

# Robust Adaptive Neural-Fuzzy Network Tracking Control for Robot Manipulator

T. Ngo, Y. Wang, T.L. Mai, M.H. Nguyen, J. Chen

**ThanhQuyen Ngo, YaoNan Wang, T. Long Mai,  
M. Hung Nguyen, Jun Chen**

College of Electrical and Information Engineering  
Hunan University, Changsha, Hunan Province 410082, P.R.China  
Faculty of Electrical Engineering  
HCM City University of Industry, HCM City, Vietnam  
thanhquyenngo2000@yahoo.com, yaonan@hnu.cn,  
mailongtk@gmail.com, manhnhung@yahoo.com, 2011junchen@gmail.com

**Abstract:** This paper presents a robust adaptive neural-fuzzy network control (RANFNC) system for an n-link robot manipulator to achieve the high-precision position tracking. Initially, the model dynamic of an n-link robot manipulator is introduced. However, it is difficult to design a conformable model-based control scheme, for instance, external disturbances, friction forces and parameter variations. In order to deal with this problem, the RANFNC system is investigated to the joint position control of an n-link robot manipulator. In this control scheme, a four-layer neural-fuzzy-network (NFN) is used for the main role, and the adaptive tuning laws of network parameters are derived in the sense of a projection algorithm and the Lyapunov stability theorem to ensure network convergence as well as stable control performance. The merits of this model-free control scheme are that not only the stable position tracking performance can be guaranteed but also unknown system information and auxiliary control design are required in the control process. The simulation results are provided to verify the effectiveness of the proposed RANFNC methodology.

**Keywords:** Adaptive control, Neural-fuzzy network, robot manipulator.

## 1 Introduction

In the past decade, the applications of intelligent control techniques (fuzzy control or neural control) to the motion control of robotic manipulator have received considerable attention [1]-[3]. In general, robotic manipulators have to face various uncertainties in their dynamics, such as friction, and external disturbance. It is difficult to establish exactly mathematical model for the design of a model-based control system. Thus, the general claim of these intelligent control approaches is that they can reduce the effects of structured parametric uncertainty and the unstructured disturbance by using their powerful learning ability without a detailed knowledge of the controlled plant in the design processes.

Recently, the concept of incorporating fuzzy logic into a neural network has grown into a popular research topic [5]-[8]. The integrated neural-fuzzy-network system possesses the merits of both fuzzy system [9] (e.g., humanlike IF-THEN rule thinking and ease of incorporating expert knowledge) and neural network [10] (e.g., learning and optimization abilities and connectionist structure). In this way, one can bring the low-level learning and computational power of neural network into fuzzy systems as well as the high-level humanlike IF-THEN rule thinking and reasoning of fuzzy systems into neural network.

The main of this paper is to design an intelligent control system scheme for the position control of an n-link robot manipulator by using neural-fuzzy-network controller to compensated uncertainty dynamic model and external disturbance via capability self-learning of neural network and human intuitive.

This paper is organized as follows: Section 2 described a dynamic model of an n-link robot manipulator briefly [11]. Section 3 presents a structure of neural-fuzzy-network. The design process of the RANFNC system is investigated to control an n-link robot manipulator for periodic motion in section 4. The design procedures of the proposed RANFNC system are described in detail. The adaptive learning laws in the RANFNC system are designed in the sense of the Lyapunov stability theorem [12], [13] so that the network convergence and system tracking stability can be guaranteed in the closed-loop control system. Numerical simulation results of a two-link robot manipulator under the possible occurrence of uncertainties are provided to demonstrate the tracking control performance of the proposed RANFNC system in section 5. Conclusions are drawn in section 6.

## 2 System Description

### 2.1 Robotic Dynamic Model

In general, the dynamic of an n-link robot manipulator may be expressed in [11] as:

$$M(q)\ddot{q} + C(q, \dot{q})\dot{q} + G(q) + F(\dot{q}) + \tau_d = \tau \tag{1}$$

Where  $q, \dot{q}, \ddot{q} \in \mathbb{R}^n$  are the joint position, velocity and acceleration vectors,  $M(q) \in \mathbb{R}^{n \times n}$  denotes the inertia matrix,  $C(q, \dot{q}) \in \mathbb{R}^{n \times n}$  expresses the matrix of centripetal and Coriolis forces,  $G(q) \in \mathbb{R}^n$  is the gravity vector, and  $F(\dot{q}) \in \mathbb{R}^n$  is the friction. Bounded unknown disturbances are denoted by  $\tau_d$  and the control input torque is  $\tau(t) \in \mathbb{R}^n$ . In this paper a robot manipulator is shown in Fig.1 which is utilized to verify dynamic properties are given in section 5.

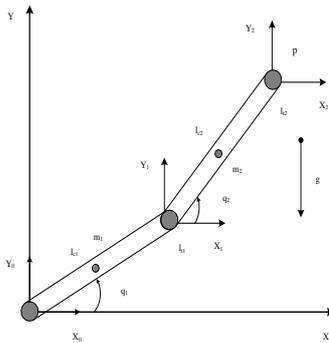


Figure 1: Architecture of two-link robot manipulator.

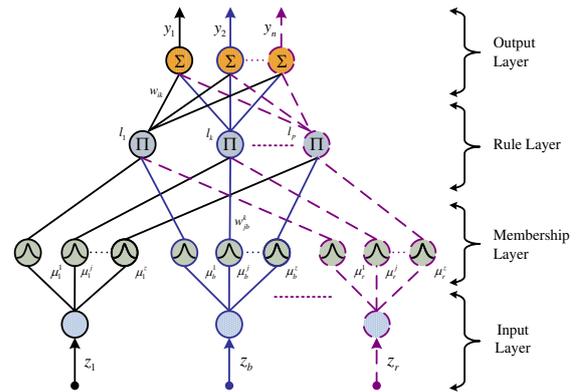


Figure 2: Structure of four-layer NFN

Given a desired arm trajectory  $q_d(t) \in \mathbb{R}^n$  the tracking error is:

$$e(t) = q_d(t) - q(t) \tag{2}$$

And the filtered tracking error (in standard use in robotics) is:

$$r(t) = \dot{e} + \lambda e \tag{3}$$

Where  $\lambda = \lambda^T > 0$ , differentiating  $r(t)$  and using (1), the arm dynamics may be written in terms of the filtered tracking error as:

$$M(q)\dot{r} = -C(q, \dot{q})r - \tau + M(q)(\dot{q}_d + \lambda\dot{e}) + C(q, \dot{q})(\dot{q} + \lambda e) + F(\dot{q}) + G(q) + \tau_d \quad (4)$$

Where the nonlinear robot function is:

$$f(x) = M(q)(\ddot{q}_d + \lambda\dot{e}) + C(q, \dot{q})(\dot{q} + \lambda e) + G(q) + F(\dot{q}) \quad (5)$$

Substituting (5) equation into (4) we have:

$$M(q)\dot{r} = -C(q, \dot{q})r - \tau + f(x) + \tau_d \quad (6)$$

And, for instance  $x = [e^T \quad \dot{e}^T \quad q^T \quad \dot{q}^T \quad \ddot{q}^T]^T$

## 2.2 Defined Control Law

Now, we define a control input torque as:

$$\tau_0 = \hat{f}(x) + K_v r \quad (7)$$

With  $\hat{f}(x)$  is an estimate of  $f(x)$  and a gain matrix  $K_v = K_v^T$  the closed-loop system becomes:

$$M(q)\dot{r} = -(K_v + C(q, \dot{q}))r + \tilde{f}(x) + \tau_d \equiv -(K_v + C(q, \dot{q}))r + \zeta_0, \quad \zeta_0 = \tilde{f}(x) + \tau_d \quad (8)$$

Where functional estimate error is given by:

$$\tilde{f}(x) = f(x) - \hat{f}(x) \quad (9)$$

This is a system error wherein filtered tracking error is driven functional estimate error.

The control  $\tau_0$  incorporates a proportional plus derivative (PD) term in  $K_v r = K_v(\dot{e} + \lambda e)$ .

In the remainder of the paper we shall use (8) to focus on selecting NFN weight tuning algorithms that guarantee the stability of the filtered tracking error  $r(t)$ . Then, since (3), with the input considered as  $r(t)$  and the output as  $e(t)$  describes a stable system, standard techniques [13], [15] guarantee that  $e(t)$  exhibits stable behaviour. In fact  $\|e\|_2 \leq \|r\|_2 / \sigma_{min}$ ,  $\|\dot{e}\|_2 \leq \|r\|_2$  with  $\sigma_{min}(\lambda)$  the minimum singular value of  $\lambda$ . Generally  $\lambda$  is diagonal, so that  $\sigma_{min}(\lambda)$  is the smallest element of  $\lambda$ .

The following properties of the robot dynamics are required [15]. They hold for all revolute rigid-link manipulator.

The inertial matrix  $M(q)$  is symmetric and positive definite. It is also bounded as a function of  $q : m_1 I \leq m_2 I$ .  $\dot{M}(q) - 2C(q, \dot{q})$  is a skew symmetric matrix, that is  $y^T [\dot{M}(q) - 2C(q, \dot{q})] y$ , where  $y$  is a  $n \times 1$  nonzero vector. The gravity vector  $G(q)$  is bounded as a function of  $q : G(q) \leq g_d$ . The unknown disturbance satisfies  $\|\tau_d\| \leq d_b$  and this property is new, that is, the dynamic (8) from  $\zeta_0(t)$  to  $r(t)$  are a state strict passive system.

## 3 Structure of NFN

In the RANFNC scheme, as shown in Fig.3, an NFN estimator is designed to tune the nonlinear dynamic function vector, and then, the estimative vector is used to indirectly develop a stable RANFNC law. An NFN controller is directly designed to imitate a predetermined model-based stabilizing control law, and then, the stable control performance can be achieved by using joint position and filtered error vector and information. In this paper, a four-layer NFN

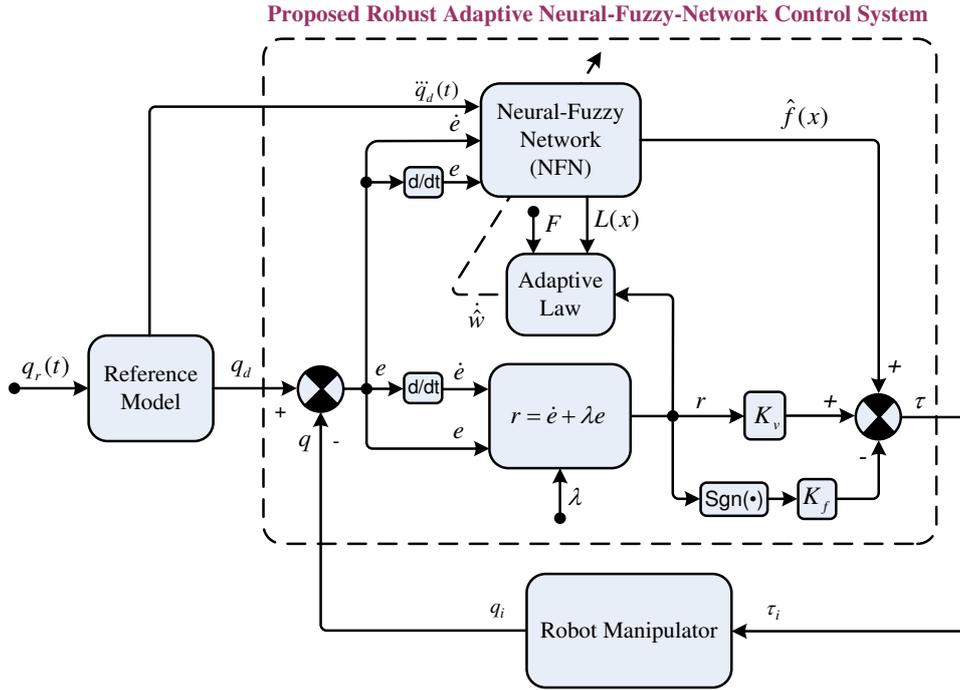


Figure 3: Block diagram of RANFNC scheme

structure shows in Fig.2, which is composed of input, membership, rule, and output layers, is adopted to implement the NFN estimate in RANFNC. The signal propagation and the basic function in each layer of the NFN are introduced as follows.

1. Input layer transmits the input linguistic variables  $z_b|_{1,2,\dots,r}$  to the next layer.
2. Membership layer represents the input values with the following Gaussian membership functions:

$$\mu_b^j(z_b) = \exp \left[ -\frac{(z_b - m_b^j)^2}{(t_b^j)^2} \right] \tag{10}$$

Where  $\exp[\cdot]$  is the exponential function,  $m_b^j$  and  $t_b^j$  ( $b = 1, 2, \dots, r; j = 1, 2, \dots, z$ ) are the mean and the standard deviation of the Gaussian function in the  $j$ th term of the  $b$ th input variable  $z_b$  to the node of membership layer, respectively. It can be referred as the fuzzification procedure.

3. The output of each node in the rule layer is determined by fuzzy *And* operation. Each node in this rule layer is denoted by  $\prod$ , which multiplies the input signals and output the result of the product. The product operation is utilized to determine the firing strength. It can be referred as the fuzzy inference mechanism. The output of this layer is given as:

$$l_k = \prod_{b=1}^r w_{jb}^k \mu_b^j(z_b) \tag{11}$$

Where  $l_k|_{k = 1, 2, \dots, p}$  represents the  $k$ th output of the rule layer,  $w_{jb}^k$  which represents the weights between the membership layer and the rule layer, is assumed to be unity, and  $p$  is the number of rules.

4. Final layer is the output layer, and nodes in this layer represent the output linguistic variables. Each node in the output layer  $y_i(1, 2, \dots, n)$  is labelled as  $\sum$ , which computes the

overall output as the summation of all input signals, and be represented as:

$$y_i = \sum_{k=1}^p w_{ik} l_k \quad (12)$$

The output node, together with the links connected to it, acts as a defuzzifier. It can be referred as the normal defuzzification procedure. Moreover, it can be rewritten in the following vector form:

$$y = [y_1 \quad y_2 \quad \cdots \quad y_n]^T = WL \quad (13)$$

Where

$$W = \begin{bmatrix} w_{11} & w_{12} & \cdots & w_{1p} \\ w_{21} & w_{22} & \cdots & w_{2p} \\ \vdots & \vdots & \ddots & \vdots \\ w_{n1} & w_{n2} & \cdots & w_{np} \end{bmatrix} = [w_1 \quad w_2 \quad \cdots \quad w_n] \in \mathfrak{R}^{n \times p},$$

$$L = [l_1 \quad l_2 \quad \cdots \quad l_n]^T \in \mathfrak{R}^{p \times 1}$$

In the RANFNC scheme, the NFN is used to estimate unmodeled nonlinear function, Moreover, the RANFNC law and adaptive tuning algorithms for NFN are introduced from the stability analyses of the closed-loop system by using Lyapunov method. The input of the NFN estimator are the elements in the filtered error vector and joint positions signal, the output of the NFN estimator are the nonlinear dynamic function vectors in the local models.

Based on the powerful approximation ability [4], there exists an optimal NFN estimator to approximate the nonlinear dynamic function in (5) such that

$$f(x) = W^* L(x) + \varepsilon(x) \quad (14)$$

With  $W^*$  the ideal weight matrix and the estimative error vector  $\varepsilon(x) \in \mathfrak{R}^{n \times 1}$  are assumed to be given by

$$\tilde{W} = \arg \min_{\hat{W} \in M_x} \left[ \sup_{x \in M_x} \| f(x) - \hat{W} L(x) \| \right], \quad \| \varepsilon \| \leq \varepsilon_N \quad (15)$$

In which  $\| \cdot \|$  is the Euclidean norm,  $M_x$  and  $M_w$  are the predefined compact sets of  $x$  and  $\hat{W}$ , and the positive constant  $\varepsilon_N$  can be reduced arbitrarily by increasing the number of rules.

## 4 RANFNC Design

Define the NFN functional estimate by

$$\hat{f}(x) = \hat{W} L(x) \quad (16)$$

With  $\hat{W}$  the current values of the NFN weight as provided by the tuning algorithm. With the ideal weights required in (14) define the weight deviations or weight estimation errors as

$$\tilde{W} = W^* - \hat{W} \quad (17)$$

With  $\tau_0$  defined to be (7), select the control input

$$\tau = \tau_0 - v = \hat{W} L(x) + K_v r - v \quad (18)$$

$$v = (\varepsilon_N + d_b) \text{sgn}(r) = K_f \text{sgn}(r) \tag{19}$$

With  $v(t)$  a function to be determined to provide robustness in the face of the net reconstruction error  $\varepsilon$ . Then, the closed-loop filtered error dynamics become

$$M\dot{r} = -(K_v + C(q, \dot{q}))r + \tilde{W}L(x) + (\varepsilon + \tau_d) + v = -(K_v + C(q, \dot{q}))r + \zeta_1 \tag{20}$$

Theorem 1: Consider an n-link robot manipulator represented (1). If the RANFNC law is designed as (18), (19) and the weight update law is designed as (21), then the stability of the proposed RANFNC system can be ensured

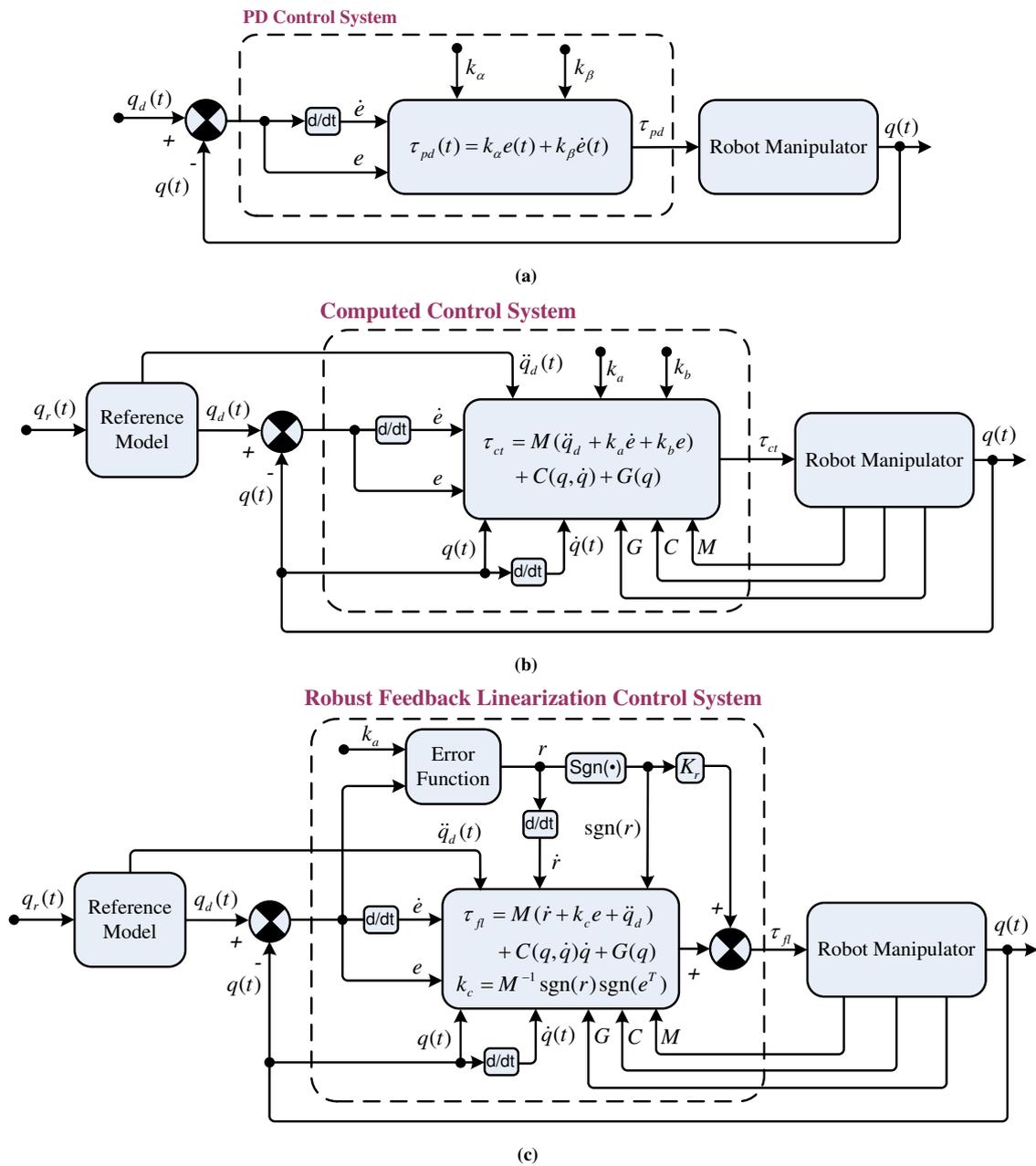


Figure 4: (a) PD control system, (b) computed torque control (CTC) system, (c) robust feedback linearization control (RFLC) system

$$\dot{\tilde{W}} = FL(x)r^T \quad (21)$$

Proof: Define a Lyapunov function candidate as

$$V(r(t), \tilde{W}) = \frac{1}{2}r^T M r + \frac{1}{2}tr(\tilde{W}^T F^{-1} \tilde{W}) \quad (22)$$

Where  $tr(\cdot)$  is a trace operator. By differentiating (22) with respect to time and using (19), (20), (21), and using properties of the robot dynamics are introduced in section 2, one can obtain.

$$\begin{aligned} \dot{V} &= \frac{1}{2}r^T M \dot{r} + \frac{1}{2}r^T \dot{M} r + tr(\tilde{W}^T F^{-1} \dot{\tilde{W}}) \\ &= -r^T K_v r + \frac{1}{2}r^T (\dot{M} - 2C)r + tr \tilde{W} (F^{-1} \dot{\tilde{W}} + Lr^T) \\ &\quad + r^T (\varepsilon + \tau_d) - r^T (\varepsilon_N + d_b) sgn(r) \\ &= -r^T K_v r + r^T (\varepsilon + \tau_d) - r^T (\varepsilon_N + d_b) sgn(r) \\ &= -r^T K_v r + r^T (\varepsilon + \tau_d) - \|r\| (\varepsilon_N + d_b) \\ &\leq -r^T K_v r \leq 0 \end{aligned} \quad (23)$$

Since  $\dot{V}(r(t), \tilde{W}) \leq 0$ ,  $\dot{V}(r(t), \tilde{W})$  is a negative semidefinite function, i.e.  $V(r(t), \tilde{W}) \leq V(r(0), \tilde{W})$ . It implies that  $r(t)$  and  $\tilde{W}$  is bounded functions. Let function  $h(t) \equiv r^T K_v r \leq -\dot{V}$  and integrate function  $h(t)$  with respect to time

$$\int_0^\tau h(t) d\tau \leq V(r(0), \tilde{W}) - V(r(t), \tilde{W}) \quad (24)$$

Because  $V(r(0), \tilde{W})$  is a bounded function, and  $V(r(t), \tilde{W})$  is a nonincreasing and bounded function, the following result can be concluded:

$$\lim_{t \rightarrow \infty} \int_0^\tau h(t) d\tau < \infty \quad (25)$$

In addition,  $\dot{h}(t)$  is bounded; thus, by Barbalats lemma can be shown that  $\lim_{t \rightarrow \infty} h(t) = 0$ . It can imply that  $r(t)$  will be converging to zero as time tends to infinite.

## 5 Numerical Simulation

A two-link robot manipulator as shown in Fig.1 is utilized in this paper to verify the effectiveness of the proposed control scheme. The detailed system parameters of this robot manipulator are given as: link mass  $m_1, m_2(kg)$ , lengths  $l_1, l_2(m)$ , angular positions  $q_1, q_2(rad)$ .

The parameters for the equation of motion (1) are adopted in [11].

$$M(q) = \begin{bmatrix} (m_1 + m_2 l_1^2) & m_2 l_1 l_2 (s_1 s_2 + c_1 c_2) \\ m_2 l_1 l_2 (s_1 s_2 + c_1 c_2) & m_2 l_2^2 \end{bmatrix}$$

$$C(q, \dot{q}) = m_2 l_1 l_2 (c_1 s_2 + s_1 c_2) \begin{bmatrix} 0 & -\dot{q}_2 \\ -\dot{q}_1 & 0 \end{bmatrix}, \quad G(q) = \begin{bmatrix} -(m_1 + m_2) l_1 g s_1 \\ -m_2 l_2 g s_2 \end{bmatrix}$$

Where  $q \in \mathfrak{R}^2$  and the shorthand notations  $c_1 = \cos(q_1)$ ,  $c_2 = \cos(q_2)$ ,  $s_1 = \sin(q_1)$  and  $s_2 = \sin(q_2)$  are used.

For the convenience of the simulation, the nominal parameters of the robotic system are given as  $m_1 = 4.6(kg)$ ,  $m_2 = 2.3(kg)$ ,  $l_1 = 0.5(m)$ ,  $l_2 = 0.2(m)$ ,  $g = 9.8(m/s^2)$  and the initial conditions  $q_1(0) = 0.5$ ,  $q_2(0) = 0.5$ ,  $\dot{q}_1(0) = 0$ ,  $\dot{q}_2(0) = 0$ . The desired reference trajectories are  $q_{d1}(t) = \sin(2t)$ ,  $q_{d2}(t) = \cos(2t)$  respectively.

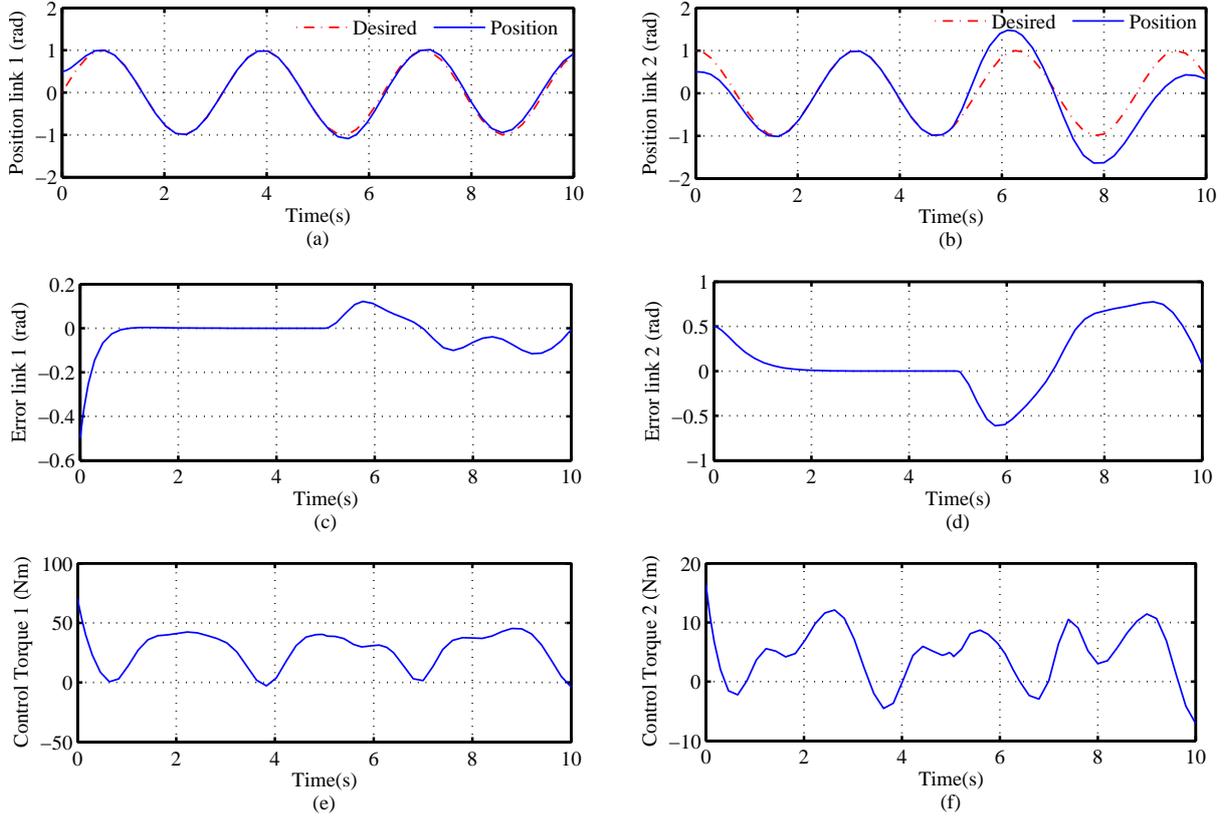


Figure 5: Simulated position responses, tracking errors, and control torques of the CTC control system at joints 1 and 2

The most important parameters that effect the control performance of the robotic system are the external disturbance and the friction term which are denoted  $t_l(t)$  and  $f(\dot{q})$ , in simulation, parameter variation situation and disturbance situation occurring at 5s are considered. The parameter variation situation is that 1(kg) weight is added to the mass of link 2, i.e.  $m_2 = 3.3(kg)$ . The disturbance situation is that external forces are injected into the robotic system, and their shapes are expressed as follows:  $t_l(t) = [5 \sin(5t) \quad 5 \cos(5t)]^T$ . In addition, friction forces are also considered in this simulation and given as:  $f(\dot{q}) = [2\dot{q}_1 + 0.8\text{sgn}(\dot{q}_1) \quad 4\dot{q}_2 + 0.1\text{sgn}(\dot{q}_2)]^T$ .

To this end, the simulation situations are adopted to demonstrate the robust property of the proposed control scheme. In order to exhibit the superior control performance of the proposed RANFNC scheme, three extra control systems including an RFLC system shows in Fig. 4(c), a conventional computed torque control (CTC) and a proportional differential (PD) control are examined in the mean time [13]. Moreover the conventional CTC system as shown in Fig.4(b) can be expressed as

$$\tau_{cd} = M(\ddot{q}_d + k_a \dot{e}(t) + k_b e(t)) + C(q, \dot{q}) + G(q) \quad (26)$$

The PD control system as shown in Fig. 4(a) can be expressed as

$$\tau_{pd} = k_{\alpha}e(t) + k_{\beta}\dot{e}(t) \quad (27)$$

The gain in these control system are given as

$$k_{\alpha} = \begin{bmatrix} 2500 & 0 \\ 0 & 1000 \end{bmatrix}, \quad k_{\beta} = \begin{bmatrix} 20 & 0 \\ 0 & 25 \end{bmatrix}, \quad k_a = \begin{bmatrix} 6 & 0 \\ 0 & 6 \end{bmatrix}, \quad k_b = \begin{bmatrix} 9 & 0 \\ 0 & 9 \end{bmatrix}, \quad K_v = \begin{bmatrix} 5 & 0 \\ 0 & 5 \end{bmatrix},$$

$$\eta = 20, \quad K_r = d_b = 5. \quad (28)$$

The gain matrices of  $k_a$ , and  $k_b$ , are determined so that the roots of the characteristic polynomial of  $k_a\dot{e} + k_b e$  lie strictly in the open left half of the complex plane, i.e.  $\lim_{t \rightarrow \infty} e(t) = 0$ . It means that the CTC system shown in (26) is globally asymptotically stable as the root dynamics (1) without the consideration of system uncertainties. However, the stability of the closed-loop control system may be destroyed if the system dynamics are perturbed by external disturbance. However gains  $k_{\alpha}$  and  $k_{\beta}$  in the PD control are selected according to the Ziegler-Nichols tuning rule. In addition, the selection of learning rates  $\eta$  is dependent on the significance of tuning objects.

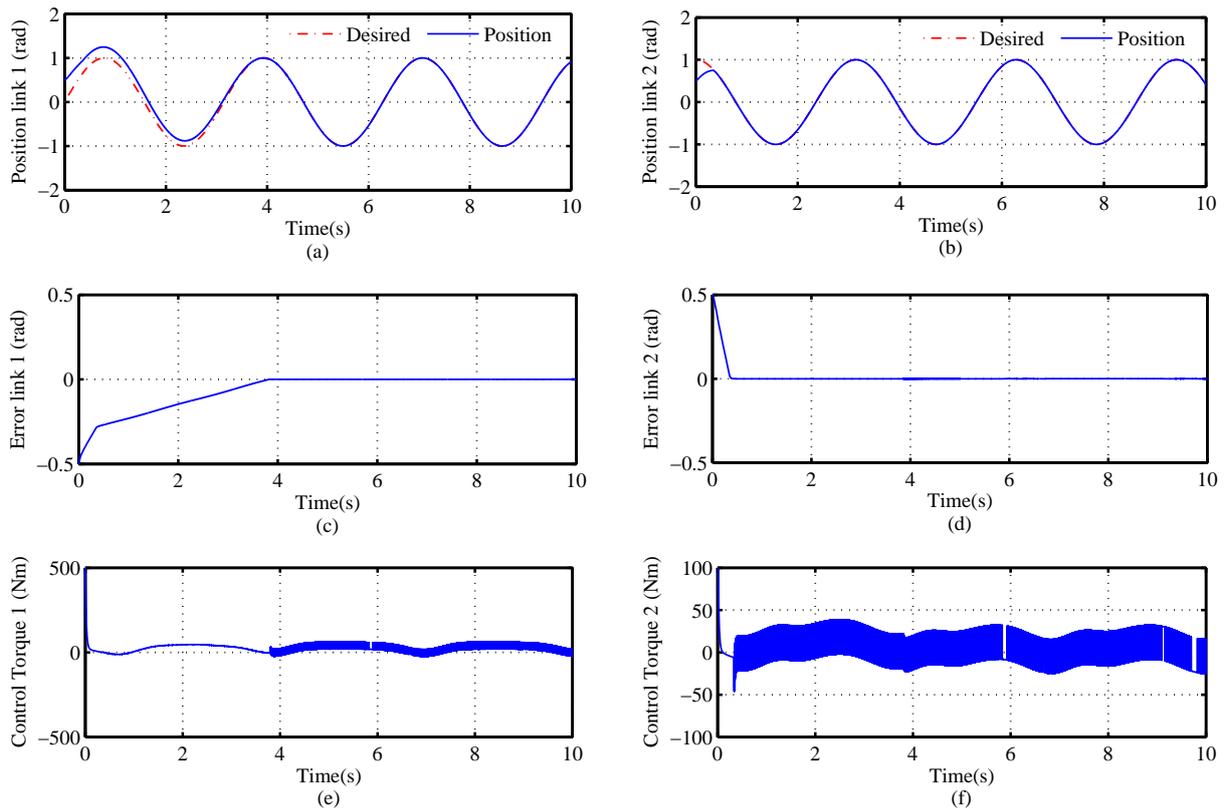


Figure 6: Simulated position responses, tracking errors, and control torques of the RFLC control system at joints 1 and 2

The simulated results of CTC system, the responses of joint position, tracking error and control torque are depicted Fig. 5(a-d), and (e-f), respectively. From the simulated results, in interval from beginning to 5s, favourable tracking responses can only be obtained for the nominal situation. However, since the control gains in (28) are determined without considering the joint

friction and external disturbance. So, poor tracking responses after 5s are resulted due to the occurrence of joint friction and external disturbance. In the RFLC system are depicted in Fig. 6. The joint position responses, tracking error and control torque are depicted in Fig. 6(a-d) and (e-f), respectively. The robust control performance of the RFLC system is obvious under the occurrence of system uncertainties. However, the undesirable chattering phenomenon in the control torque.

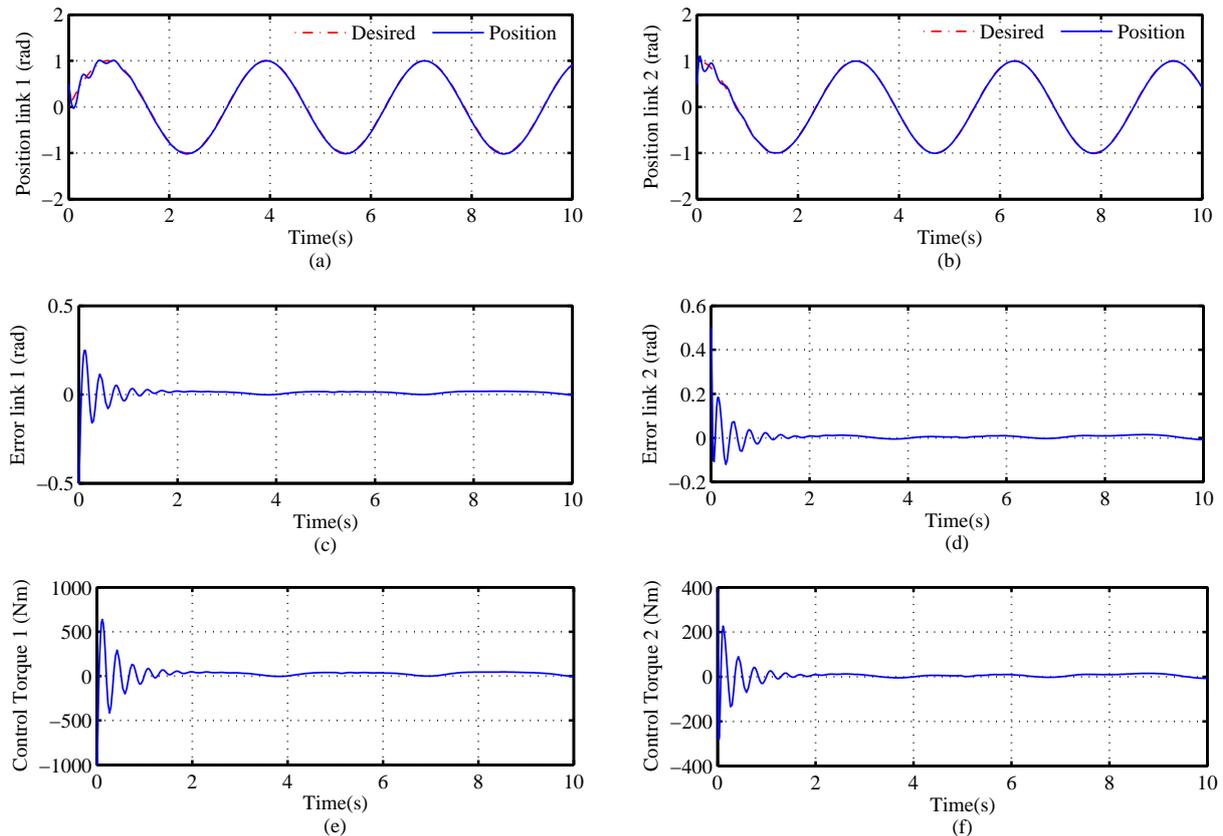


Figure 7: Simulated position responses, tracking errors, and control torques of the PD control system at joints 1 and 2

The PD control system based on model-free design is offered to apply comparable responses to manifest the performance of the RANFNC system. The simulated responses of joint position, tracking error, and control torque, are represented in Fig. 7(a-d) and (e-f). From simulated results see that, the tracking responses are greatly improved and the chattering phenomenon are much reduced.

Now, the proposed RANFNC system depicted in Fig. 3 is applied to control the robot manipulator for comparison. The simulated results of joint position responses, tracking error and control torque are depicted in Fig. 8(a-d) and (e-f), respectively. Because of, all parameters in the NFN are roughly initialized, the tracking errors are gradually decreased though online training process whether the uncertainties exist or not. Furthermore, robust control performance of the RANFNC system, both in the condition of joint friction, parameter variation, and external disturbance are obvious. Compared these results with the CTC, RFLC and PD control systems, the control torque of proposed RANFNC system is not chattering phenomenon.

## 6 Conclusions

This paper has successfully implemented an RANFNC system to control the joint position of a two-link robot manipulator for achieving desired position control. All the system dynamic could be unknown and no strict constraints. The NFN is used to compensate the uncertainty of the system. All adaptive learning laws in the RANFNC system were derived in the sense of a projection algorithm and Lyapunov theorem so that the network convergence and system-tracking stability of the closed-loop control system can be ensured whether or not the uncertainties occur. Simulated results of a two link robot manipulator via various existing control frameworks including CTC RFLC and PD control were also applied in this paper to compare and display the manipulative performance of the proposed control system. According to the result as depict in Figs. 5-8, the desired position tracking response of the RANFNC system can be controlled closely follow specific reference trajectories under wide range of disturbance. The main of the paper is to construct a simpler and more efficient intelligent control system without dynamic knowledge of plant. While ensuring the convergence and tracking stability of the closed-loop system. The proposed RANFNC system can also be applied to other systems, such as mobile robotic, AC servo system and so on.

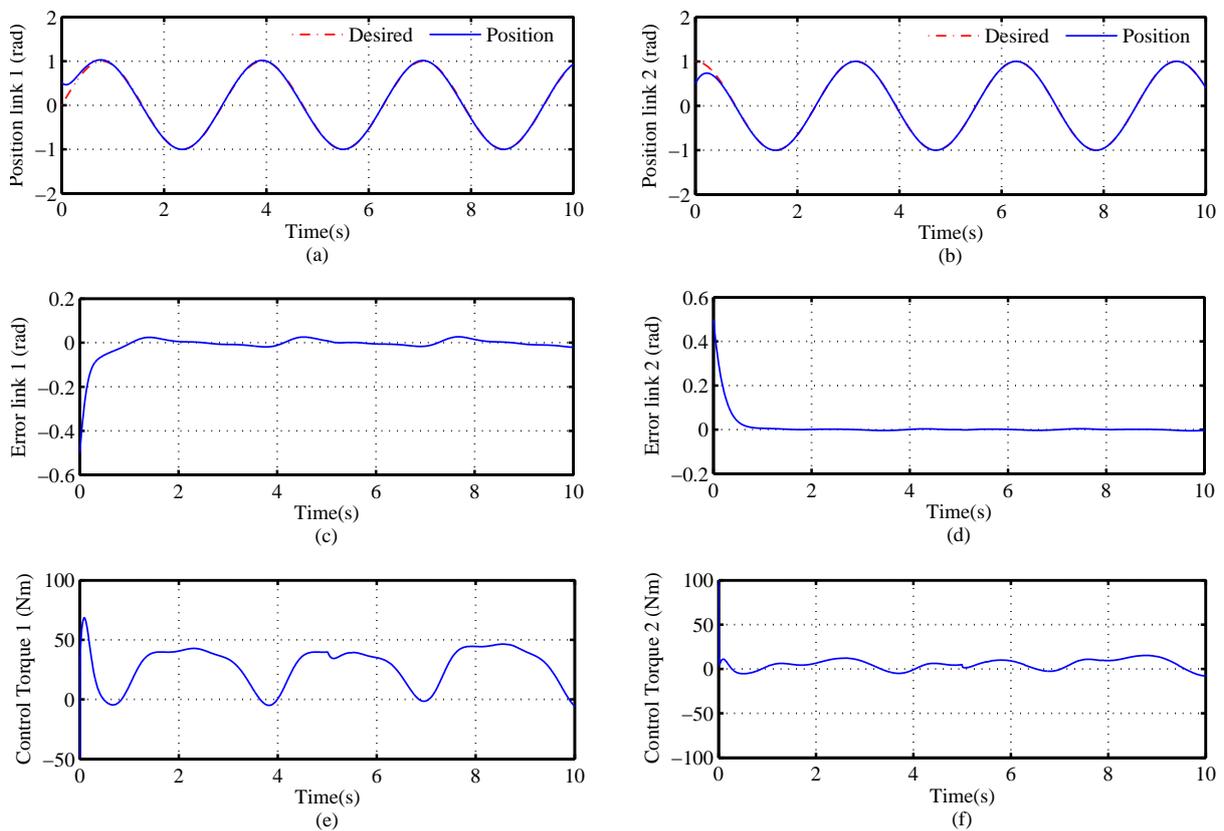


Figure 8: Simulated position responses, tracking errors, and control torques of the RANFNC control system at joints 1 and 2

## 7 Acknowledgment

This work was supported by the National Natural Science Foundation of China (60775047; 60835004), the National High Technology Research and Development Program of China (863

Program) (2007AA04Z244; 2008AA04Z214). The authors would like to thank the associate editor and the reviewers for their valuable comments.

## Bibliography

- [1] Jinzhu Peng, Yaonan Wang, Wei Sun, Yan Liu, A neural network sliding mode controller with application to robotic manipulator, *IEEE Conf. Int. Control*, 1:2011-2015, 2000
- [2] B.K. Yoo and W.C Ham, Adaptive control of robot manipulator using fuzzy compensator, *IEEE Trans. Ind. Electron*, 8(2):186-199, 2000
- [3] Shuzhi S. Ge, Adaptive neural network control of robot manipulator in task space, *IEEE Trans. Ind. Electron*, 44(6):746-752, 1997
- [4] C.T. Lin and C.S. George Lee, Neural Fuzzy Systems, *Englewood Cliffs*, Prentice-Hall, 1996
- [5] Y.Q. Zhang and A. Kandel, Compensatory neural-fuzzy systems with fast learning algorithms, *IEEE Trans. Neural Newt*, 9(1):83-105, 1998
- [6] Vesselenyi T., Dzitac S., Dzitac I., Manolescu M.-J., Fuzzy and Neural Controllers for a Pneumatic Actuator, *NT J COMPUT COMMUN*, ISSN 1841-9836. 2(4): 375-387, 2007
- [7] Alavandar S., Nigam M.J., Neuro-Fuzzy based Approach for Inverse Kinematics Solution of Industrial Robot Manipulators, *INT J COMPUT COMMUN*, ISSN 1841-9836, 3(3):224-234, 2008
- [8] AlavandarS., Nigam M.J., Inverse Kinematics Solution of 3DOF Planar Robot using ANFIS, *NT J COMPUT COMMUN*, ISSN 1841-9836, 3(S):150-155, 2008
- [9] L.X. Wang, A course in Fuzzy Systems and Control, *Englewood Cliffs*, NJ:Prentice Hall, 1997
- [10] O. Omidvar and D.L. Elliott, Neural Systems for Control, *Englewood Cliffs*, NJ: Prentice-Hall, 1997
- [11] B.S. Chen, H.J. Uang, and C.S. Tseng, Robust tracking enhancement of robot systems including motor dynamics: A fuzzy-based dynamic game approach, *IEEE Trans. Fuzzy syst*, 11(4):538-553, 1998
- [12] R.J. Schilling, Fundamentals of Robotics, *Analysis and control. Hoboken*, NJ: Prentice-Hall, 1998
- [13] J.J.E. Slotime and W. Li, Applied Nonlinear Control, *Hoboken*, NJ: Prentice-Hall, 1991
- [14] H. K. Khalil, Nonlinear Systems. Englewood Cliffs, *Englewood Cliffs*, NJ: Prentice-Hall, 1996
- [15] K. Liu and F.L. Lewis, Robust control techniques for general dynamic system, *J. intell. Robotic syst*, 6:33-49, 1992
- [16] F.L. Lewis, C.T. Abdallah, and D.M Dawson, Control of Robot Manipulators, *New York: Macmillan*, 1993