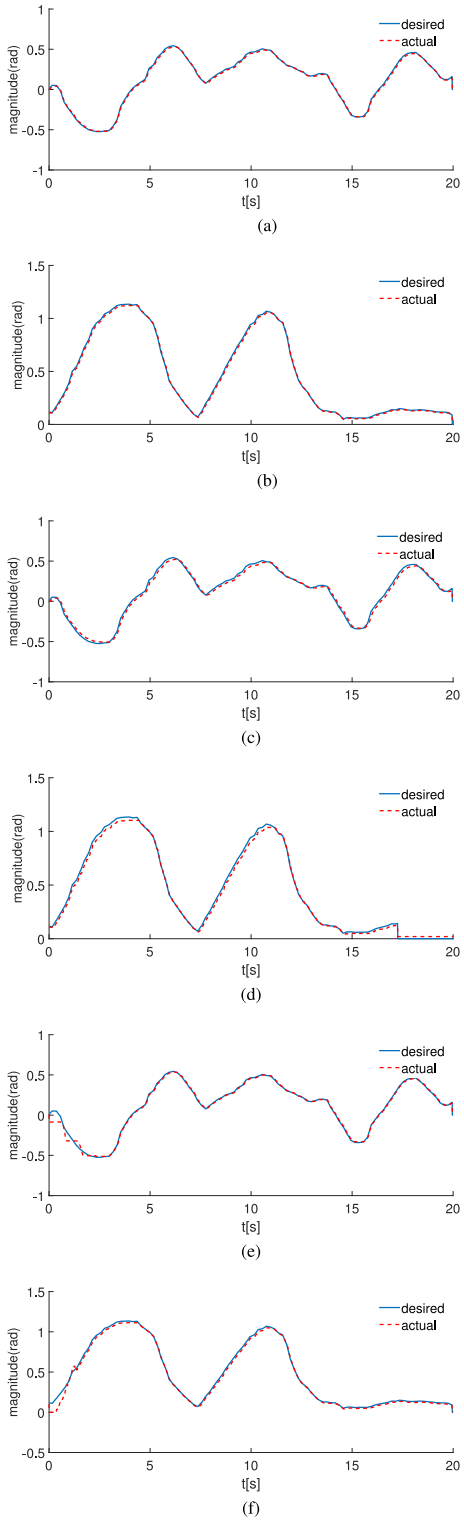


Fig. 10. Results of tracking performance of joints with three different methods. (a) Tracking performance of joint 1 with proposed controller. (b) Tracking performance of joint 2 with proposed controller. (c) Tracking performance of joint 1 with under controller in [48]. (d) Tracking performance of joint 2 with under controller in [48]. (e) Tracking performance of joint 1 with under controller in [49]. (f) Tracking performance of joint 2 with under controller in [49].

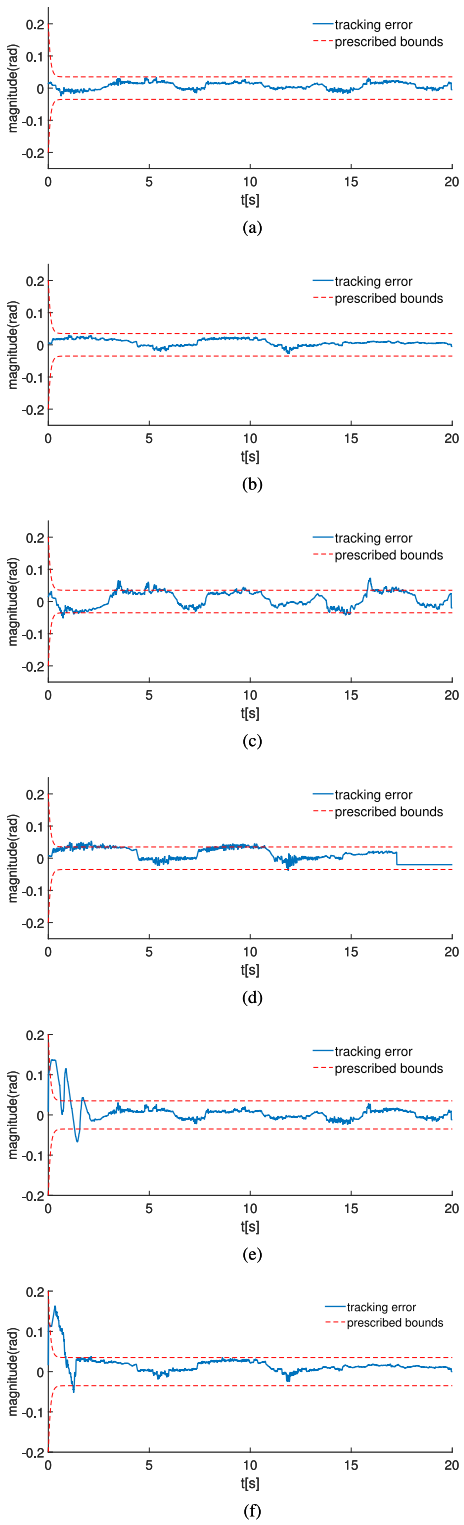


Fig. 11. Results of tracking errors of joints with three different methods. (a) Tracking error of joint 1 with proposed controller. (b) Tracking error of joint 2 with proposed controller. (c) Tracking error of joint 1 with under controller in [48]. (d) Tracking error of joint 2 with under controller in [48]. (e) Tracking error of joint 1 with under controller in [49]. (f) Tracking error of joint 2 with under controller in [49].

The experimental results show that our proposed controller can guarantee the tracking errors never violate the prescribed bounds in both transient and stable stage.

C. Test of Admittance Control Performance

The last experiment is mainly about the test of performance of an admittance control. In the experiment, the external torque

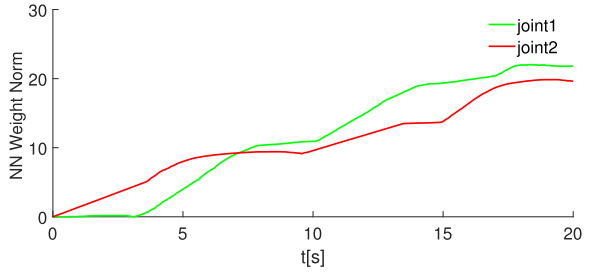
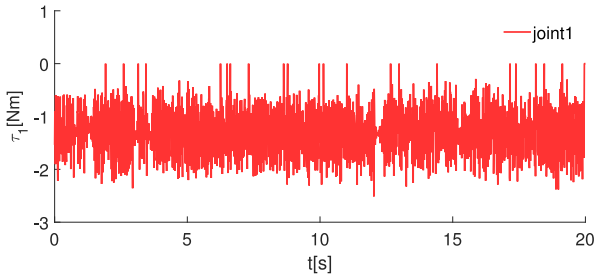
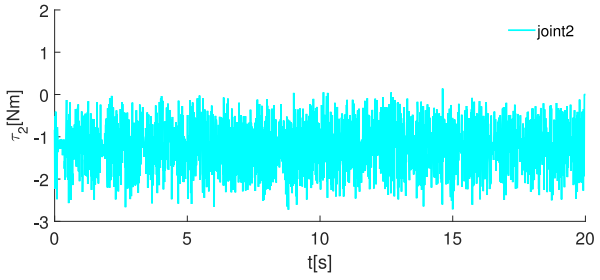


Fig. 12. NN weight norm of each joint.



(a)



(b)

Fig. 13. Control inputs of each joint. (a) Control input of joint 1. (b) Control input of joint 2.

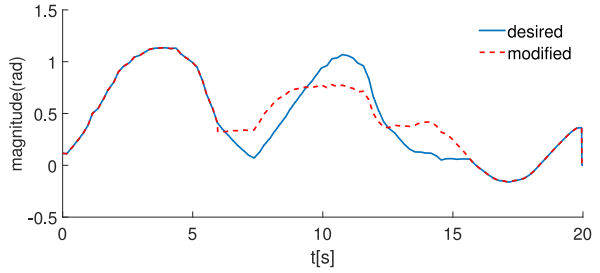


Fig. 14. Modified trajectory and the desired trajectory.

is set by the designer and applied at the manipulator from 6 s to 16 s. An admittance control is designed to track the modified trajectory affected by the external torque, which is estimated by the observer based on the generalized momentum approach. The experimental results are presented in Figs. 14–17. As depicted in Fig. 14, the desired trajectory q_d of joint w_0 will be modified by the external torque to enable the robot have a compliant behavior. The desired trajectory of joint e_1 will not be modified for the reason that the external torque is applied in the vertical direction and the trajectory of joint e_1 is in the horizontal direction. The tracking error under an admittance control is shown in Fig. 16 and the estimation of the external

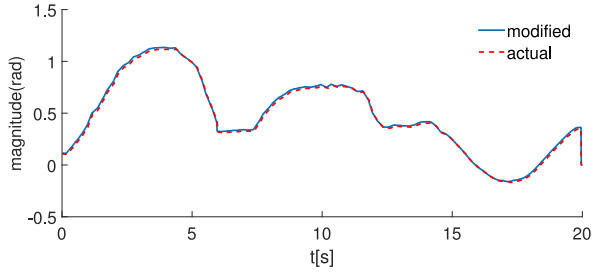


Fig. 15. Tracking performance under admittance control.

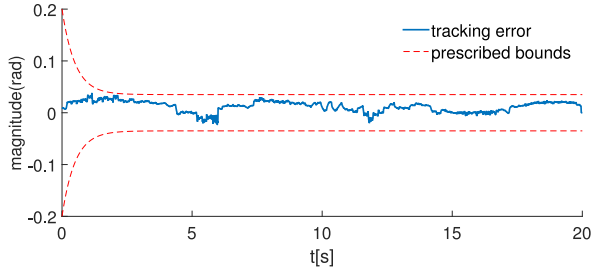


Fig. 16. Tracking error under admittance control.

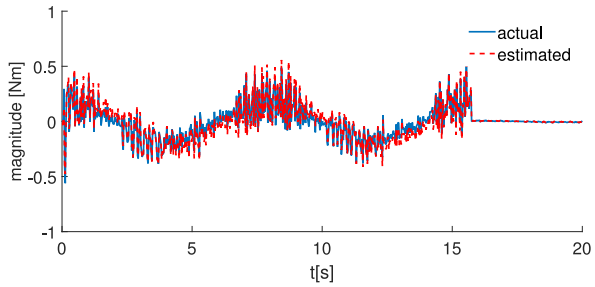


Fig. 17. Estimation of external torque.

torque is presented in Fig. 17. From the figures, the experimental results demonstrate the effectiveness of the proposed admittance control method.

V. Conclusion

In this paper, we proposed a sensorless control scheme for uncertain robot manipulator using NNs. We used a kinematics geometry vector-based method to calculate each joint angle of a human arm with Kinect sensor. The observer is used to estimate the external torque, which in turn is the input to admittance control. The error transformation method is used to ensure steady state performance and transient performance. The settling time, overshoot, and the final error can be achieved by changing the parameters of the error transformation functions. The RBFNN is employed to approximate the uncertainties of the manipulator dynamics in the system. The experimental results are provided to demonstrate the effectiveness of our developed methods. In the future, more effort will be taken to validate the proposed methods.

References

[1] G. Hirzinger, J. Bals, M. Otter, and J. Stelter, "The DLR-KUKA success story: Robotics research improves industrial robots," *IEEE Robot. Autom. Mag.*, vol. 12, no. 3, pp. 16–23, Sep. 2005.

[2] Y. Kim and W. C. Yoon, "Generating task-oriented interactions of service robots," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 44, no. 8, pp. 981–994, Aug. 2014.

[3] L. B. Kratchman, T. L. Bruns, J. J. Abbott, and R. J. Webster, "Guiding elastic rods with a robot-manipulated magnet for medical applications," *IEEE Trans. Robot.*, vol. 33, no. 1, pp. 227–233, Feb. 2017.

[4] H. Liu, J. Qin, F. Sun, and D. Guo, "Extreme kernel sparse learning for tactile object recognition," *IEEE Trans. Cybern.*, vol. 47, no. 12, pp. 4509–4520, Dec. 2017.

[5] H. Liu, F. Sun, D. Guo, B. Fang, and Z. Peng, "Structured output-associated dictionary learning for haptic understanding," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 7, pp. 1564–1574, Jul. 2017.

[6] A. Cappozzo, A. Cappello, U. D. Croce, and F. Pensalfini, "Surface-marker cluster design criteria for 3-D bone movement reconstruction," *IEEE Trans. Biomed. Eng.*, vol. 44, no. 12, pp. 1165–1174, Dec. 1997.

[7] A. Assa and F. Janabi-Sharifi, "A robust vision-based sensor fusion approach for real-time pose estimation," *IEEE Trans. Cybern.*, vol. 44, no. 2, pp. 217–227, Feb. 2014.

[8] Q. Peng, W. Chen, X. Wu, and J. Wang, "A novel vision-based human motion capture system using dual-kinect," in *Proc. IEEE 10th Conf. Ind. Electron. Appl. (ICIEA)*, Auckland, New Zealand, Jun. 2015, pp. 51–56.

[9] H. Boessenkool, D. A. Abbink, C. J. M. Heemskerk, F. C. T. van der Helm, and J. G. W. Wildenbeest, "A task-specific analysis of the benefit of haptic shared control during telemanipulation," *IEEE Trans. Haptics*, vol. 6, no. 1, pp. 2–12, Jan. 2013.

[10] G. Borenstein, *Making Things See: 3D Vision With Kinect, Processing, Arduino, and MakerBot*. Sebastopol, CA, USA: O'Reilly, 2012.

[11] J. Abhijit, *Kinect for Windows SDK Programming Guide*. Birmingham, U.K.: Packt, 2012.

[12] P. Liang, L. Ge, Y. Liu, L. Zhao, R. Li, and K. Wang, "An augmented discrete-time approach for human–robot collaboration," *Discr. Dyn. Nat. Soc.*, vol. 2016, p. 13, Feb. 2016.

[13] N. Hogan, "Impedance control: An approach to manipulation," in *Proc. Amer. Control Conf.*, San Diego, CA, USA, 1984, pp. 304–313.

[14] A. Alcocera, A. Robertssona, A. Valerac, and R. Johanssona, "Force estimation and control in robot manipulators," in *Proc. Vol. 7th IFAC Symp. Robot Control (SYROCO)*, vol. 1. Wrocław, Poland, Sep. 2004, pp. 55–60.

[15] A. Wahrburg, S. Zeiss, B. Matthias, and H. Ding, "Contact force estimation for robotic assembly using motor torques," in *Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)*, Taipei, Taiwan, 2014, pp. 1252–1257.

[16] A. De Luca and R. Mattone, "Sensorless robot collision detection and hybrid force/motion control," in *Proc. IEEE Int. Conf. Robot. Autom. (ICRA)*, Barcelona, Spain, 2005, pp. 999–1004.

[17] A. M. Smith, C. Yang, H. Ma, P. Culverhouse, A. Cangelosi, and E. Burdet, "Novel hybrid adaptive controller for manipulation in complex perturbation environments," *PLoS ONE*, vol. 10, no. 6, 2015, Art. no. e0129281.

[18] Z. Peng, G. Wen, A. Rahmani, and Y. Yu, "Distributed consensus-based formation control for multiple nonholonomic mobile robots with a specified reference trajectory," *Int. J. Syst. Sci.*, vol. 46, no. 8, pp. 1447–1457, 2015.

[19] F. Ficuciello, R. Carloni, L. C. Visser, and S. Stramigioli, "Port-Hamiltonian modeling for soft-finger manipulation," in *Proc. IEEE/RSJ Int. Conf. Intell. Robots Syst.*, Taipei, Taiwan, 2017, pp. 4281–4286.

[20] P. M. Kebria, H. Abdi, M. M. Dalvand, A. Khosravi, and S. Nahavandi, "Control methods for Internet-based teleoperation systems: A review," *IEEE Trans. Human–Mach. Syst.*, vol. 49, no. 1, pp. 32–46, Feb. 2019.

[21] W. He, S. Zhang, and S. S. Ge, "Robust adaptive control of a thruster assisted position mooring system," *Automatica*, vol. 50, no. 7, pp. 1843–1851, 2014.

[22] M. Li, Y. Li, S. S. Ge, and T. H. Lee, "Adaptive control of robotic manipulators with unified motion constraints," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 1, pp. 184–194, Jan. 2017.

[23] C. L. P. Chen, Y.-J. Liu, and G.-X. Wen, "Fuzzy neural network-based adaptive control for a class of uncertain nonlinear stochastic systems," *IEEE Trans. Cybern.*, vol. 44, no. 5, pp. 583–593, May 2014.

[24] P. M. Kebria, A. Khosravi, S. Nahavandi, Z. Najdovski, and S. J. Hilton, "Neural network adaptive control of teleoperation systems with uncertainties and time-varying delay," in *Proc. IEEE 14th Int. Conf. Autom. Sci. Eng. (CASE)*, Munich, Germany, 2018, pp. 252–257.

[25] Z. Liu, G. Lai, Y. Zhang, X. Chen, and C. L. P. Chen, "Adaptive neural control for a class of nonlinear time-varying delay systems with unknown hysteresis," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 12, pp. 2129–2140, Dec. 2014.

[26] Y.-J. Liu and S. Tong, "Adaptive NN tracking control of uncertain nonlinear discrete-time systems with nonaffine dead-zone input," *IEEE Trans. Cybern.*, vol. 45, no. 3, pp. 497–505, Mar. 2015.

[27] Z. Liu, C. Chen, Y. Zhang, and C. L. P. Chen, "Adaptive neural control for dual-arm coordination of humanoid robot with unknown nonlinearities in output mechanism," *IEEE Trans. Cybern.*, vol. 45, no. 3, pp. 507–518, Mar. 2015.

[28] G. Lai, Z. Liu, Y. Zhang, and C. L. P. Chen, "Adaptive position/attitude tracking control of aerial robot with unknown inertial matrix based on a new robust neural identifier," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 27, no. 1, pp. 18–31, Jan. 2016.

[29] L. Ding, S. Li, Y.-J. Liu, H. Gao, C. Chen, and Z. Deng, "Adaptive neural network-based tracking control for full-state constrained wheeled mobile robotic system," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 47, no. 8, pp. 2410–2419, Aug. 2017.

[30] C. Yang, Y. Jiang, Z. Li, W. He, and C.-Y. Su, "Neural control of bimanual robots with guaranteed global stability and motion precision," *IEEE Trans. Ind. Informat.*, vol. 13, no. 3, pp. 1162–1171, Jun. 2016.

[31] C. Yang, Z. Li, R. Cui, and B. Xu, "Neural network-based motion control of an underactuated wheeled inverted pendulum model," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 25, no. 11, pp. 2004–2016, Nov. 2014.

[32] 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.

[33] 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.

[34] S. Bianco, C. Cusano, and R. Schettini, "Single and multiple illuminant estimation using convolutional neural networks," *IEEE Trans. Image Process.*, vol. 26, no. 9, pp. 4347–4362, Sep. 2017.

[35] D. Wang, D. Liu, H. Li, B. Luo, and H. Ma, "An approximate optimal control approach for robust stabilization of a class of discrete-time nonlinear systems with uncertainties," *IEEE Trans. Syst., Man, Cybern., Syst.*, vol. 46, no. 5, pp. 713–717, May 2016.

[36] 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.

[37] Q. Liu and J. Wang, "A one-layer recurrent neural network for constrained nonsmooth optimization," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 41, no. 5, pp. 1323–1333, Oct. 2011.

[38] Z. Liu, F. Wang, Y. Zhang, and C. L. P. Chen, "Fuzzy adaptive quantized control for a class of stochastic nonlinear uncertain systems," *IEEE Trans. Cybern.*, vol. 46, no. 2, pp. 524–534, Feb. 2016.

[39] Y. Xu, R. Lu, H. Peng, and S. Xie, "Filtering for fuzzy systems with multiplicative sensor noises and multidensity quantizer," *IEEE Trans. Fuzzy Syst.*, vol. 26, no. 2, pp. 1011–1022, Apr. 2018.

[40] J. Ma, T. Yang, Z.-G. Hou, and M. Tan, "Adaptive neural network controller of a stewart platform with unknown dynamics for active vibration isolation," in *Proc. IEEE Int. Conf. Robot. Biomimetics*, Bangkok, Thailand, 2009, pp. 1631–1636.

[41] L. Cheng, Z.-G. Hou, and M. Tan, "Adaptive neural network tracking control for manipulators with uncertain kinematics, dynamics and actuator model," *Automatica*, vol. 45, no. 10, pp. 2312–2318, 2009.

[42] L. Cheng, Z.-G. Hou, M. Tan, and W. J. Zhang, "Tracking control of a closed-chain five-bar robot with two degrees of freedom by integration of an approximation-based approach and mechanical design," *IEEE Trans. Syst., Man, Cybern. B, Cybern.*, vol. 42, no. 5, pp. 1470–1479, Oct. 2012.

[43] A. Wahrburg, E. Morara, G. Cesari, B. Matthias, and H. Ding, "Cartesian contact force estimation for robotic manipulators using Kalman filters and the generalized momentum," in *Proc. IEEE Int. Conf. Autom. Sci. Eng. (CASE)*, Gothenburg, Sweden, 2015, pp. 1230–1235.

[44] B. Siciliano and O. Khatib, *Springer Handbook of Robotics*. Heidelberg, Germany: Springer, 2016.

[45] H. Kwakernaak and R. Sivan, *Linear Optimal Control Systems*, vol. 1. New York, NY, USA: Wiley-Intersci., 1972.

[46] J. Park and I. W. Sandberg, "Universal approximation using radial-basis-function networks," *Neural Comput.*, vol. 3, no. 2, pp. 246–257, Jun. 1991.

[47] W. H. Young, "On classes of summable functions and their Fourier series," *Proc. Roy. Soc. A Math. Phys. Eng. Sci.*, vol. 87, no. 594, pp. 225–229, 1912.

[48] G. Peng, C. Yang, Y. Jiang, L. Cheng, and P. Liang, "Teleoperation control of Baxter robot based on human motion capture," in *Proc. IEEE Int. Conf. Inf. Autom. (ICIA)*, Ningbo, China, Aug. 2016, pp. 1026–1031.

[49] L. Zhang, Z. Li, and C. Yang, "Adaptive neural network based variable stiffness control of uncertain robotic systems using disturbance observer," *IEEE Trans. Ind. Electron*, vol. 64, no. 3, pp. 2236–2245, Mar. 2017.
[50] H. Reddivari, C. Yang, Z. Ju, P. Liang, Z. Li, and B. Xu, "Teleoperation control of Baxter robot using body motion tracking," in *Proc. 2014 Int. Conf. Multisensor Fusion Inf. Integr. Intell. Syst. (MFI)*, Sep. 2014, pp. 1–6.

Chenguang Yang (M'10–SM'16) received the Ph.D. degree in control engineering from the National University of Singapore, Singapore, in 2010.

He was a Post-Doctoral Research Fellow of Human Robotics with Imperial College London, London, U.K., from 2009 to 2010. He is a Professor of Robotics with the University of the West of England, Bristol, U.K. His current research interests include human–robot interaction and intelligent system design.

Dr. Yang was a recipient of 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 conference best paper awards.

Guangzhu Peng received the B.Eng. degree in automation from Yangtze University, Jingzhou, China, in 2014, and the M.Eng. degree in pattern recognition and intelligent system from the College of Automation Science and Engineering, South China University of Technology, Guangzhou, China, in 2018. He is currently pursuing the Ph.D. degree in computer science with the Faculty of Science and Technology, University of Macau, Macau, China.

His current research interests include robotics, human–robot interaction, and intelligent control.

Long Cheng (SM'14) received the B.S. degree (Hons.) in control engineering from Nankai University, Tianjin, China, in 2004, and the Ph.D. degree (Hons.) in control theory and control engineering from the Institute of Automation, Chinese Academy of Sciences, Beijing, China, in 2009.

He is currently a Full Professor with the Institute of Automation, Chinese Academy of Sciences. He is also an Adjunct Professor with the University of Chinese Academy of Sciences, Beijing. He has published over 100 technical papers in peer-refereed journals and prestigious conference proceedings. His current research interests include rehabilitation robot, intelligent control, and neural networks.

Dr. Cheng was a recipient of the IEEE TRANSACTIONS ON NEURAL NETWORKS Outstanding Paper Award from IEEE Computational Intelligence Society, the Aharon Katzir Young Investigator Award from International Neural Networks Society, and the Young Researcher Award from Asia–Pacific Neural Networks Society. He is currently serving as an Associate Editor/Editorial Board Member for the IEEE TRANSACTIONS ON CYBERNETICS, *Neural Processing Letters*, *Neurocomputing*, *International Journal of Systems Science*, and *Acta Automatica Sinica*.

Jing Na (M'15) received the B.Sc. degree in automation and the Ph.D. degree in dynamics and control 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 Post-Doctoral Fellow with the ITER Organization, Saint-Paul-lès-Durance, France. From 2015 to 2017, he was a Marie Curie Intra-European 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. His current research interests include intelligent control, adaptive parameter estimation, and nonlinear control and applications.

Dr. Na was a recipient of the Best Application Paper Award of the Third IFAC International Conference on Intelligent Control and Automation Science in 2013 and the 2017 Hsue-Shen Tsien Paper Award.

Zhijun Li (M'07–SM'09) received the Ph.D. degree in mechatronics from Shanghai Jiao Tong University, Shanghai, China, in 2002.

From 2003 to 2005, he was a Post-Doctoral Fellow with the Department of Mechanical Engineering and Intelligent Systems, University of Electro-Communications, Tokyo, Japan. From 2005 to 2006, he was a Research Fellow with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, and Nanyang Technological University, Singapore. From 2012 to 2017, he was a Professor with the College of Automation Science and Engineering, South China University of Technology, Guangzhou, China. Since 2017, he has been a Professor with the Department of Automation, University of Science and Technology of China, Hefei, China. His current research interests include service robotics, teleoperation systems, nonlinear control, and neural network optimization.