

# Robust adaptive position and force controller design of robot manipulator using fuzzy neural networks

Ching-Hung Lee · Wei-Chen Wang

Received: 7 May 2015 / Accepted: 17 February 2016 / Published online: 9 March 2016  
© Springer Science+Business Media Dordrecht 2016

**Abstract** This paper presents a robust adaptive position and force control scheme for an  $n$ -link robot manipulator under unknown environment. The robot manipulator's **model** and the stiffness **coefficient** of contact environment are assumed to be not exactly known. Therefore, the traditional impedance force controller cannot be applied. We herein adopt the fuzzy neural networks (FNNs) to estimate the unknown model matrices of robot manipulator and the adaptive tracking position and force control is developed by the proposed adaptive scheme. Based on the Lyapunov stability theory, the stability of the closed-loop system and convergence of adjustable parameters are guaranteed. The corresponding update laws of FNNs' parameters and estimated stiffness coefficient of contacting environment can be derived. Finally, simulation results of a two-link robot manipulator with environment constraint are introduced to illustrate the performance and effectiveness of our approach.

**Keywords** Robot manipulator · Force control · Adaptive control · Fuzzy neural networks

## List of symbols

$\mathbf{q}$  Joint position

C.-H. Lee (✉) · W.-C. Wang  
Department of Mechanical Engineering, National Chung Hsing University, Taichung, Taiwan, ROC  
e-mail: chleenchu@dragon.nchu.edu.tw

$\dot{\mathbf{q}}$	Joint velocity
$\ddot{\mathbf{q}}$	Joint acceleration
$\mathbf{x}$	End-effector position in Cartesian space
$\dot{\mathbf{x}}$	End-effector velocity in Cartesian space
$\ddot{\mathbf{x}}$	End-effector acceleration in Cartesian space
$\mathbf{x}_d$	Desired end-effector trajectory
$\mathbf{D}(\mathbf{q})$	Robot's inertial matrix in joint space
$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$	Robot's Coriolis matrix in joint space
$\mathbf{g}(\mathbf{q})$	Robot's gravitational vector in joint space
$\mathbf{D}_x(\mathbf{x})$	Robot's inertial matrix in Cartesian space
$\mathbf{C}_x(\mathbf{x}, \dot{\mathbf{x}})$	Robot's Coriolis matrix in Cartesian space
$\mathbf{g}_x(\mathbf{x})$	Robot's gravitational vector in Cartesian space
$\boldsymbol{\tau}$	Joint torque vector
$\mathbf{J}$	Robot's Jacobian
$\mathbf{F}_{\text{ext}}$	External force at the end-effector
$\boldsymbol{\tau}_{\text{robust}}$	Robust controller
$\hat{\mathbf{D}}_x$	FNN's estimated robot's inertial matrix in Cartesian space
$\hat{\mathbf{C}}_x$	FNN's estimated robot's Coriolis matrix in Cartesian space
$\hat{\mathbf{g}}_x$	FNN's estimated robot's gravitational vector in Cartesian space
$\mathbf{w}$	Weighting vector of the FNN
$\psi$	Normalized firing strength of the FNN
$\mathbf{K}_e$	Stiffness coefficient of contact environment
$\hat{\mathbf{K}}_e$	Estimated stiffness coefficient of contact environment

$\eta\kappa_e$	Learning rate for tuning $\hat{\mathbf{K}}_e$
$\mathbf{e}_f$	Tracking force error

## 1 Introduction

Recently, various control strategies for treating the **interaction control problem** of robot manipulator are proposed, e.g., impedance control and the hybrid force control algorithms [1,3,5–13,15,16,28–36,38,40]. The impedance control of robot manipulators is to adjust the end-effector position and sense the contact force in response to satisfy the target impedance behavior. Several robust control schemes and adaptive control strategies have been proposed for motion control of robot manipulator [1,3,4,10,31,34–36,38]. In the literature [5,12,13,40], adaptive impedance control schemes using approximation technique for an  $n$ -link robot manipulator without regressor are introduced. The modifications of impedance function have been proposed to solve the force tracking problem on unknown environment [16].

As above, the purpose of this paper is to introduce a robust adaptive force and position tracking problem for robot manipulator with model uncertainties and the task environment is unknown. Many approaches are proposed to deal with the adaptive controller design for systems with uncertainties and disturbance. In [17–19], the neural network approach was proposed to treat the system uncertainties in force control problem. In addition, an on-line learning method was introduced to regulate all impedance parameters as well as desired trajectory [39]. Similarly, the fuzzy logic control was developed by human experience to implement nonlinear control algorithms [21–25]. Besides, the recurrent fuzzy neural network is introduced to deal with uncertainties and disturbances [33]. Therefore, the fuzzy neural networks (FNNs) are adopted to treat the tracking control of robot manipulator.

In this paper, a robust FNN-based adaptive force tracking control is proposed for robot manipulator with uncertainties and unknown stiffness coefficient. The dynamic model of the robot manipulator and the stiffness coefficient of contact environment are assumed to be unknown. Therefore, we cannot solve the force tracking control by using the traditional approaches. Thus, the FNN estimators are adopted to approximate the system's dynamic model and to develop the proposed adaptive tracking control scheme. In addition,

the adaptive force control law is established by gradient method. Based on the Lyapunov stability approach, the update laws of FNNs' parameters and estimated stiffness coefficient are obtained. In addition, the stability of the closed-loop system and convergence of parameters are guaranteed. Finally, simulation results of a two-link robot manipulator with environment constraint are introduced to illustrate the performance and effectiveness of our approach.

The rest of this paper is as follows. Section 2 introduces the problem formulation and the used fuzzy neural network system. The proposed adaptive force control scheme is introduced in Sect. 3. The theoretical analysis is also introduced here. In Sect. 4, simulation results of a two-link robot manipulator are presented. Finally, conclusion is given.

## 2 Problem formulation

### 2.1 System description

The dynamic model of an  $n$ -link rigid robot manipulator in **joint space** coordinate is

$$\mathbf{D}(\mathbf{q})\ddot{\mathbf{q}} + \mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} + \mathbf{g}(\mathbf{q}) = \boldsymbol{\tau} - \mathbf{J}^T \mathbf{F}_{\text{ext}} \quad (1)$$

where  $\mathbf{q} \in \mathbb{R}^n$  denotes the vector of generalized displacement in robot coordinates,  $\boldsymbol{\tau} \in \mathbb{R}^n$  is the vector of joint torque,  $\mathbf{D}(\mathbf{q}) \in \mathbb{R}^{n \times n}$  is the inertia matrix,  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})\dot{\mathbf{q}} \in \mathbb{R}^n$  is the vector of centrifugal and Coriolis forces,  $\mathbf{g}(\mathbf{q}) \in \mathbb{R}^n$  is the gravity vector.  $\mathbf{J}(\mathbf{q}) \in \mathbb{R}^{n \times n}$  is the Jacobian matrix, which is assumed to be non-singular, and  $\mathbf{F}_{\text{ext}} \in \mathbb{R}^n$  is the external force at the end-effector. In the case of  $n > 6$ , the manipulator is redundant; then, the inverse of Jacobian ( $\mathbf{J}^{-1}$ ) should be replaced by pseudo-inverse  $\mathbf{J}^+ = \mathbf{J}^T(\mathbf{J}\mathbf{J}^T)^{-1}$ . System (1) can be transferred to **the Cartesian space** as

$$\mathbf{D}_x(\mathbf{x})\ddot{\mathbf{x}} + \mathbf{C}_x(\mathbf{x}, \dot{\mathbf{x}})\dot{\mathbf{x}} + \mathbf{g}_x(\mathbf{x}) = \mathbf{J}^{-T}\boldsymbol{\tau} - \mathbf{F}_{\text{ext}} \quad (2)$$

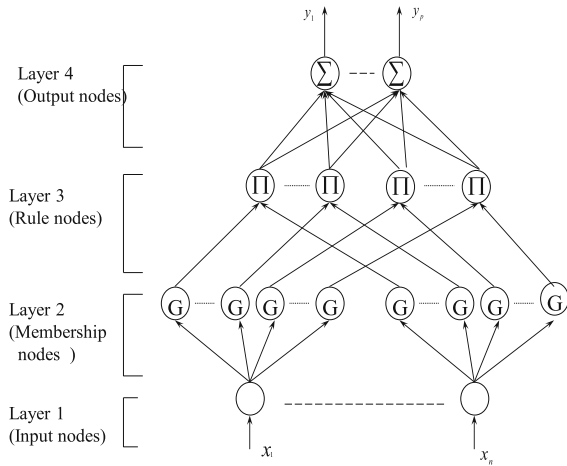
where  $\mathbf{x} \in \mathbb{R}^n$  is the location vector of the end-effector in Cartesian space and

$$\mathbf{D}_x(\mathbf{x}) = \mathbf{J}^{-T}\mathbf{D}(\mathbf{q})\mathbf{J}^{-1} \quad (3)$$

$$\mathbf{C}_x(\mathbf{x}, \dot{\mathbf{x}}) = \mathbf{J}^{-T}(\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) - \mathbf{D}(\mathbf{q})\mathbf{J}^{-1}\dot{\mathbf{J}})\mathbf{J}^{-1} \quad (4)$$

$$\mathbf{g}_x(\mathbf{x}) = \mathbf{J}^{-T}\mathbf{g}(\mathbf{q}). \quad (5)$$

In most practical cases, the system dynamic model are not exactly known, thus identification or modeling of the robot manipulator model are needed for the controller design. In this paper, the FNN estimators solve



**Fig. 1** Schematic diagram of fuzzy neural network [21–23]

this problem, i.e., the FNNs estimate the functions of  $\mathbf{D}_x$ ,  $\mathbf{C}_x$ , and  $\mathbf{g}_x$ .

The purpose of this paper is to provide proper control signals such that the end-effector of robot manipulator follows the desired trajectory for free space and tracks the desired force for contact space even the system with uncertainties and works under unknown environment (stiffness coefficient is unknown).

## 2.2 Fuzzy neural network system

Herein, we introduce the FNN systems briefly as follows [21–23]. The schematic diagram of the used FNN with four layers is shown in Fig. 1, where “G” denotes the Gaussian fuzzy term set. **Layer 1** is the input layer that accepts input variables and its nodes represent fuzzy input linguistic variables. The nodes in this layer only transmit input variables to the next layer directly, i.e.,  $O_i^{(1)} = x_i$ . **Layer 2** operates as the fuzzification; each node calculates the corresponding membership grade for input  $x_i$

$$O_{ij}^{(2)} = e^{-\frac{(x_i - m_{ij})^2}{(\sigma_{ij})^2}} \quad (6)$$

where  $m_{ij}$  and  $\sigma_{ij}$  are the center and the width of the Gaussian function, respectively. In **layer 3** (rule layer), each node represents the fuzzy rule and the  $t$ -norm product operation is adopted. Links before layer 3 represent the preconditions of the rules, and the links after layer 3 represent the consequences of the fuzzy rule, the output of Layer 3 is

$$O_j^{(3)} = \prod_i O_{ij}^{(2)}. \quad (7)$$

**Layer 4** is the output layer. Each node is for actual output of FNN system that is connected by weighting value  $w_j$ , the  $p^{\text{th}}$  output is

$$y_p = O^{(4)} = \mathbf{w}^T \boldsymbol{\Psi} = \frac{\sum_{j=1}^R w_j O_j^{(3)}}{\sum_{j=1}^R O_j^{(3)}} \quad (8)$$

where  $\mathbf{w} = [w_1 w_2, \dots, w_R]^T$  is the weighting vector and  $\boldsymbol{\Psi} = [\Psi^1 \Psi^2, \dots, \Psi^R]$ ,  $\Psi^j = O_j^{(3)} / \sum_j O_j^{(3)}$  and  $R$  is the chosen rule number. As above, there is  $R$  adjustable parameters for a single output FNN with  $R$  rules. Herein, the FNN is adopted to estimate the unknown matrices of robot manipulator. In addition, the FNN realizes the fuzzy inference into the network structure. Assume that an FNN system has  $R$  rules and  $n$  inputs; the  $j$ th rule can be expressed as

$$\text{Rule } j: \text{ IF } x_1 \text{ is } A_{1j} \text{ and } \dots x_n \text{ is } A_{nj}, \text{ THEN } y_1 \text{ is } \omega_{1j} \text{ and } \dots y_p \text{ is } \omega_{pj} \quad (9)$$

where  $x_j$  denotes the input linguistic variable,  $A_{1j}$  is the fuzzy set represented by the Gaussian fuzzy term set, and  $\omega_{pj}$  is the consequent part parameter for inference output  $y_p$ . That is, the FNN provides the linguistic representation for the approximated function. In addition, as the mention of [21], the FNNs have the properties of universal approximator, fast convergence, less adjustable parameters, and parallel distribution scheme. The universal approximation theorem states “if the FNN has a sufficiently large number of fuzzy logical rules (or neurons), then it can approximate any continuous function in  $C(\mathcal{R}^n)$  over a compact subset of  $\mathcal{R}^n$ .” Therefore, the FNNs have the abilities of estimating the system dynamic matrices.

## 3 Robust adaptive position and force control scheme under unknown stiffness coefficient

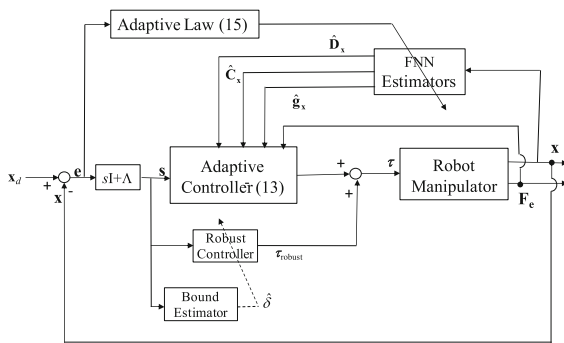
As the results of traditional impedance control [2], the corresponding system parameters are exactly known, and then the closed-loop system satisfies the target impedance

$$\begin{aligned} \mathbf{M}_m (\ddot{\mathbf{x}} - \ddot{\mathbf{x}}_d) + \mathbf{B}_m (\dot{\mathbf{x}} - \dot{\mathbf{x}}_d) + \mathbf{K}_m (\mathbf{x} - \mathbf{x}_d) \\ = -\mathbf{F}_{\text{ext}} \end{aligned} \quad (10)$$

where  $\mathbf{M}_m$ ,  $\mathbf{B}_m$ , and  $\mathbf{K}_m$  are diagonal  $n \times n$  matrices gain,  $\mathbf{x}_d \in \mathcal{R}^n$  is the reference end-effector trajectory

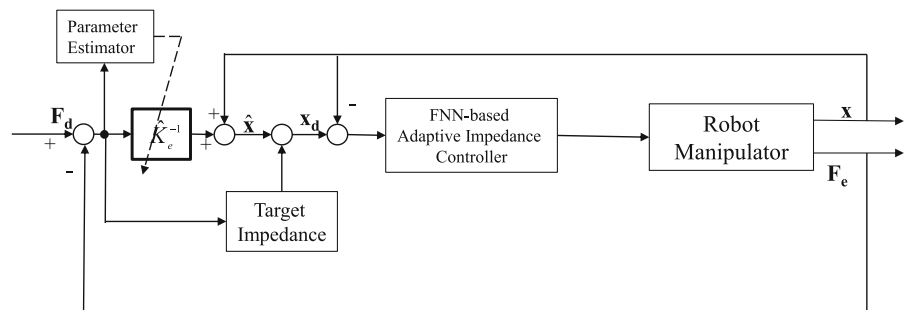
(reference location). However, system dynamic model is not exactly known, the proposed adaptive control utilizes the FNNs to approximate the system model matrices (denote as  $\hat{\mathbf{D}}_x$ ,  $\hat{\mathbf{C}}_x$ , and  $\hat{\mathbf{g}}_x$ ) for solving the tracking control problem.

Herein, the control objective is divided into two cases; **Case (a)** is to design an adaptive controller such that  $\mathbf{x}$  approaches to  $\mathbf{x}_d$  asymptotically for free space ( $\mathbf{F}_{\text{ext}} = 0$ ); **Case (b)**, we should pay additional effort to design the adaptive force controller such that  $\mathbf{F}_{\text{ext}}$  is equal to the desired force  $\mathbf{F}_d$  for contact space. However, the environment stiffness is not exactly known, then we cannot design the reference trajectory  $\mathbf{x}$  to achieve the desired force directly. In this paper, we solve the problem by using FNN-based adaptive tracking controller firstly, shown in Fig. 2. Based on the Lyapunov stability approach, the stability analysis of the closed-loop system and the update laws of FNNs' parameters are obtained. After solving the tracking control problem, the adaptive force tracking scheme is introduced for contact environment with unknown stiffness coefficient, shown in Fig. 3. The corresponding parameter estimator and update law are proposed based



**Fig. 2** The proposed adaptive tracking control scheme using FNNs

**Fig. 3** Force tracking control scheme using FNNs



on the Lyapunov approach that guarantees the convergence. More details are introduced below.

### 3.1 Adaptive tracking controller design

Herein, we first consider **the adaptive tracking control for free space**. The system model matrices  $\mathbf{D}_x$ ,  $\mathbf{C}_x$ , and  $\mathbf{g}_x$  are not exactly known which are estimated by FNNs. Before introducing the proposed approach, we first define  $\mathbf{e} = \mathbf{x} - \mathbf{x}_d$  and  $\mathbf{s} = \dot{\mathbf{e}} + \Lambda \mathbf{e}$ , where  $\Lambda = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_n)$  with  $\lambda_i > 0$  for  $i = 1, \dots, n$ . Rewrite system (2) as

$$\hat{\mathbf{D}}_x \dot{\mathbf{s}} + \hat{\mathbf{C}}_x \mathbf{s} + \hat{\mathbf{g}}_x + \hat{\mathbf{D}}_x \ddot{\mathbf{x}}_d - \hat{\mathbf{D}}_x \Lambda \dot{\mathbf{e}} + \hat{\mathbf{C}}_x \dot{\mathbf{x}}_d - \hat{\mathbf{C}}_x \Lambda \mathbf{e} = \mathbf{J}^{-T} \boldsymbol{\tau} - \mathbf{F}_{\text{ext}} \quad (11)$$

where  $\hat{\mathbf{D}}_x = \hat{\mathbf{w}}_D^T \boldsymbol{\psi}_D$ ,  $\hat{\mathbf{C}}_x = \hat{\mathbf{w}}_C^T \boldsymbol{\psi}_C$ , and  $\hat{\mathbf{g}}_x = \hat{\mathbf{w}}_g^T \boldsymbol{\psi}_g$  are the estimations of  $\mathbf{D}_x$ ,  $\mathbf{C}_x$ , and  $\mathbf{g}_x$ , respectively.  $\hat{\mathbf{w}}$  denotes the weighting matrix of the FNN, and  $\boldsymbol{\psi}$  is the normalized firing strength of the corresponding fuzzy rules (it also viewed as basis function matrix). According to the universal approximation property of the FNNs, there exists optimal weighting matrix  $\mathbf{w}^*$  such that the unknown functions are represented as follows with approximation error  $\varepsilon$ , i.e.,

$$\mathbf{D}_x = \mathbf{w}_D^{*T} \boldsymbol{\psi}_D + \varepsilon_{D_x} \quad (12a)$$

$$\mathbf{C}_x = \mathbf{w}_C^{*T} \boldsymbol{\psi}_C + \varepsilon_{C_x} \quad (12b)$$

$$\mathbf{g}_x = \mathbf{w}_g^{*T} \boldsymbol{\psi}_g + \varepsilon_{g_x} \quad (12c)$$

where  $\mathbf{w}_D^* \in \mathbb{R}^{n^2 R}$ ,  $\mathbf{w}_C^* \in \mathbb{R}^{n^2 R}$ , and  $\mathbf{w}_g^* \in \mathbb{R}^{n R}$  and  $\boldsymbol{\psi}_D \in \mathbb{R}^{n^2 R}$ ,  $\boldsymbol{\psi}_C \in \mathbb{R}^{n^2 R}$ ,  $\boldsymbol{\psi}_G \in \mathbb{R}^{n R}$ .  $R$  represents the rule number of FNN. Note that  $\hat{\mathbf{D}}_x \in \mathbb{R}^{n \times n}$  has  $n \times n$  functions, then we have  $n \times n \times R$  adjustable parameters when each unknown function is approximated by an FNN having  $R$  rules.

As above, we have the following theorem to guarantee the stability of the closed-loop systems for tracking control problem.

**Theorem 1** Consider the nonlinear system (2) with uncertainties and nonsingular property, i.e.,  $\mathbf{D}_x$ ,  $\mathbf{C}_x$ , and  $\mathbf{g}_x$  are not exactly known, the adaptive controllers are designed as

$$\boldsymbol{\tau} = \mathbf{J}^T \left( \mathbf{F}_{\text{ext}} + \hat{\mathbf{g}}_x + \hat{\mathbf{D}}_x \ddot{\mathbf{x}}_d - \hat{\mathbf{D}}_x \Lambda \dot{\mathbf{e}} + \hat{\mathbf{C}}_x \dot{\mathbf{x}}_d - \hat{\mathbf{C}}_x \Lambda \mathbf{e} - \mathbf{K}_d \mathbf{s} \right) + \boldsymbol{\tau}_{\text{robust}} \quad (13)$$

$$\boldsymbol{\tau}_{\text{robust}} = -\delta [\text{sgn}(s_1), \dots, \text{sgn}(s_n)]^T \quad (14)$$

where  $\boldsymbol{\tau}_{\text{robust}}$  is the robust controller which plays the role of compensator,  $s_i$ ,  $i = 1, \dots, n$  is the  $i$ th element of  $\mathbf{s}$ ,  $\delta$  is the selected amplitude of robust controller (robust control gain), and  $\mathbf{K}_d$  is positive matrix in which is usually chosen as diagonal matrix, i.e.,  $\mathbf{K}_d = \text{diag}(k_{d1}, k_{d2}, \dots, k_{dn})$ . In addition, the update laws of FNNs are chosen as

$$\dot{\hat{\mathbf{W}}}_D = -\mathbf{Q}_D^{-1} [\boldsymbol{\Psi}_D (\ddot{\mathbf{x}}_d - \Lambda \dot{\mathbf{e}}) \mathbf{s}^T] \quad (15a)$$

$$\dot{\hat{\mathbf{W}}}_C = -\mathbf{Q}_C^{-1} [\boldsymbol{\Psi}_C (\dot{\mathbf{x}}_d - \Lambda \mathbf{e}) \mathbf{s}^T] \quad (15b)$$

$$\dot{\hat{\mathbf{W}}}_g = -\mathbf{Q}_g^{-1} (\boldsymbol{\Psi}_g \mathbf{s}^T) \quad (15c)$$

where  $\mathbf{Q}_D \in \mathbb{R}^{n^2 R \times n^2 R}$ ,  $\mathbf{Q}_C \in \mathbb{R}^{n^2 R \times n^2 R}$ ,  $\mathbf{Q}_g \in \mathbb{R}^{n R \times n R}$  are positive definite matrices. Then, the asymptotically convergence of tracking error is guaranteed, i.e., the state  $\mathbf{x}$  will follow the desired trajectory  $\mathbf{x}_d$  when  $t$  approaches to infinity.

*Proof* Substituting (11) into (2)

$$\begin{aligned} \mathbf{D}_x \dot{\mathbf{s}} + \mathbf{C}_x \mathbf{s} + \mathbf{K}_d \mathbf{s} &= (\hat{\mathbf{D}}_x - \mathbf{D}_x) (\ddot{\mathbf{x}}_d - \Lambda \dot{\mathbf{e}}) \\ &+ (\hat{\mathbf{C}}_x - \mathbf{C}_x) (\dot{\mathbf{x}}_d - \Lambda \mathbf{e}) + (\hat{\mathbf{g}}_x - \mathbf{g}_x) + \boldsymbol{\tau}_{\text{robust}}. \end{aligned} \quad (16)$$

By the representation of (11) and (12), we have

$$\begin{aligned} \mathbf{D}_x \dot{\mathbf{s}} + \mathbf{C}_x \mathbf{s} + \mathbf{K}_d \mathbf{s} &= \tilde{\mathbf{W}}_D^T \boldsymbol{\Psi}_D (\ddot{\mathbf{x}}_d - \Lambda \dot{\mathbf{e}}) \\ &+ \tilde{\mathbf{W}}_C^T \boldsymbol{\Psi}_C (\dot{\mathbf{x}}_d - \Lambda \mathbf{e}) + \tilde{\mathbf{W}}_g^T \boldsymbol{\Psi}_g + \boldsymbol{\varepsilon}_1 + \boldsymbol{\tau}_{\text{robust}} \end{aligned} \quad (17)$$

where  $\tilde{\mathbf{W}} = \hat{\mathbf{W}} - \mathbf{W}^*$  and  $\boldsymbol{\varepsilon}_1 = \boldsymbol{\varepsilon}_1(\boldsymbol{\varepsilon}_{D_x}, \boldsymbol{\varepsilon}_{C_x}, \boldsymbol{\varepsilon}_{g_x}, \mathbf{s}, \ddot{\mathbf{x}})$  denotes the lumped bounded approximation error of  $\mathbf{D}_x$ ,  $\mathbf{C}_x$ , and  $\mathbf{g}_x$ . Define the Lyapunov candidate function

$$\begin{aligned} V(\mathbf{s}, \tilde{\mathbf{W}}_D, \tilde{\mathbf{W}}_C, \tilde{\mathbf{W}}_g) &= \frac{1}{2} \mathbf{s}^T \mathbf{D}_x \mathbf{s} \\ &+ \frac{1}{2} \text{Trace} \left( \tilde{\mathbf{W}}_D^T \mathbf{Q}_D \tilde{\mathbf{W}}_D + \tilde{\mathbf{W}}_C^T \mathbf{Q}_C \tilde{\mathbf{W}}_C + \tilde{\mathbf{W}}_g^T \mathbf{Q}_g \tilde{\mathbf{W}}_g \right) \end{aligned} \quad (18)$$

where  $\mathbf{Q}_D \in \mathbb{R}^{n^2 R \times n^2 R}$ ,  $\mathbf{Q}_C \in \mathbb{R}^{n^2 R \times n^2 R}$ ,  $\mathbf{Q}_g \in \mathbb{R}^{n R \times n R}$  are positive definite matrices and  $\text{Trace}(\cdot)$  is

the trace operation. Hence

$$\begin{aligned} \dot{V} &= \mathbf{s}^T \left( -\mathbf{C}_x \mathbf{s} - \mathbf{K}_d \mathbf{s} + \tilde{\mathbf{W}}_D^T \boldsymbol{\Psi}_D (\ddot{\mathbf{x}}_d - \Lambda \dot{\mathbf{e}}) \right. \\ &\quad \left. + \tilde{\mathbf{W}}_C^T \boldsymbol{\Psi}_C (\dot{\mathbf{x}}_d - \Lambda \mathbf{e}) + \tilde{\mathbf{W}}_g^T \boldsymbol{\Psi}_g + \boldsymbol{\varepsilon}_1 + \boldsymbol{\tau}_{\text{robust}} \right) \\ &+ \frac{1}{2} \mathbf{s}^T \dot{\mathbf{D}}_x \mathbf{s} + \text{Trace}(\tilde{\mathbf{W}}_D^T \mathbf{Q}_D \dot{\tilde{\mathbf{W}}}_D + \tilde{\mathbf{W}}_C^T \mathbf{Q}_C \dot{\tilde{\mathbf{W}}}_C \\ &\quad + \tilde{\mathbf{W}}_g^T \mathbf{Q}_g \dot{\tilde{\mathbf{W}}}_g) \end{aligned} \quad (19)$$

then

$$\begin{aligned} \dot{V} &= -\mathbf{s}^T \mathbf{K}_d \mathbf{s} + \frac{1}{2} \mathbf{s}^T [\dot{\mathbf{D}}_x - 2\mathbf{C}_x] \mathbf{s} \\ &+ \text{Trace} \left[ \tilde{\mathbf{W}}_D^T \left( \boldsymbol{\Psi}_D (\ddot{\mathbf{x}}_d - \Lambda \dot{\mathbf{e}}) \mathbf{s}^T + \mathbf{Q}_D \dot{\tilde{\mathbf{W}}}_D \right) \right. \\ &\quad \left. + \tilde{\mathbf{W}}_C^T \left( \boldsymbol{\Psi}_C (\dot{\mathbf{x}}_d - \Lambda \mathbf{e}) \mathbf{s}^T + \mathbf{Q}_C \dot{\tilde{\mathbf{W}}}_C \right) \right. \\ &\quad \left. + \tilde{\mathbf{W}}_g^T \left( \boldsymbol{\Psi}_g \mathbf{s}^T + \mathbf{Q}_g \dot{\tilde{\mathbf{W}}}_g \right) \right] + \mathbf{s}^T \boldsymbol{\varepsilon}_1 + \mathbf{s}^T \boldsymbol{\tau}_{\text{robust}}. \end{aligned} \quad (20)$$

Note that  $\mathbf{s}^T [\dot{\mathbf{D}}_x(\mathbf{x}) - 2\mathbf{C}_x(\mathbf{x}, \dot{\mathbf{x}})] \mathbf{s} = 0$  for all  $\mathbf{s} \in \mathbb{R}^n$ . We choose the FNNs' update laws (15) and robust controller (14) such that

$$\begin{aligned} \dot{V} &\leq -\mathbf{s}^T \mathbf{K}_d \mathbf{s} + \mathbf{s}^T \boldsymbol{\varepsilon}_1 - \delta |\mathbf{s}| \\ &\leq -\mathbf{s}^T \mathbf{K}_d \mathbf{s} + |\mathbf{s}| |\boldsymbol{\varepsilon}_1| - \delta |\mathbf{s}| \\ &\leq -\mathbf{s}^T \mathbf{K}_d \mathbf{s} - |\mathbf{s}| (\delta - |\boldsymbol{\varepsilon}_1|) \\ &\leq 0. \end{aligned} \quad (21)$$

This concludes that the convergence of  $\mathbf{s}$  is guaranteed if the proper value of  $\delta$  is selected. This completes the proof.  $\square$

**Remark 1** To avoid the highly control gain of robust controller, we have the robust controller with adaptive boundary as follows

$$\boldsymbol{\tau}_{\text{robust}} = -\hat{\delta} [\text{sgn}(s_1), \dots, \text{sgn}(s_n)]^T \quad (22)$$

$$\dot{\hat{\delta}} = -\gamma_{\delta} |\mathbf{s}|. \quad (23)$$

where  $\boldsymbol{\tau}_{\text{robust}}$  is the robust controller (or compensation controller) with adaptive boundary  $\hat{\delta}$  and  $\delta^*$  denotes the existed optimal bound of robust controller. Thus, the corresponding Lyapunov function candidate of  $V$  is

$$\begin{aligned} V(\mathbf{s}, \tilde{\mathbf{W}}_D, \tilde{\mathbf{W}}_C, \tilde{\mathbf{W}}_g, \tilde{\delta}) &= \frac{1}{2} \mathbf{s}^T \mathbf{D}_x \mathbf{s} + \frac{1}{2\gamma_{\delta}} \tilde{\delta}^2 \\ &+ \frac{1}{2} \text{Trace} \left( \tilde{\mathbf{W}}_D^T \mathbf{Q}_D \tilde{\mathbf{W}}_D + \tilde{\mathbf{W}}_C^T \mathbf{Q}_C \tilde{\mathbf{W}}_C + \tilde{\mathbf{W}}_g^T \mathbf{Q}_g \tilde{\mathbf{W}}_g \right) \end{aligned} \quad (24)$$

where  $\tilde{\delta} \equiv \hat{\delta} - \delta^*$  denotes the estimated error of approximated error bound. According the above results, we can obtain

$$\begin{aligned}\dot{V} &\leq -\mathbf{s}^T \mathbf{K}_d \mathbf{s} + |\mathbf{s}| |\mathbf{e}_1| + \frac{1}{\gamma_\delta} \tilde{\delta} \dot{\hat{\delta}} + \mathbf{s}^T \boldsymbol{\tau}_{\text{robust}} \\ &\leq -\mathbf{s}^T \mathbf{K}_d \mathbf{s} + |\mathbf{s}| |\mathbf{e}_1| - \hat{\delta} |\mathbf{s}| + \frac{1}{\gamma_\delta} \tilde{\delta} \dot{\hat{\delta}} \\ &\leq -\mathbf{s}^T \mathbf{K}_d \mathbf{s} + |\mathbf{s}| |\mathbf{e}_1| - \hat{\delta} |\mathbf{s}| + (\hat{\delta} - \delta^*) |\mathbf{s}| \\ &\leq -\mathbf{s}^T \mathbf{K}_d \mathbf{s} - |\mathbf{s}| (\delta^* - |\mathbf{e}_1|) \leq 0.\end{aligned}\quad (25)$$

Therefore, substitute (23) into (25), we have  $\dot{V} \leq 0$  and asymptotic convergence of  $\mathbf{s}$  can be concluded. This guarantees the stability of the closed-loop system.

The remaining work is to develop the adaptive force tracking control for contact space when the environment stiffness is unknown.

### 3.2 Adaptive force control scheme

As above description, our goal is to achieve  $\mathbf{F}_{\text{ext}} = \mathbf{F}_d$ . In contact cases, the end-effector position of robot manipulator  $\mathbf{x}$  satisfies the target impedance equation (10). The inertia and damping parameters only influence the transient response of the end-effector [14, 26]; thus, the steady-state contact force is

$$\mathbf{F}_{\text{ext}} \cong \mathbf{K}_m (\mathbf{x} - \mathbf{x}_d). \quad (26)$$

Considering for simplicity the environment modeled by a linear spring with stiffness  $\mathbf{K}_e$ , the contact force is given by

$$\mathbf{F}_{\text{ext}} \cong \mathbf{K}_e (\mathbf{x} - \mathbf{x}_e) \quad (27)$$

where  $\mathbf{K}_e$  is the stiffness coefficient of contact environment and  $\mathbf{x}_e$  is the position of environment. This leads to the following steady-state position and contact force [36]

$$\mathbf{x}_{ss} = (\mathbf{K}_m + \mathbf{K}_e)^{-1} (\mathbf{K}_d \mathbf{x}_d + \mathbf{K}_e \mathbf{x}_e). \quad (28)$$

As above description, the desired force  $\mathbf{F}_d$  between robot and environment is obtained when  $\mathbf{K}_e$  is known

$$\mathbf{F}_d \cong \mathbf{K}_e (\mathbf{x}^* - \mathbf{x}_e) \quad (29)$$

where  $\mathbf{x}^*$  is ideal trajectory such that  $\mathbf{F}_{\text{ext}} = \mathbf{F}_d$  which is achieved by designing reference trajectory from (28)

$$\mathbf{x}_d^* = \mathbf{K}_m^{-1} [(\mathbf{K}_m + \mathbf{K}_e) \mathbf{x}^* - \mathbf{K}_e \mathbf{x}_e]. \quad (30)$$

However, the stiffness coefficient  $\mathbf{K}_e$  is usually unknown; therefore,  $\mathbf{x}^*$  cannot be obtained directly.

From (26) and (29), the tracking error of force can be represented as

$$\mathbf{F}_d - \mathbf{F}_{\text{ext}} \cong \mathbf{K}_e (\mathbf{x}^* - \mathbf{x}). \quad (31)$$

Since the stiffness should be estimated, thus we rewrite (29) as

$$\mathbf{F}_d \cong \hat{\mathbf{K}}_e (\mathbf{x}^* - \mathbf{x}_e) \quad (32)$$

where  $\hat{\mathbf{K}}_e$  is the estimated stiffness coefficient. In this paper, the update law of  $\hat{\mathbf{K}}_e$  is derived by the gradient method and the convergence is guaranteed by Lyapunov approach. To obtain the adaptation of  $\hat{\mathbf{K}}_e$ , we first define the force error and the corresponding objective function to be minimized as

$$\mathbf{e}_f = \mathbf{F}_d - \mathbf{F}_{\text{ext}} \quad (33)$$

and the Lyapunov candidate function is defined

$$V_f = \frac{1}{2} \mathbf{e}_f^T \mathbf{e}_f. \quad (34)$$

For simplicity, we consider that the force is applied to only one direction, therefore,  $\mathbf{e}_f$ ,  $\mathbf{F}_d$ , and  $\mathbf{F}_{\text{ext}}$  are scalar, i.e.,  $e_f = f_d - f_{\text{ext}}$  where  $f_d$  and  $f_{\text{ext}}$  denote the desired force and contact force for one direction. Then, let  $\hat{K}_e$  be an element of  $\hat{\mathbf{K}}_e$ , thus,  $V_F = \frac{1}{2} e_f^2$  and the corresponding gradient of  $V_F$  is

$$\frac{\partial V_F}{\partial \hat{K}_e} = e_f \cdot \frac{\partial e_f}{\partial \hat{K}_e}. \quad (35)$$

Thus, the update of  $\hat{K}_e$  is chosen as follows by the gradient method

$$\Delta \hat{K}_e = -\eta_{K_e} \frac{\partial V_F}{\partial \hat{K}_e} = -\eta_{K_e} \cdot e_f \cdot \frac{\partial e_f}{\partial \hat{K}_e} \quad (36)$$

where  $\eta_{K_e}$  is the learning rate. For the training process, the learning rate plays an important role for this case. A small value of  $\eta_{K_e}$  leads a slower convergence and a large value may have unstable result. Hence, the selection of  $\eta_{K_e}$  is important. Herein, the Lyapunov approach is adopted to guarantee the convergence of  $\hat{K}_e$ . We then have the convergence theorem.

**Theorem 2** Consider the force tracking control problem as above description and let  $\eta_{K_e}$  be the learning rate of the estimation of stiffness. The force tracking can be achieved if the learning rate  $\eta_{K_e}$  is selected satisfies

$$0 < \frac{\eta_{K_e}}{2} \left( \frac{\partial e_f}{\partial \hat{K}_e} \right)^2 < 1. \quad (37)$$



*Proof* Rewrite (34) as

$$V_F(k) = \frac{1}{2}e_f^2(k) = \frac{1}{2}(f_d(k) - f_{\text{ext}}(k))^2 \quad (38)$$

where  $e_f(k) = f_d(k) - f_{\text{ext}}(k)$  and  $k$  is time-instant for parameter update. Let  $\Delta e_f(k) = e_f(k+1) - e_f(k)$  and then we have

$$\begin{aligned} \Delta V_F(k) &= V_F(k+1) - V_F(k) \\ &= \frac{1}{2}[e_f(k+1) + e_f(k)][e_f(k+1) - e_f(k)] \\ &= \frac{1}{2}[2e_f(k) + \Delta e_f(k)] \cdot \Delta e_f(k) \\ &= \frac{\Delta e_f^2(k)}{2} + e_f(k) \cdot \Delta e_f(k). \end{aligned} \quad (39)$$

By the gradient result,  $\Delta e_f(k) \approx \frac{\partial e_f}{\partial \hat{K}_e} \Delta \hat{K}_e$  and  $\Delta \hat{K}_e = -\eta_{K_e} \left( \frac{\partial V_F}{\partial \hat{K}_e} \right)$ , we obtain

$$\Delta e_f(k) = \frac{\partial e_f}{\partial \hat{K}_e} \left( -\eta_{K_e} \frac{\partial V_F}{\partial \hat{K}_e} \right) = -\eta_{K_e} e_f \left( \frac{\partial e_f}{\partial \hat{K}_e} \right)^2 \quad (40)$$

and

$$\begin{aligned} \Delta V(k) &= \frac{\eta_{K_e}^2 e_f^2}{2} \left( \frac{\partial e_f}{\partial \hat{K}_e} \right)^4 - \eta_{K_e} e_f^2 \left( \frac{\partial e_f}{\partial \hat{K}_e} \right)^2 \\ &= -\eta_{K_e} e_f^2 \left( \frac{\partial e_f}{\partial \hat{K}_e} \right)^2 \\ &\quad \cdot \left[ 1 - \frac{1}{2} \eta_{K_e} \left( \frac{\partial e_f}{\partial \hat{K}_e} \right)^2 \right]. \end{aligned} \quad (41)$$

As above, we can obtain the stability condition for selection of learning rate  $\eta_{K_e}$ , condition (37), this implies  $\Delta V(k) < 0$ ,  $\forall k > 0$ , this means that the force error will approach to zero. In addition, the convergence of stiffness coefficient can be guaranteed, i.e., actual value of  $K_e$  can be obtained. This completes the proof.  $\square$

After estimating the coefficient  $\mathbf{K}_e$ , we remain to find the desired trajectory  $\mathbf{x}^*$  such that  $\mathbf{F}_{\text{ext}} = \mathbf{F}_d$ . In addition, to accomplish force control, we should design the suitable  $\mathbf{x}$  for contact space. From (28) and estimation of  $\mathbf{K}_e$ , we have the variation of desired position

$$\Delta \mathbf{x} = \hat{\mathbf{K}}_e^{-1} \Delta \mathbf{F} \quad (42)$$

where  $\Delta \mathbf{x} = \mathbf{x}_{ss} - \mathbf{x}$  and  $\Delta \mathbf{F} = \mathbf{F}_d - \mathbf{F}_{\text{ext}}$ , then we can design the ideal trajectory to achieve force control as follows iteratively

$$\mathbf{x}^*(k+1) = \mathbf{x}(k) + \Delta \mathbf{x}(k). \quad (43)$$

According equation (30), in considering the steady state, we obtain the new desired trajectory for the contact case

$$\mathbf{x}_d^*(k+1) = \mathbf{K}_m^{-1} \left[ (\mathbf{K}_m + \hat{\mathbf{K}}_e) \mathbf{x}^*(k+1) - \hat{\mathbf{K}}_e \mathbf{x}_e \right]. \quad (44)$$

As the results of Theorem 1, the end-effector of robot will follows the desired trajectory, thus, the above discussion guarantees the convergence of force tracking error even the stiffness gain  $K_e$  is unknown, i.e.,  $\mathbf{F}_{\text{ext}}$  approaches to  $\mathbf{F}_d$ .

#### 4 Simulation results

Herein, the proposed control approach is applied on the impedance force tracking control of a robot manipulator system with two rigid links and two rigid revolute joints. For  $i = 1, 2$ ,  $m_i$ ,  $l_i$ ,  $l_{ci}$ , and  $I_i$  are mass, length, gravity center length and inertia of link  $i$ , respectively. The actual values of quantities are  $m_1 = 2$  kg,  $m_2 = 1$  kg,  $l_1 = l_2 = 0.75$  m,  $l_{c1} = l_{c2} = 0.375$  m,  $I_1 = 0.09375$  kg m<sup>2</sup>,  $I_2 = 0.046975$  kg m<sup>2</sup> [40]. The inertia matrix  $\mathbf{D}(\mathbf{q})$  is defined as

$$\mathbf{D}(\mathbf{q}) = \begin{bmatrix} d_{11} & d_{12} \\ d_{21} & d_{22} \end{bmatrix} \quad (45)$$

where  $d_{11} = m_1 l_{c1}^2 + m_2 l_1^2 + I_1 + m_2 l_{c2}^2 + I_2 + 2m_2 l_1 l_{c2} \cos q_2$ ,  $d_{12} = d_{21} = m_2 l_{c2}^2 + I_2 + m_2 l_1 l_{c2} \cos q_2$ , and  $d_{22} = m_2 l_{c2}^2 + I_2$ .  $\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}})$  and  $\mathbf{g}(\mathbf{q})$  are defined as

$$\mathbf{C}(\mathbf{q}, \dot{\mathbf{q}}) = \begin{bmatrix} -m_2 l_1 l_{c2} \dot{q}_2 \sin q_2 & -m_2 l_1 l_{c2} (\dot{q}_1 + \dot{q}_2) \sin q_2 \\ m_2 l_1 l_{c2} \dot{q}_1 \sin q_2 & 0 \end{bmatrix}, \quad (46a)$$

$$\mathbf{g}(\mathbf{q}) = \begin{bmatrix} (m_1 l_{c2} + m_2 l_1) g \cos q_1 + m_2 l_{c2} g \cos(q_1 + q_2) \\ m_2 l_{c2} g \cos(q_1 + q_2) \end{bmatrix}. \quad (46b)$$

The Jacobian matrix is

$$\mathbf{J}(\mathbf{q}) = \begin{bmatrix} -m_2 l_1 l_{c2} \dot{q}_2 \sin q_2 & -m_2 l_1 l_{c2} (\dot{q}_1 + \dot{q}_2) \sin q_2 \\ m_2 l_1 l_{c2} \dot{q}_1 \sin q_2 & 0 \end{bmatrix}. \quad (47)$$

The initial condition for the end-effector and reference state are  $\mathbf{x}(0) = [0.75 \ 0.75 \ 0 \ 0]^T$  and  $\mathbf{x}_m(0) = [0.8 \ 0.8 \ 0 \ 0]^T$ . The desired trajectory  $\mathbf{x}_d$  is a 0.2 m-radius circle centered at (0.8, 1.0 m) in 10 s as follows

$$\begin{aligned} x_d(t) &= 0.8 + 0.2 \sin \frac{2\pi}{10} t \\ y_d(t) &= -0.2 \cos \frac{2\pi}{10} t. \end{aligned} \quad (48)$$

As above discussion, system parameter matrices  $\mathbf{D}$ ,  $\mathbf{C}$ , and  $\mathbf{g}$  are assumed to be not exactly known and are approximated by the FNNs. Initial weighting vectors of FNN are chosen as to be zero to avoid initial larger control effort and the fuzzy rule number are chosen as seven for  $\hat{\mathbf{D}}_x$ ,  $\hat{\mathbf{C}}_x$ , and  $\hat{\mathbf{g}}_x$ . The update laws gain matrices in (15) are chosen with

$$\begin{aligned} \mathbf{Q}_D^{-1} &= \mathbf{Q}_C^{-1} = \text{diag}(q_1, \dots, q_7), \\ q_i &= 1, i = 1, \dots, 28 \end{aligned} \quad (49a)$$

$$\begin{aligned} \mathbf{Q}_g^{-1} &= \text{diag}(q_1, \dots, q_7), q_i = 100, \\ i &= 1, \dots, 28. \end{aligned} \quad (49b)$$

According to the approximation theorem of FNNs, we assume  $\mathbf{e}_1$  cannot be eliminated and there exists a positive constant  $\delta$  such that  $\|\mathbf{e}_1\| \leq \delta$  for all  $t \geq 0$ . Herein,  $\delta$  is selected to be one. The constraint surface is a flat wall with a triangle crack, and the environment can be modeled as linear spring  $f_{\text{ext}} = k_w(x - x_w)$ .  $f_{\text{ext}}$  is the force acting on the surface,  $k_w = 5000$  N/m is the environment stiffness,  $x$  is the coordinate of the end-effector in the  $x$ -direction, and  $x_w = 0.95$  m is the position of the surface. Therefore, the external force vector in equation (2) becomes  $\mathbf{F}_{\text{ext}} = [f_{\text{ext}} \ 0]^T$ . The controller gains are selected as

$$K_d = \begin{bmatrix} 100 & 0 \\ 0 & 100 \end{bmatrix}, \quad \Lambda = \begin{bmatrix} 10 & 0 \\ 0 & 10 \end{bmatrix}. \quad (50)$$

Herein, the gain matrices in the target impedance are described in [5]. Several illustrated examples are introduced to demonstrate the performance and effectiveness of the proposed adaptive control approach.

#### 4.1 Adaptive tracking and force control

In this example, the desired force is selected as 3 Nt. Figure 4 shows the trajectory tracking performance in the Cartesian space (solid: robot position; dashed: environment trajectory; dash-dotted: estimated environment trajectory). We can find that the end point of robot contact the constraint surface is a flat wall with a triangle crack and track the reference trajectory in free space. For the force control, herein, we test the different learning rates  $\eta = 10000, 30000$  and  $60000$  to have the comparison results in Fig. 5. Obviously, the

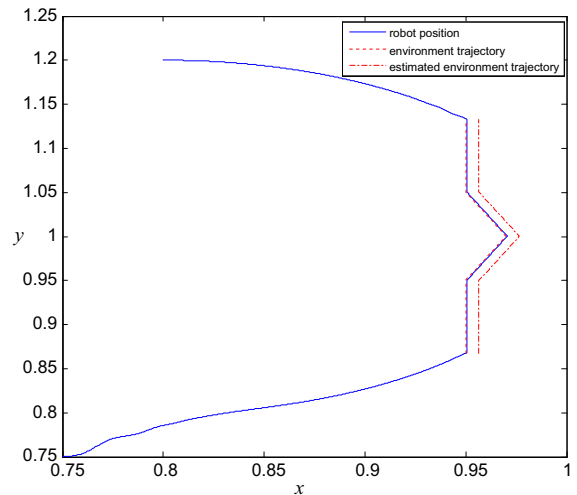


Fig. 4 Position tracking performance result at  $t = 0-5$  s

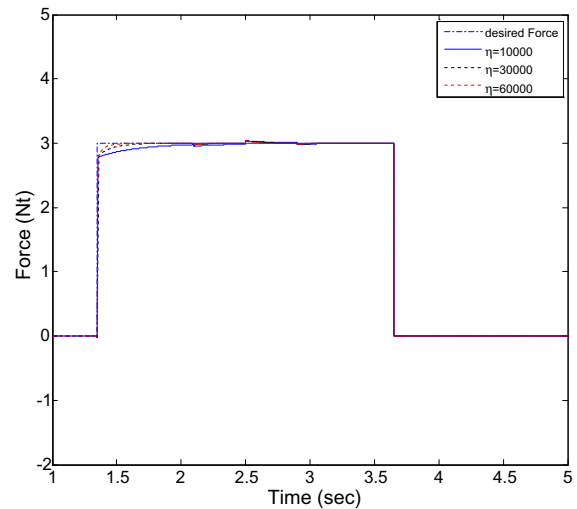
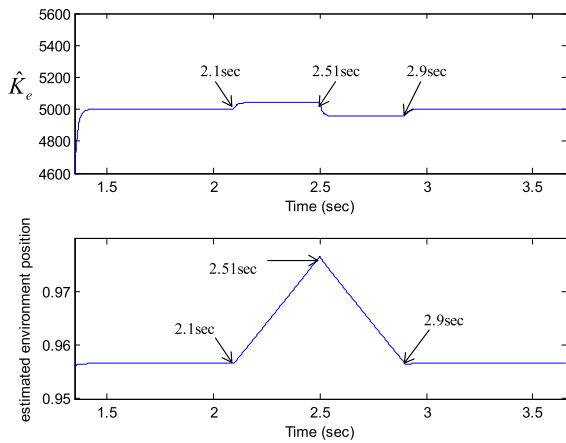


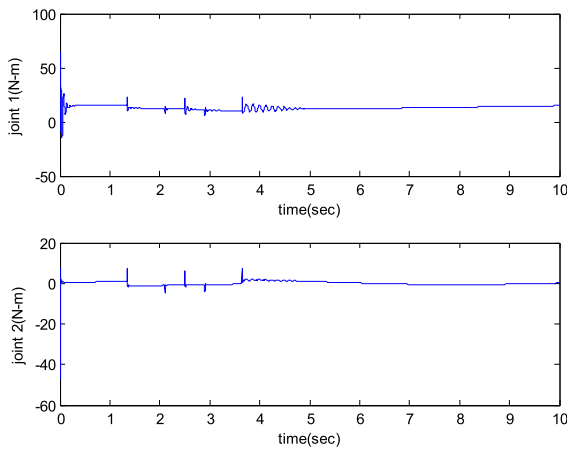
Fig. 5 Force tracking performance with different learning rates

larger value of  $\eta$  performs quickly. Figure 6 shows the estimated stiffness  $\hat{\mathbf{K}}_e$  and environment position. One can observe that the variations of the estimated  $\hat{\mathbf{K}}_e$  and  $x^*$  are changed at 1.35 s, 2.1 s, 2.51 s, 2.9 s due to the end-effector touches the wall and triangle crack. From the observation of Fig. 6, we can find that the estimated stiffness coefficient is convergent and the estimated environment position is contained to achieve the adaptive force control. Figure 7 shows the control effort at  $t = 1-10$  s. We can observe that the impulse control torque is obtained due to the change of environment. In addition, the fast convergence of tracking error and





**Fig. 6** Stiffness and estimated environment position

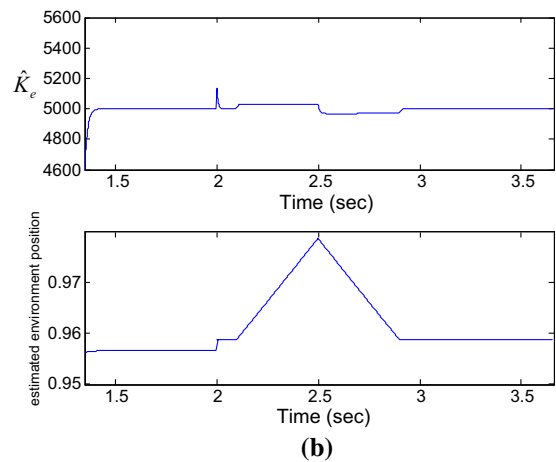
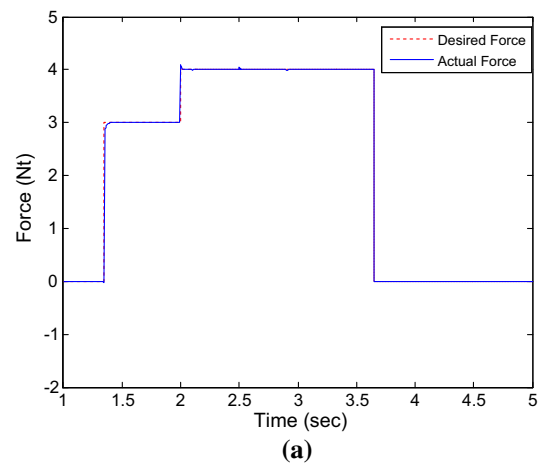


**Fig. 7** Control effort of example 1

control efforts also shows the performance and effectiveness of our approach.

#### 4.2 Variation of tracking force

From the observation of Fig. 5, we can find that the estimated stiffness coefficient is convergent and the estimated environment position is obtained to achieve the adaptive force control. Herein, simulation results of variant tracking force example are introduced. Figure 8a introduces the force variation (dashed: desired force; solid: actual force). It can be observed that the contact force follows the desired force even the environment surface is changed. Figure 8b shows the estimated stiffness  $\hat{K}_e$  and environment position. One can find that the adaptive force tracking control is achieved



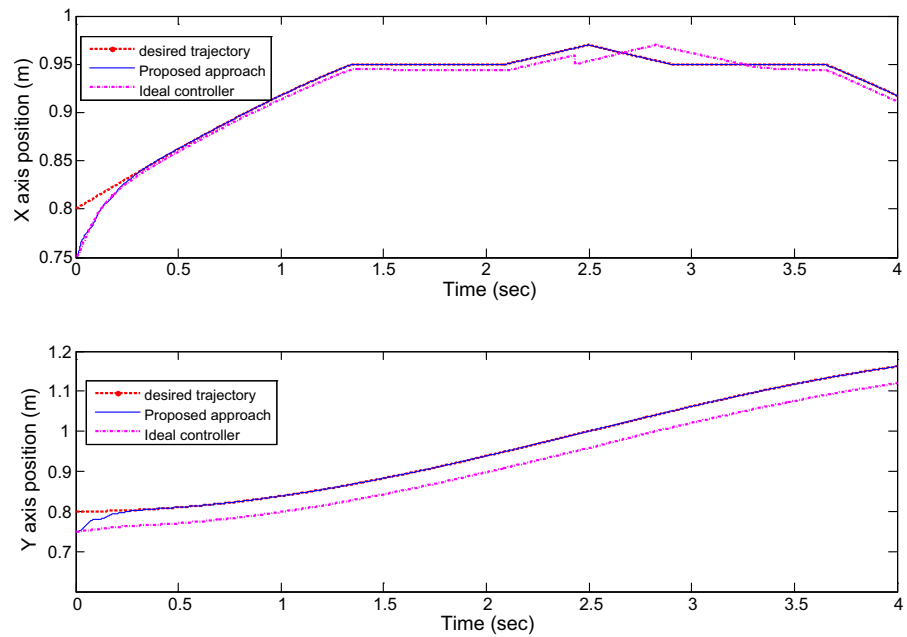
**Fig. 8** Simulation results of variant tracking force, **a** force tracking performance with different force, **b** estimated stiffness and estimated environment position

by our proposed approach and the estimated  $x^*$  is also obtained such that  $f_{\text{ext}}$  approaches to  $f_d$ . From this illustrated examples, we can observe that the proposed adaptive approach performs well (the estimated  $\hat{K}_e$  and  $x^*$  can vary at the same time due to the change of environment and desired tracking force).

#### 4.3 Robustness illustration

Herein, we introduce the robustness of the proposed adaptive approach and the necessity of the estimation of robot model ( $\mathbf{D}_x$ ,  $\mathbf{C}_x$ ,  $\mathbf{g}_x$ ) for uncertain systems. In order to illustrate the effectiveness of our approach for system with uncertainties, we assumed the nominal and actual masses of link two are 1 and 5 kg, respectively. The

**Fig. 9** The controller performance with uncertainties as mass = 5 kg of link 2 (*dashed* desired trajectory; *dotted* our proposed approach, and *dash-dotted* ideal controller)



comparison results between our approach and the ideal impedance controller are presented, where the ideal controller is derived by nominal model ( $\mathbf{D}_x$ ,  $\mathbf{C}_x$ ,  $\mathbf{g}_x$  are given), i.e.,  $\hat{\mathbf{D}}_x$ ,  $\hat{\mathbf{C}}_x$ ,  $\hat{\mathbf{g}}_x$  in equation (15) are replaced by  $\mathbf{D}_x$ ,  $\mathbf{C}_x$ ,  $\mathbf{g}_x$ . Figure 9 shows the tracking trajectories of  $(x, y)$  space, dashed: desired trajectory; dotted: our proposed approach, and dash-dotted: ideal controller. From Fig. 9, we can observe that the proposed approach perform well with smaller tracking error than the ideal controller. This illustrates the effectiveness of the proposed approach.

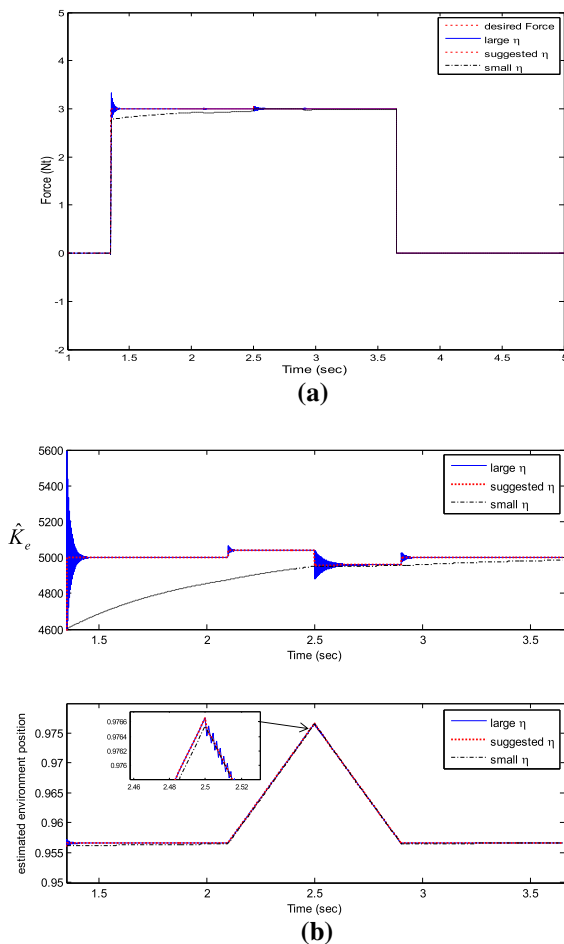
#### 4.4 Discussion of selecting the learning rate $\eta_{K_e}$

As above description, choosing the suitable learning rate is important for force tracking control. A small value of  $\eta_{K_e}$  results slower convergence and a large one leads higher amplitude of transient response or unstable result. Herein, the discussion of learning rate selection is introduced, in which these selections satisfy the convergent condition (37). Figure 10a, b show the comparison results of force tracking control, the estimated stiffness and desired position, respectively (solid-line: large  $\eta_{K_e}$ ; dotted-line: suggested  $\eta_{K_e}$ ; dash-dotted: small  $\eta_{K_e}$ ). The adopted large  $\eta_{K_e}$ , suggested  $\eta_{K_e}$ , and small  $\eta_{K_e}$  are selected such that  $\lambda = 0.8$ ;  $\lambda = 0.45$ ; and  $\lambda = 0.001$ , respectively, where  $\lambda \equiv \frac{\eta}{2} \left( \frac{\partial e_f}{\partial \hat{K}_e} \right)^2$ . That

is, the suggested learning rate is  $\eta_{K_e} = \frac{9}{10 \left( \frac{\partial e_f}{\partial \hat{K}_e} \right)^2}$  as our experience. From Fig. 10, large  $\eta_{K_e}$  results faster response in force tracking control, but higher transient amplitude of force and  $\hat{K}_e$  are obtained. In addition, this results the larger tracking error when the environment is changed (shown in Fig. 10b). On the other hand, small  $\eta_{K_e}$  has slow convergence. It can be found that the suggested learning rate  $\eta_{K_e} = \frac{9}{10 \left( \frac{\partial e_f}{\partial \hat{K}_e} \right)^2}$  preserves the stability and good performance in force tracking and estimation of stiffness and desired position.

## 5 Conclusion

In this paper, we have proposed the adaptive tracking force control scheme using FNNs for an  $n$ -link robot manipulator under unknown task environment. The unknown system parameter matrices are estimated by the FNNs to develop the tracking control scheme, and the adaptive force control laws are established by gradient method. Based on the Lyapunov stability analysis, we guarantee the stability of the closed-loop system and the convergence of parameters. In addition, the update laws of FNNs and estimation coefficient are designed. Finally, simulation results of a two-link robot manipulator with environment constraint are



**Fig. 10** Discussion of selecting learning rate, **a** force tracking performance, **b** stiffness, estimated environment position

introduced to illustrate the performance and effectiveness of our approach.

**Acknowledgments** The authors would like to thank anonymous reviewers for their insightful comments and valuable suggestions. This work was supported in part by the Ministry of Science and Technology, Taiwan, R.O.C., under contracts MOST-102-2221-E-005-061-MY3 and MOST-103-2218-E-005-005-MY2.

## References

1. Abdallah, A., Dawson, D., Dorato, P., Jamishidi, M.: Survey of robust control for rigid robots. *IEEE Control Syst. Mag.* **11**(2), 24–30 (1991)
2. Asada, H., Slotine, J.-J.E.: *Robot-Analysis and Control*. Wiley, New York (1986)

3. Cai, D., Dai, Y.: A globally convergent robust controller for robot manipulator. In: *Proceedings of IEEE international conference on control applications*, pp. 328–332, (2001)
4. Cheah, C.C., Liu, C., Slotine, J.J.E.: Adaptive tracking control for robots with unknown kinematic and dynamic properties. *Int. J. Robot. Res.* **25**(3), 283–296 (2006)
5. Chien, M.C., Huang, A.C.: Adaptive impedance control of robot manipulator based on function approximation technique. *Robotica* **22**, 395–403 (2004)
6. Chiu, C.S., Lian, K.Y., Wu, T.C.: Robust adaptive motion/force tracking control design for uncertain constrained robot manipulators. *Automatica* **40**(12), 2111–2119 (2004)
7. Fanaei, A., Farrokhi, M.: Robust adaptive neuro-fuzzy controller for hybrid position/force control of robot manipulator in contact with unknown environment. *J. Intell. Fuzzy Syst.* **17**(2), 125–144 (2006)
8. Fateh, M.M., Khorashadizadeh, S.: Robust control of electrically driven robots by adaptive fuzzy estimation of uncertainty. *Nonlinear Dyn.* **69**(3), 1465–1477 (2012)
9. Fateh, M.M.: Nonlinear control of electrical flexible-joint robots. *Nonlinear Dyn.* **67**(4), 2549–2559 (2012)
10. Filaretov, V. F., Zuev, A. V.: Adaptive force/position control of robot manipulators. In: *2008 IEEE/ASME international conference on advanced intelligent mechatronics*, pp. 96–101, July 2–5, Xi'an, China (2008)
11. Hogan, N.: Impedance control: an approach to manipulation: part 1—theory, part 2—implementation, part 3—an approach to manipulation. *ASME J. Dyn. Syst. Meas. Control* **107**, 1–24 (1985)
12. Huang, A.C., Chien, M.C.: *Adaptive control of robot manipulators: a unified regressor-free approach*. World Scientific Publishing Co., Pte. Ltd, Singapore (2010)
13. Huang, A.C., Kuo, Y.S.: Sliding control of nonlinear systems containing time varying uncertainties with unknown bound. *Int. J. Control* **74**(3), 252–264 (2001)
14. Huang, A.C., Chen, Y.C.: Adaptive sliding control for single link flexible-joint robot with mismatched uncertainties. *IEEE Trans. Control Syst. Technol.* **12**(5), 770–775 (2004)
15. Jafari, A., Monfaredi, R., Rezaei, R., Talebi, M., Ghidary, S. S.: Sliding mode hybrid impedance control of robot manipulators interacting with unknown environments using VSMRC method. In: *ASME 2012 international mechanical engineering congress and exposition*, pp. 1071–1081, (2012)
16. Jung, S., Hsia, T.C., Bonitz, R.G.: Force tracking impedance control of robot manipulators under unknown environment. *IEEE Trans. Control Syst. Technol.* **12**(2), 474–483 (2004)
17. Jung, S., Hsia, T.C.: Neural network impedance force control of robot manipulator. *IEEE Trans. Ind. Electron.* **45**(3), 451–461 (1998)
18. Jung, S., Yim, S.B., Hsia, T. C.: Experimental studies of neural impedance force control of robot manipulators. In: *Proceedings of IEEE conference on robotics and automations*, pp. 3453–3458, Seoul, Korea (2001)
19. Jung, S., Hsia, T. C.: Reference compensation technique of neural force tracking impedance control for robot manipulators. In: *World congress on intelligent control and automation*, Jinan, China, July 6–9, (2010)
20. Khatib, O.: A unified approach for motion and force control of robot manipulators: the operational space formulation. *IEEE J. Robot. Autom.* **RA-3**(1), 43–53 (1987)

21. Lee, C.H., Teng, C.C.: Identification and control of dynamic systems using recurrent fuzzy neural network. *IEEE Trans. Fuzzy Syst.* **8**(4), 349–366 (2000)
22. Lee, C.H., Chung, B.R.: FNN-based disturbance observer controller design for nonlinear uncertain systems. *WSEAS Trans. Syst. Control* **2**(3), 255–260 (2007)
23. Lee, C.H., Chung, B.R.: Adaptive backstepping controller design for nonlinear uncertain systems using fuzzy neural systems. *Int. J. Syst. Sci.* **43**(10), 1855–1869 (2012)
24. Lee, C.H., Chang, F.Y., Lin, C.M.: An efficient interval type-2 fuzzy CMAC for chaos time-series prediction and synchronization. *IEEE Trans. Cybern.* **44**(3), 329–341 (2014)
25. Lee, C.H., Lee, Y.C.: Nonlinear systems design by a novel fuzzy neural system via hybridization of EM and PSO algorithms. *Inf. Sci.* **186**(1), 59–81 (2012)
26. Lewis, F.L., Dawson, D.M., Abdallah, C.T.: *Robot Manipulator Control Theory and Practice*, Second Edition, Revised and Expanded. Marcel Dekker, New York (2004)
27. Li, Y., Ge, S.S., Zhang, Q., Lee, T.H.: Neural networks impedance control of robot interacting with environment. *IET Control Theory Appl.* **7**(11), 1509–1519 (2013)
28. Li, Z., Ge, S.S., Adams, M., Wijesoma, W.S.: Robust adaptive control of uncertain force/motion constrained non-holonomic mobile manipulators. *Automatica* **44**, 776–784 (2008)
29. Li, Z., Chen, W.: Adaptive neural-fuzzy control of uncertain constrained multiple coordinated nonholonomic mobile manipulators. *Eng. Appl. Artif. Intell.* **21**(7), 98–1000 (2008)
30. Li, Z., Chen, W., Luo, J.: Adaptive compliant force-motion control of coordinated non-holonomic mobile manipulators interacting with unknown non-rigid environments. *Neurocomputing* **71**(7–9), 1330–1334 (2008)
31. Ortega, R., Spong, M.W.: Adaptive motion feedback control of rigid robot: a tutorial. In: *Proceedings of 27th conference on decision and control*, pp. 1575–1584, (1988)
32. Raibert, M., Craig, J.J.: Hybrid position/force control of manipulators. *ASME J. Dyn. Syst. Meas. Control* **102**, 126–133 (1981)
33. Ren, T.J.: A robust impedance control using recurrent fuzzy neural networks. In: *The 3rd international conference on innovative computing information and control (ICICIC'08)*, (2008)
34. Roy, J., Whitcomb, L.L.: Adaptive force control of position/velocity controlled robots: theory and experiment. *IEEE Trans. Robot. Autom.* **18**(2), 121–137 (2002)
35. Sedegh, N., Horowitz, R.: Stability and robustness analysis of a class of adaptive controller for robotic manipulators. *Int. J. Robot. Res.* **9**(3), 74–92 (1990)
36. Singh, S., Popa, D.: An analysis of some fundamental problems in adaptive control of force and impedance behavior: theory and experiments. *IEEE Trans. Robot. Autom.* **11**, 912–921 (1995)
37. Slotine, J.J.E., Li, W.: *Applied Nonlinear Control*, Englewood Cliffs. Prentice-Hall, NJ (1991)
38. Sun, D., Mills, J.K.: Performance improvement of industrial robot trajectory tracking using adaptive-learning scheme. *ASME J. Dyn. Syst. Meas. Control* **121**(2), 285–292 (1999)
39. Tanaka, Y., Tsuji, T.: On-line learning of robot arm impedance using neural network. In: *Proceedings of IEEE international conference on robotic and biomimetics*, Shenyang, China, August 22–26, (2004)
40. Wu, S.C.: *Adaptive Controller Design for Robot Manipulators Based on Function Approximation Technique*, Master Thesis, National Taiwan University of Science and Technology, Taiwan, July (2002)