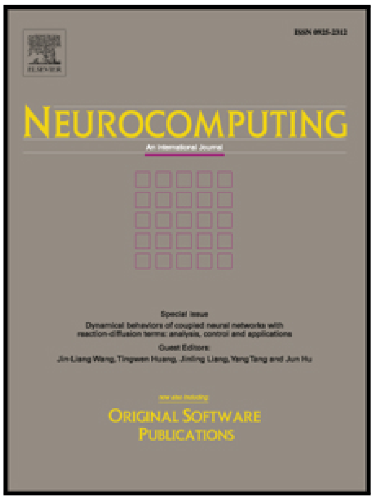
Paper:
Yang, C. Neuro-Adaptive Observer based Control of Flexible Joint Robot. *Neurocomputing*
http://dx.doi.org/10.1016/j.neucom.2017.05.011

Accepted Manuscript



Neuro-Adaptive Observer based Control of Flexible Joint Robot

Xin Liu, Chenguang Yang, Zhiguang Chen, Min Wang, Chun-Yi Su

PII: S0925-2312(17)30818-4
DOI: 10.1016/j.neucom.2017.05.011
Reference: NEUCOM 18415

To appear in: *Neurocomputing*

Received date: 15 March 2017
Revised date: 12 April 2017
Accepted date: 8 May 2017

Please cite this article as: Xin Liu, Chenguang Yang, Zhiguang Chen, Min Wang, Chun-Yi Su, Neuro-Adaptive Observer based Control of Flexible Joint Robot, *Neurocomputing* (2017), doi: 10.1016/j.neucom.2017.05.011

Neuro-Adaptive Observer based Control of Flexible Joint Robot

Xin Liu[a], Chenguang Yang[a,b,*], Zhiguang Chen[a], Min Wang[a], Chun-Yi Su[a,c]

[a] *Key Lab of Autonomous Systems and Networked Control, School of Automation Science and Engineering, South China University of Technology, Guangzhou 510641, China.*
[b] *Zienkiewicz Centre for Computational Engineering, Swansea University, UK*
[c] *Concordia University, Canada*

Abstract

Due to high nonlinearity, strong coupling and time-varying characteristics of flexible joint robot manipulators, their control design is generally a challenging problem. There are inevitable uncertainties associated with their kinematics and dynamics, so that accurate models would not be available for control design. Furthermore, practically we may face the problem that state variables required by the controller are not measurable. In this paper, we focus on the study of control system design using a neural network observer to solve the aforementioned unmeasurable problem. First, we propose an observer based on Radial basis function (RBF) neural network to estimate state variables of the normal system. We then design the controller based on dynamic surface control method for a single link flexible joint manipulator whose model is unknown. The unknown model of the manipulator is constructed by RBF neural network. The stability of the observer and controller is shown by Lyapunov method. Finally, simulation studies are performed to test and verify the effectiveness of the proposed controller.

Keywords: Flexible joint manipulator system; Dynamic surface control; Neural network; State observer

*Corresponding author
Email address: cyang@ieee.org (Chenguang Yang)

1. INTRODUCTION

Programmed robots performing repeated tasks play an important role in automated manufacturing, especially on the assembly line, to reduce labor cost through mass production. Towards Industry 4.0, mass customization is coming because each individual customers need for a unique product is greatly promoted [1]. Highly increased product variety and production process variability bring considerable uncertainty to assembly line design. Reconfiguration of the assembly line involves reprogramming of robots, which involves more than 40% of capitalized cost [2]. Modern manufacturing thus calls for a flexible approach to managing production process variability. The most flexible factor in a manufacturing process is perhaps human operators, whom have natural abilities of sensitivity and improvisation to unpredictable events, fast processing of varying information as well as quick adaption when switching tasks. In fact, manual assembly usually reduces initial investment, but cost significantly increases with less automation thereafter. In this instance, the best solution seems to be to exploit human-robot physical cooperation to close the gap between fully automated manufacturing lines and fully manual assembly. It has been experimentally demonstrated that human and robot working collaboratively will be more efficient and flexible than human or robot working alone [3]. Enabled by the revised ENISO 10218 standard Parts 1&2 and the ISO/TS 15066 specification, collaborative robots are now allowed to work hand in hand with our humans. It is expected that these robots will eventually cost less and have a greater range of capabilities than those used in manufacturing today. Meanwhile, collaborative robots are required to safely perform physical interactions in the dynamically changing and unstructured working environments [4]. A common approach to improve the physical interactiveness of the manipulators mechanical structure is to make use of a flexible joint actuation. Flexible joints provide the manipulator with a valuable compliant behaviour so if the flexible joint manipulators encounter obstacles during operation, the contact force between manipulators and obstacles may be relatively slight and manipulators may stop immediately.

Thus, flexible joint manipulators have been widely used in the fields of human-robot physical cooperation. In [5], They propose a variable stiffness joint for a robot manipulator, the leaf springs is used to generate compliance and the position and stiffness of the joint are controlled by four-bar linkages. In [6] a control method to regulate the driving torque of flexible joint manipulators is proposed, A servo-controller which can asymptotically regulate the driving torque with unknown parameters is derived.

In recent years, there are many control methods of manipulator have been proposed such as adaptive neural network control [7, 8, 9], robust control [10], vibration control [11, 12], fuzzy control [13] and cooperative control [14]. For the complex nonlinear systems whose model is uncertain backstepping is an advanced control method. But it suffers from the curse of dimension in the process of controller design. In order to avoid differentiating virtual control signals, dynamic surface control method was proposed in [15, 16] by using first-order filters within the backstepping controller design. Thereby the control law is easier to compute and achieve. This method is suitable for high-dimensional nonlinear systems such as flexible joint manipulators. Due to the uncertainties existing in the dynamic models of flexible joint manipulator systems, it is difficult to obtain the models of flexible joint manipulator systems accurately [17]. Neural network is an applicable method to approximate unknown models. In[18], the neural networks is employed to compensate for uncertainties in dynamics of both the robot arms and the manipulated object . In [19], they use RBF neural networks to compensate for the effect caused by the uncertainties from internal and external. In [20, 21, 22, 23], neural networks are used to approximate the unknown models.

Due to constraints of sensor deployment, we may not be able to measure all the state variables, then the controller based on state feedback is unavailable. Thus, observers which can estimate the state variables unmeasurable are necessary. Conventional nonlinear observers are generally applicable to systems whose models are precisely known [24, 25], while we consider systems with unknown models. Neural network has become a powerful tool for state observation with uncertain models. In [26] a Kalman filter is combined with a neural network for a general multiple-input-multiple-output (MIMO) nonlinear system and the multilayer feedforward neural network is used to deduce the gain of the observer. In [27], they estimate the states of affine single-input-single-output (SISO) nonlinear systems by two separate linear-in-parameter neural networks (LPNN). In [28], an RBF neural network was adopted to approximate the nonlinearities. In [29], they propose a recurrent neural network for a general MIMO nonlinear systems. The weights of neural network were updated based on the backpropagation (BP) algorithm. In this paper, we adopt RBF network to build the observer, then combined the observer with the controller to control the flexible joint manipulator system in which the angular position and angular velocity of motor shaft is unknown.

Radial Basis Function (RBF) neural network which can approach to any nonlinear function can replace the unknown nonlinear system, help us to design the controller[30]. The structure of the RBF network is shown in Fig. 1 [31]. The input signal is transmitted to the hidden layer through the input layer. The nodes in the hidden layer are the basis functions, and the output layer is a linear function. There are many kinds of basis functions available. Here we use the Gaussian function as the basis function.

$$S_j(x) = exp[-\frac{\|x - c_j\|^2}{b_j^2}] \tag{1}$$

where $j = 1, 2, 3, \cdots, q$, $x = [x_1, x_2, \cdots, x_n]^T$ is the input sample. c_j is the center of the hidden layer nodes, b_j^2 is the width of the Gaussian function, S_j is the output of the hidden layer, q is the number of nodes in the hidden layer. The output of the RBF network is the linear superposition of the hidden layer nodes

$$y = \sum_{j=1}^{q} W_j S_j(x) \tag{2}$$

W_j is the weight vector of the neural network. By selecting the appropriate weight vector, RBF network can approximate a continuous function with arbi-

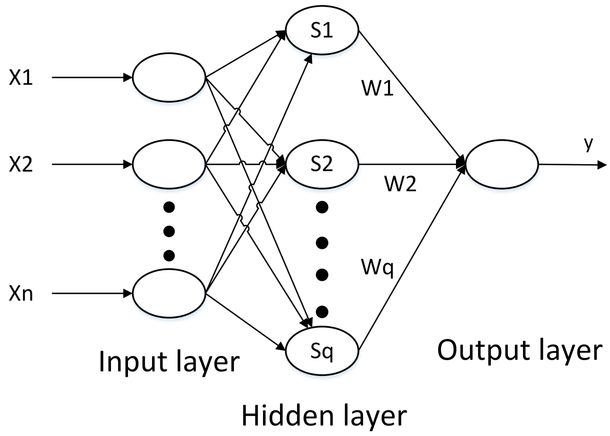


Figure 1: RBF neural network

trary precision.

$$h(Z) = W^T S(Z) + \epsilon \tag{3}$$

where $\forall Z \in \Omega_Z$, W is the optimal weight vector, ϵ is approximation error.

In this paper, an observer based on RBF neural network is proposed for flexible joint manipulators with unknown systems in Section 2. In Section 3 we design the controller based on dynamic surface control method for a single link flexible joint manipulator whose model is unknown. The unknown model of the manipulator is constructed by RBF neural network. Finally, in Section 4 simulation studies are performed to test and verify the effectiveness of the proposed controller.

2. NEURO-ADAPTIVE OBSERVER

2.1. The proposed neuro-adaptive observer

In this section we propose a state observer based on RBF neural network. The general model of a nonlinear system is

$$\begin{aligned} \dot{x}(t) &= f(x,u) \\ y(t) &= Cx(t) \end{aligned} \tag{4}$$

$u \in R^{M_u}$ is the input, $y \in R^{M_y}$ is the output, $x \in R^{M_x}$ is the state vector of the system, and f is an unknown function. C is the output matrix of the system. Let us define $g(x,u) = f(x,u) - Ax$ we can obtain:

$$\begin{aligned} \dot{x}(t) &= Ax + g(x,u) \\ y(t) &= Cx(t) \end{aligned} \tag{5}$$

A is a Hurwitz matrix, (C,A) is observable.

First, we build the observer model as:

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x} + \hat{g}(\hat{x},u) + G(y - C\hat{x}) \\ \hat{y}(t) &= C\hat{x}(t) \end{aligned} \tag{6}$$

where $\hat{x}$ is the state of the observer. We select the observer gain $G \in R^{n \times m_y}$ such that $A - GC$ is a Hurwitz matrix. We use RBF neural network approximates the nonlinear system. Thus, $g(x,u)$ can be represented as:

$$g(x,u) = W^T S(\overline{x}) + \epsilon(x) \tag{7}$$

W is the weight matrix of the output layer, $\overline{x} = [x^T, u^T]^T$, $\epsilon(x)$ is the approximation error of neural network. $S(\cdot)$ is the transfer function of the hidden neurons which is a Gaussian function:

$$S_j(\overline{x}) = exp(-\frac{\|\overline{x} - c_j\|^2}{b_j^2}). \tag{8}$$

Assumption 1: There is the upper bound on optimal weight matrices W such that

$$\|W\| \leq W_m. \tag{9}$$

Property 1: The Gaussian function is bounded by

$$\|S(\overline{x})\| \leq S_m. \tag{10}$$

Let us assume $g(x, u)$ can be approximated by

$$\hat{g}(\hat{x}, u) = \hat{W}^T S(\hat{\overline{x}}). \tag{11}$$

The proposed observer is given by

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x} + \hat{W}^T S(\hat{\overline{x}}) + G(y - C\hat{x}) \\ \hat{y}(t) &= C\hat{x}(t). \end{aligned} \tag{12}$$

In order to prove the stability of the observer we define the weight error $\tilde{W} = \hat{W} - W$ and state variable error $\tilde{x} = \hat{x} - x$. According to (5) and (12) we obtain

$$\begin{aligned} \dot{\tilde{x}}(t) =& A\hat{x} + \hat{W}^T S(\hat{\overline{x}}) - Ax - W^T S(\overline{x}) - G(C\hat{x} - Cx) + \epsilon(x) \\ \tilde{y}(t) =& C\tilde{x}(t). \end{aligned} \tag{13}$$

By adding $W^T S(\hat{\overline{x}})$ and subtracting $W^T S(\hat{\overline{x}})$ on left hand side of (13), we obtain

$$\begin{aligned} \dot{\tilde{x}}(t) &= A_c\tilde{x} + \tilde{W}^T S(\hat{\overline{x}}) + w(t) \\ \tilde{y}(t) &= C\tilde{x}(t) \end{aligned} \tag{14}$$

$A_c = A - GC$, $w(t) = W^T[S(\hat{\overline{x}}) - S(\overline{x})] + \epsilon(x)$, $w_m > 0$ satisfy $\|w(t)\| \leq w_m$ due to the boundedness of the optimal neural network weight.

2.2. STABILITY ANALYSIS

Definition 1: The adaptive update rate of weights is

$$\dot{\hat{W}} = \Gamma[S(\hat{\hat{x}})\tilde{y}^T C - \rho\|C\tilde{x}\|\hat{W}] \tag{15}$$

where $\Gamma = \Gamma^T$ is a positive definite matrix.

Theorem 1: Considering the general nonlinear system described by (4), the observer described by (12) and the adaptive update rates of weights described by (15). For any bounded initial conditions, there exists suitable parameters G, Γ and ρ such that the proposed observer scheme guarantees:

1) all the signals in the observer are uniformly ultimately bounded;

2) the estimated errors x_i converges to a arbitrarily small neighborhood of zero.

Proof: Since the optimal weight W is a constant matrix thus we obtain $\dot{W} = 0$. In terms of the weight errors $\tilde{W} = \hat{W} - W$ we obtain

$$\begin{aligned} \dot{\tilde{W}} = \dot{\hat{W}} &= \Gamma[S(\hat{\hat{x}})\tilde{y}^T C - \rho\|C\tilde{x}\|(\tilde{W} + W)] \\ &= \Gamma[S(\hat{\hat{x}})\tilde{x}^T C^T C - \rho\|C\tilde{x}\|(\tilde{W} + W)]. \end{aligned} \tag{16}$$

Definition 2: The Lyapunov function is

$$L = \frac{1}{2}\tilde{x}^T P\tilde{x} + \frac{1}{2}tr(\tilde{W}^T\Gamma^{-1}\tilde{W}) \tag{17}$$

where P is a positive definite matrix. Then, we define Q

$$Q = -(A_c^T P + PA_c). \tag{18}$$

A_c is Hurwitz matrix, Q is a positive definite matrix. Then, we obtain the time derivative of L

$$\dot{L} = \frac{1}{2}\tilde{x}^T P\dot{\tilde{x}} + \frac{1}{2}\dot{\tilde{x}}^T P\tilde{x} + tr(\tilde{W}^T\Gamma^{-1}\dot{\tilde{W}}). \tag{19}$$

Now, by substituting (14), (16) and (18) into (19), we obtain

$$\dot{L} = -\frac{1}{2}\tilde{x}^T Q\tilde{x} + \tilde{x}^T P\tilde{W}^T S(\hat{\bar{x}}) + \tilde{x}^T Pw + tr[\tilde{W}^T S(\hat{\bar{x}})\tilde{x}^T C^T C - \tilde{W}^T\rho\|C\tilde{x}\|(\tilde{W} + W)]. \tag{20}$$

Using the property

$$tr[\tilde{W}^T(-W - \tilde{W})] \le W_m\|\tilde{W}\| - \|\tilde{W}\|^2$$

$$tr(\tilde{W}^T S(\hat{\bar{x}})\tilde{x}^T C^T C) \le S_m\|C\|^2\|\tilde{x}\|\|\tilde{W}\| \tag{21}$$

we obtain

$$\begin{aligned}\dot{L} \le& -\frac{1}{2}\lambda_{min}(Q)\|\tilde{x}\|^2 + S_m\|P\|\|\tilde{x}\|\|\tilde{W}\| + w_m\|P\|\|\tilde{x}\| + S_m\|C\|^2\|\tilde{x}\|\|\tilde{W}\| \\ &+ \rho W_m\|C\|\|\tilde{x}\|\|\tilde{W}\| - \rho\|C\|\|\tilde{x}\|\|\tilde{W}\|^2 \\ =& M_1\end{aligned} \tag{22}$$

where $\lambda_{min}(Q) > 0$ is the smallest eigenvalue of Q. Then, we obtain

$$\begin{aligned}M_1 =& -\frac{1}{2}\lambda_{min}(Q)\|\tilde{x}\|^2 + \|\tilde{x}\|(S_m\|P\|\|\tilde{W}\| + S_m\|C\|^2\|\tilde{W}\| + \rho W_m\|C\|\|\tilde{W}\| \\ &- \rho\|C\|\|\tilde{W}\|^2) + w_m\|P\|\|\tilde{x}\| \\ =& -\frac{1}{2}\lambda_{min}(Q)\|\tilde{x}\|^2 + \Bigg[-\rho\|C\|\left(\|\tilde{W}\| - \frac{S_m\|P\| + S_m\|C\|^2 + \rho\|C\|W_m}{2\rho\|C\|}\right)^2 \\ &+ \frac{(S_m\|P\| + S_m\|C\|^2 + \rho\|C\|W_m)^2}{4\rho\|C\|}\Bigg]\|\tilde{x}\| + w_m\|P\|\|\tilde{x}\|.\end{aligned} \tag{23}$$

Since

$$-\rho\|C\|\left(\|\tilde{W}\| - \frac{S_m\|P\| + S_m\|C\|^2 + \rho\|C\|W_m}{2\rho\|C\|}\right)^2 < 0 \tag{24}$$

we can obtain

$$M_1 < -\frac{1}{2}\lambda_{min}(Q)\|\tilde{x}\|^2 + w_m\|P\|\|\tilde{x}\| + \frac{(S_m\|P\| + S_m\|C\|^2 + \rho\|C\|W_m)^2}{4\rho\|C\|}\|\tilde{x}\| = M_2. \tag{25}$$

Let us define

$$K = \frac{(S_m\|P\| + S_m\|C\|^2 + \rho\|C\|W_m)^2}{4\rho\|C\|} > 0 \tag{26}$$

we can obtain

$$M_2 = -\frac{1}{2}\lambda_{min}(Q)\|\tilde{x}\|^2 + w_m\|P\|\|\tilde{x}\| + K\|\tilde{x}\|. \tag{27}$$

And if

$$\|\tilde{x}\| > \frac{2Pw_m + 2K}{\lambda_{min}(Q)} = v \tag{28}$$

then

$$\dot{L} < M_1 < M_2 < 0. \tag{29}$$

When $\tilde{x} > v$, $\dot{L} < 0$ is negative, then $\tilde{x}$ is bounded. If we choose the observer gain G such that the eigenvalue of A_c is big enough then the value of $\lambda_{min}(Q)$ is small relative to $2Pw_m + 2K$. Thus, we can obtain a v small enough to ensure the accuracy and stability of the system to meet the requirements.

Then, we notice that $\tilde{x}$, W, C and $S(\hat{\tilde{x}})$ are all bounded, and $\rho > 0$. Thus according to (16) we obtain a system whose inputs $\Gamma[S(\hat{\tilde{x}})\tilde{x}^T C^T C - \rho\|C\tilde{x}\|W]$ are bounded and state matrix of the system $-\rho\|C\tilde{x}\|$ is a Hurwitz matrix. Thus, this system is stable, $\tilde{W}$ is also bounded.

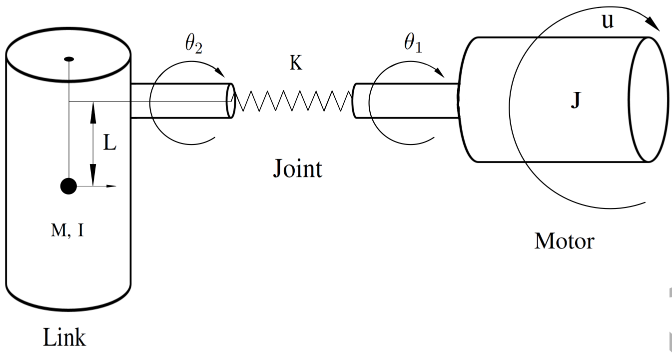


Figure 2: The model of single link flexible joint manipulator

3. CONTROLLER OF THE FLEXINLE JOINT MANIPULATOR BASED ON NEURO-ADAPTIVE OBSERVER

3.1. Problem formulation

First, we establish precise the mathematical model of flexible joint manipulator [32, 33]. We consider a single-joint flexible joint manipulator which can rotate in vertical plane. The joint can deform on the direction of rotation only and the connecting rod is rigid, we ignore viscous damping [34]. The model of single link flexible joint manipulator is shown in Fig. 2. According to Fig. 2 we see the system consists of two parts. The left part is the motor while the right part is the manipulator. In the middle there is a spring. Driving torque provided by motor is u. Rotary inertia of motor is J. The angular position of motor shaft is θ_1. The stiffness of connecting rod is K. The angular position of manipulator shaft is θ_2. The distance from the centroid of the link to the axis of the joint is L. The quality and inertia of the manipulator are M and I respectively. The angular velocity and angular acceleration of motor shaft are denoted by $\dot{\theta}_1$ and $\ddot{\theta}_1$ respectively. Similarly, the angular velocity and angular acceleration of manipulator shaft are denoted by $\dot{\theta}_2$ and $\ddot{\theta}_2$ respectively. Hence, the system can be described by following differential equation

$$\begin{cases} I\ddot{\theta}_2 + MgLsin\theta_2 + K(\theta_2 - \theta_1) = 0 \\ J\ddot{\theta}_1 - K(\theta_2 - \theta_1) = u \end{cases} \tag{30}$$

Let us define: $x_1 = \theta_2$. $x_2 = \dot{\theta}_2$. $x_3 = \theta_1$. $x_4 = \dot{\theta}_1$. We can obtain the equation of state as follows.

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = -\frac{MgL}{I} sinx_1 - \frac{K}{I}(x_1 - x_3) \\ \dot{x}_3 = x_4 \\ \dot{x}_4 = \frac{K}{J}(x_1 - x_3) + \frac{1}{J}u \end{cases} \tag{31}$$

Then, we use Neuro-Adaptive Observer proposed in the Section 2 to estimate the state variables of single link flexible joint manipulator with unknown model. We assume we can't measure the angular position and angular velocity of motor shaft directly. We obtain the state space of single link flexible joint manipulator

$$\dot{X} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{K}{I} & 0 & \frac{K}{I} & 0 \\ 0 & 0 & 0 & 1 \\ \frac{K}{J} & 0 & -\frac{K}{J} & 0 \end{bmatrix} X - \begin{bmatrix} 0 \\ \frac{MgL}{I} sinx_1 \\ 0 \\ 0 \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J} \end{bmatrix} u \tag{32}$$

$$Y = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix} X \tag{33}$$

where

$$X = \begin{bmatrix} x_1 \\ x_2 \\ x_3 \\ x_4 \end{bmatrix} \tag{34}$$

is the state variable of system. Y is the output of system. u is the input of system. Let us define

$$f_2(\overline{x}_2) = -\frac{MgL}{I} sinx_1 - \frac{K}{I} x_1$$

$$f_4(\overline{x}_4) = \frac{K}{J}(x_1 - x_3)$$

where $\overline{x}_2 = [x_1, x_2]^T$ and $\overline{x}_4 = [x_1, x_2, x_3, x_4]^T$ then we can convert (31) into

$$\dot{X} = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{K}{I} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 \\ f_2(\overline{x}_2) \\ 0 \\ f_4(\overline{x}_4) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J} \end{bmatrix} u \tag{35}$$

We assume $f_2(\overline{x}_2)$ and $f_4(\overline{x}_4)$ are unknown.

Then, according to (32) and (33) we can obtain

$$\begin{aligned} \dot{x}(t) &= f(x, u) \\ y(t) &= Cx(t) \end{aligned} \tag{36}$$

$$f(x, u) = \begin{bmatrix} 0 & 1 & 0 & 0 \\ 0 & 0 & \frac{K}{I} & 0 \\ 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 \end{bmatrix} X + \begin{bmatrix} 0 \\ f_2(\overline{x}_2) \\ 0 \\ f_4(\overline{x}_4) \end{bmatrix} + \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{1}{J} \end{bmatrix} u$$

$$C = \begin{bmatrix} 1 & 0 & 0 & 0 \end{bmatrix}.$$

Then by extracting Ax from (36), we obtain

$$\begin{aligned} \dot{x}(t) &= Ax + g(x, u) \\ y(t) &= Cx(t) \end{aligned} \tag{37}$$

where A is a Hurwitz matrix adjustable, $g(x, u) = f(x, u) - Ax$. Since $g(x, u)$ contains unknown parts we use the neural network shown in Section 2 to estimate it, we build the observer as follows

$$\begin{aligned} \dot{\hat{x}}(t) &= A\hat{x} + \hat{W}^T S(\hat{\overline{x}}) + G(y - C\hat{x}) \\ \hat{y}(t) &= C\hat{x}(t). \end{aligned} \tag{38}$$

3.2. Controller design

Assumption 2: In this section we assume we can not measure the angular position x_3 and angular velocity x_4 of motor shaft.

We use the observer proposed in the previous section estimate x_3 and x_4. Thus, we define $\hat{x}_3 - x_3 = \tilde{x}_3$, $\hat{x}_4 - x_4 = \tilde{x}_4$. According to the previous section

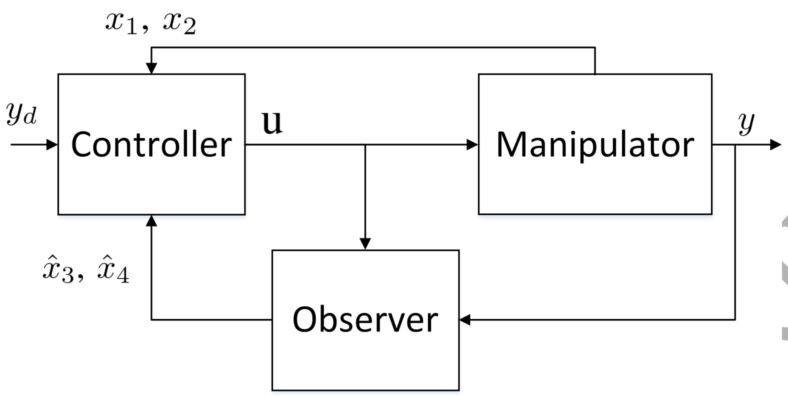


Figure 3: Controller structure based on neural network observer

135 we can obtain if $\|\tilde{x}\| > v$ then $\dot{L} \leq 0$. $\dot{L}$ is negative definite outside the ball with radius v described as $\{\|\tilde{x}\| \mid \|\tilde{x}\| > v\}$. Thus, $\|\tilde{x}\|$ is bounded in this ball, by the rational choice of parameters P and A_C we can keep v get any small value. Thus, we assume $\tilde{x}_3$ and $\tilde{x}_4$ can be ignored. $\hat{x}_3$ and $\hat{x}_4$ can substitute for x_3 and x_4.

$$\begin{cases} \dot{x}_1 = x_2 \\ \dot{x}_2 = f_2(\overline{x}_2) + \frac{K}{I}(\hat{x}_3) \\ \dot{\hat{x}}_3 = \hat{x}_4 \\ \dot{\hat{x}}_4 = f_4(\overline{x}_4) + \frac{1}{J}u \end{cases} \tag{39}$$

140 We rewrite $\overline{x}_4 = [x_1, x_2, \hat{x}_3, \hat{x}_4]^T$. Since the model of the single joint flexible manipulator has a fourth order, there are four steps in the design [35]:

step 1: First notice the first subsystem

$$\dot{x}_1 = x_2 \tag{40}$$

let us define the desired trajectory as y_d. then we can define tracking error of the first subsystem

$$e_1 = x_1 - y_d. \tag{41}$$

Define the first virtual control variable x_{2d} as

$$x_{2d} = -c_1 e_1 + \dot{y}_d \tag{42}$$

where c_1 is the design constant. Then we let x_{2d} pass through a first-order low-pass filter, we obtain a new variable x_{2c} to be the expecting variable for the next step

$$\tau_2 \dot{x}_{2c} + x_{2c} = x_{2d},\ x_{2c}(0) = x_{2d}(0). \tag{43}$$

The time constant τ_2 is the design constant. Then we can obtain tracking error of the second subsystem

$$e_2 = x_2 - x_{2c} \tag{44}$$

And the derivative of e_1 can be obtained

$$\begin{aligned} \dot{e}_1 &= \dot{x}_1 - \dot{y} \\ &= x_2 - \dot{y}_d \\ &= -c_1 e_1 + e_2 + (x_{2c} - x_{2d}) \end{aligned} \tag{45}$$

Notice that there is an error $x_{2c} - x_{2d}$ in the above equation, to remove the effect of it we define the compensating variable α_1 as

$$\dot{\alpha}_1 = -c_1 \alpha_1 + \alpha_2 + (x_{2c} - x_{2d}), \alpha_1(0) = 0 \tag{46}$$

where α_2 will be defined in the next step. Then we can obtain tracking error of the compensating variable

$$v_1 = e_1 - \alpha_1 \tag{47}$$

and

$$v_2 = e_2 - \alpha_2 \tag{48}$$

step 2: Considering the second subsystem and using RBF neural network to approximate the unknown function $f_2(\overline{x}_2)$

$$\dot{x}_2 = f_2(\overline{x}_2) + \frac{K}{I}\hat{x}_3 = W_2^T S_2(\overline{x}_2) + \epsilon_2(\overline{x}_2) + \frac{K}{I}\hat{x}_3 \tag{49}$$

where W_2 is the optimal weight matrix, S_2 is the basis function of RBF neural network which is a Gaussian function and $\epsilon_2(\overline{x}_2)$ is the approximation error of RBF neural network with $\|\epsilon_2(\overline{x}_2)\| \leq \epsilon_2$. We define the second virtual control variable x_3^d as

$$\dot{x}_{3d} = \frac{I}{K}\left[-\hat{W}_2^T S(\overline{x}_2) - c_2 e_2 - e_1 + \dot{x}_{2c}\right] \tag{50}$$

where c_2 is a design constant, $\hat{W}_2$ is the estimation of W_2. Then we let x_{3d} pass through a first-order low-pass filter, we obtain a new variable x_{3c} to be the expecting variable for the next step

$$\tau_3 \dot{x}_{3c} + x_{3c} = x_{2d},\ x_{3c}(0) = x_{3d}(0). \tag{51}$$

The time constant τ_3 is the design constant. Then we can obtain tracking error of the third subsystem

$$e_3 = x_3 - x_{3c} \tag{52}$$

And the derivative of e_2 can be obtained

$$\begin{aligned} \dot{e}_2 &= \dot{x}_2 - \dot{x}_{2c} \\ &= \tilde{W}_2^T S(\overline{x}_2) + \epsilon(\overline{x}_2) - c_2 e_2 - e_1 + \frac{K}{I}\left[e_3 + (x_{3c} - x_{3d})\right] \end{aligned} \tag{53}$$

where $\tilde{W}_2 = W_2 - \hat{W}_2$. Notice that there is an error $x_{3c} - x_{3d}$ in the above equation, to remove the effect of it we define the compensating variable α_2 as

$$\dot{\alpha}_2 = -c_2\alpha_2 - \alpha_1 + \frac{K}{I}\alpha_3 + \frac{K}{I}(x_{3c} - x_{3d}), \alpha_2(0) = 0 \tag{54}$$

where α_3 will be defined in the next step. Then we can obtain tracking error of the compensating variable

$$v_3 = e_3 - \alpha_3 \tag{55}$$

We define the prediction error

$$\alpha_{2NN} = x_2 - \hat{x}_2 \tag{56}$$

where $\hat{x}_2$ is defined as

$$\dot{\hat{x}}_2 = \hat{W}_2^T S_2(\overline{x}_2) + \frac{K}{I}\hat{x}_3 + \beta_2\alpha_{2NN}, \hat{x}_2(0) = x_2(0) \tag{57}$$

with $\beta_2 >$ is a design constant. we choose the update law of $\hat{W}_2$ as

$$\dot{\hat{W}}_2 = \Gamma_2\left[(v_2 + \Gamma_{z2}\alpha_{2NN})S_2(\tilde{x}_2) - \rho_2\hat{W}_2\right] \tag{58}$$

where $\Gamma_2 > 0$,$\Gamma_{z2} > 0$ and $\rho_2 > 0$ are design constants.

step 3: Then notice the third subsystem

$$\dot{\hat{x}}_3 = \hat{x}_4 \tag{59}$$

Define the third virtual control variable x_{4d} as

$$x_4^d = -c_3e_3 - \frac{K}{I}e_2 + \dot{x}_3^c \tag{60}$$

where c_3 is the design constant. Then we let x_{4d} pass through a first-order low-pass filter, we obtain a new variable x_{4c} to be the expecting variable for the next step

$$\tau_4\dot{x}_{4c} + x_{4c} = x_{4d},\ x_{4c}(0) = x_{4d}(0). \tag{61}$$

The time constant τ_4 is the design constant. Then we can obtain tracking error of the fourth subsystem

$$e_4 = \hat{x}_4 - x_{4c} \tag{62}$$

And the derivative of e_3 can be obtained

$$\begin{aligned}\dot{e}_3 &= \dot{\hat{x}}_3 - \dot{x}_{3c} \\ &= -c_3 e_3 - \frac{K}{I} e_2 + e_4 + (x_{4c} - x_{4d})\end{aligned} \tag{63}$$

Notice that there is an error $x_{4c} - x_{4d}$ in the above equation, to remove the effect of it we define the compensating variable α_3 as

$$\dot{\alpha}_3 = -c_3\alpha_3 - \frac{K}{I}\alpha_2 + \alpha_4 + (x_{4c} - x_{4d}), \alpha_3(0) = 0 \tag{64}$$

where α_4 will be defined in the next step. Then we can obtain tracking error of the compensating variable

$$v_4 = e_4 - \alpha_4 \tag{65}$$

step 4: Considering the fourth subsystem and using RBF neural network to approximate the unknown function $f_4(\overline{x}_4)$

$$\dot{\hat{x}}_4 = f_4(\overline{x}_4) + \frac{1}{J}u = W_4^T S_4(\overline{x}_4) + \epsilon_4(\overline{x}_4) + \frac{1}{J}u \tag{66}$$

where W_4 is the optimal weight matrix, S_4 is the basis function of RBF neural network which is a Gaussian function and $\epsilon_4(\overline{x}_4)$ is the approximation error of RBF neural network with $\|\epsilon_4(\overline{x}_4)\| \leq \epsilon_4$. We define the final control variable u as

$$u = J\left[-\hat{W}_4^T S(\overline{x}_4) - c_4 e_4 - e_3 + \dot{x}_{4c}\right] \tag{67}$$

where c_4 is a design constant, $\hat{W}_4$ is the estimation of W_4. Then we can obtain the derivative of e_4

$$\begin{aligned}\dot{e}_4 &= \dot{\hat{x}}_4 - \dot{x}_{4c} \\ &= \tilde{W}_4^T S(\overline{x}_4) + \epsilon(\overline{x}_4) - c_4 e_4 - e_3\end{aligned} \tag{68}$$

where $\tilde{W}_4 = W_4 - \hat{W}_4$. The compensating variable α_4 is defined as

$$\dot{\alpha}_4 = -c_4\alpha_4 - \alpha_3, \alpha_4(0) = 0 \tag{69}$$

We define the prediction error

$$\alpha_{4NN} = \hat{x}_4 - \hat{\hat{x}}_4 \tag{70}$$

where $\hat{\hat{x}}_4$ is defined as

$$\dot{\hat{\hat{x}}}_4 = \hat{W}_4^T S_4(\overline{x}_4) + \frac{1}{J}u + \beta_4\alpha_{4NN}, \hat{\hat{x}}_4(0) = \hat{x}_4(0) \tag{71}$$

with $\beta_4 >$ is a design constant. we choose the update law of $\hat{W}_4$ as

$$\dot{\hat{W}}_4 = \Gamma_4\left[(v_4 + \Gamma_{z4}\alpha_{4NN})S_4(\overline{x}_4) - \rho_4\hat{W}_4\right] \tag{72}$$

where $\Gamma_4 > 0$,$\Gamma_{z4} > 0$ and $\rho_4 > 0$ are design constants.

3.3. Stability analysis

145 *Theorem 2*: Considering the single-joint flexible joint manipulator described by (31), the dynamic surface controller described by (67), the observer described by (12) and the adaptive update rates of weights described by (15),(58) and (72). For any bounded initial conditions, there exists suitable parameters G, c_i, β_i ($i = 1, 2, 3, 4$) and Γ, Γ_2, Γ_4, ρ, ρ_2, ρ_4 such that the proposed control scheme
150 guarantees:

1) all the signals in the controller system are uniformly ultimately bounded;
2) the tracking errors v_i and e_i converges to a arbitrarily small neighborhood of zero.

Proof: *Definition 3*: The Lyapunov function of the controller is

$$H = \frac{1}{2}\sum_{i=1}^{4} v_i^2 + \frac{1}{2}\sum_{i=2,4} \tilde{W}_{2i}^T\Gamma_I^{-1}\tilde{W}_{2i} + \frac{1}{2}\sum_{i=2,4}\Gamma_{zi}\alpha_{iNN}^2 + \frac{1}{2}\tilde{x}^T P\tilde{x} + \frac{1}{2}tr(\tilde{W}^T\Gamma^{-1}\tilde{W}). \tag{73}$$

Consider the time-derivative of the tracking error, we can get

$$
\begin{aligned}
\dot{v}_1 =& \dot{e}_1 - \dot{\alpha}_1 \\
=& -c_1(e_1 - \alpha_1) + (e_2 - \alpha_2) \\
=& -c_1 v_1 + v_2 && (74) \\
\dot{v}_2 =& \dot{e}_2 - \dot{\alpha}_2 \\
=& \tilde{W}_2^T S_2(\overline{x}_2) + \epsilon_2(\overline{x}_2) - c_2(e_2 - \alpha_2) - (e_1 - \alpha_1) + \frac{K}{I}(e_3 - \alpha_3) \\
=& \tilde{W}_2^T S_2(\overline{x}_2) + \epsilon_2(\overline{x}_2) - c_2 v_2 - v_1 + \frac{K}{I} v_3 && (75) \\
\dot{v}_3 =& \dot{e}_3 - \dot{\alpha}_3 \\
=& -c_3(e_3 - \alpha_3) - \frac{K}{I}(e_2 - \alpha_2) + (e_4 - \alpha_4) \\
=& -c_3 v_3 - \frac{K}{I} v_2 + v_4 && (76) \\
\dot{v}_4 =& \dot{e}_4 - \dot{\alpha}_4 \\
=& \tilde{W}_4^T S_4(\overline{x}_4) + \epsilon_4(\overline{x}_4) - c_4(e_4 - \alpha_4) - (e_3 - \alpha_3) \\
=& \tilde{W}_4^T S_4(\overline{x}_4) + \epsilon_4(\overline{x}_4) - c_4 v_4 - v_3 && (77) \\
\dot{\alpha}_{2NN} =& \dot{x}_2 - \dot{\hat{x}}_2 && (78) \\
=& \tilde{W}_2^T + \epsilon_2(\overline{x}_2) - \beta_2 \alpha_{2NN}^2 && (79) \\
\dot{\alpha}_{4NN} =& \dot{x}_4 - \dot{\hat{x}}_4 && (80) \\
=& \tilde{W}_4^T + \epsilon_4(\overline{x}_4) - \beta_4 \alpha_{4NN}^2 && (81)
\end{aligned}
$$

Then we obtain the time-derivative of the Lyapunov function

$$
\begin{aligned}
\dot{H} =& \frac{1}{2}\tilde{x}^T P \dot{\tilde{x}} + \frac{1}{2}\dot{\tilde{x}}^T P \tilde{x} + tr(\tilde{W}^T \Gamma^{-1} \dot{\hat{W}}) + \sum_{i=1}^{4} v_i \dot{v}_i + \sum_{i=2,4} \Gamma_{zi} \alpha_{iNN} \dot{\alpha}_{iNN} \\
& - \sum_{i=2,4} \tilde{W}_i^T \Gamma_i^{-1} \dot{\hat{W}}_i \\
=& \sum_{i=1}^{4} (-c_i v_i^2) + \sum_{i=2,4} \left[v_i \epsilon_i(\overline{x}_i) + \Gamma_{zi} \alpha_{iNN} \epsilon_i(\overline{x}_i) - \Gamma_{zi} \beta_i \alpha_{iNN}^2 - \rho_i \hat{W}_i^T \tilde{W} \right] + \dot{L} \\
& && (82)
\end{aligned}
$$

According to the stability analysis of observer, we can see if

$$\|\tilde{x}\| > \frac{2Pw_m + 2K}{\lambda_{min}(Q)} = v \tag{83}$$

then

$$\dot{L} < 0. \tag{84}$$

Then we obtain

$$\dot{H} < \sum_{i=1}^{4}(-c_i v_i^2) + \sum_{i=2,4}\left[\Gamma_{zi}\alpha_{iNN}\epsilon_i(\overline{x}_i) + v_i\epsilon_i(\overline{x}_i) - \Gamma_{zi}\beta_i\alpha_{iNN}^2 - \rho_i\tilde{W}_i^T\tilde{W}\right] \tag{85}$$

Let us define

$$V = \frac{1}{2}\sum_{i=1}^{4} v_i^2 + \frac{1}{2}\sum_{i=2,4}\tilde{W}_{2i}^T\Gamma_I^{-1}\tilde{W}_{2i} + \frac{1}{2}\sum_{i=2,4}\Gamma_{zi}\alpha_{iNN}^2 \tag{86}$$

We can obtain if

$$\|\tilde{x}\| > \frac{2Pw_m + 2K}{\lambda_{min}(Q)} = v \tag{87}$$

then $\dot{H} < \dot{V}$. Consider the following inequality

$$v_i\epsilon_i(\tilde{x}_i) - c_i v_i^2 \leq -c_i\left(v_i - \frac{\epsilon_i(\tilde{x}_i)}{2c_i}\right)^2 + \frac{1}{4c_i}\epsilon_i(\tilde{x}_i)^2 \tag{88}$$

$$\alpha_{iNN}\epsilon_i(\tilde{x}_i) - \beta_i\alpha_{iNN}^2 \leq -\beta\left(\alpha_{iNN} - \frac{\epsilon_i(\tilde{x}_i)}{2\beta_i}\right)^2 + \frac{1}{4\beta_i}\epsilon_i(\tilde{x}_i)^2 \tag{89}$$

$$\tilde{W}_i^T W_i - \tilde{W}_i^T\tilde{W}_i \leq -\left\|\tilde{W}_i - \frac{W}{2}\right\|^2 + \frac{1}{4}\|W_i\|^2 \tag{90}$$

Then we can obtain

$$\dot{V} \leq -\sum_{i=1,3} c_i v_i - \sum_{i=2,4}\left[c_{min}\left(v_i - \frac{\epsilon_i(\tilde{x}_i)}{2c_i}\right)^2 + \Gamma_{zmin}\beta_{min}\left(z_{iNN} - \frac{\epsilon_i(\tilde{x}_i)}{2\beta_i}\right)^2 + \rho_{min}\left\|\tilde{W}_i - \frac{W}{2}\right\|^2\right] + \frac{1}{2c_{min}}\epsilon_{max}^2 + \frac{\Gamma_{zmax}}{2\beta_{main}}\epsilon_{max}^2 + \frac{\rho_{max}}{2}W_{max}^2 \tag{91}$$

where $c_{min} = min[c_i]$, $\beta_{min} = min[\beta]$, $\Gamma_{zmin} = min[\Gamma_{zi}]$, $\rho_{min} = min[\rho_i]$, $\Gamma_{zmax} = max[\Gamma_{zi}]$, $W_{max} = max[\|W\|]$ and $\rho_{max} = max[\rho_i]$. Let us defined

$$D = \frac{1}{2c_{min}}\epsilon_{max}^2 + \frac{\Gamma_{zmax}}{2\beta_{main}}\epsilon_{max}^2 + \frac{\rho_{max}}{2}W_{max}^2 \tag{92}$$

If

$$\left|v_i - \frac{\epsilon_i(\tilde{x}_i)}{2c_i}\right| \geq \sqrt{\frac{D}{c_{min}}} \tag{93}$$

or

$$\left|\alpha_{iNN} - \frac{\epsilon_i(\tilde{x}_i)}{2\beta_i}\right| \geq \sqrt{\frac{D}{\Gamma_{zmin}\beta_{min}}} \tag{94}$$

or

$$\left\|\tilde{W}_i - \frac{\tilde{W}_i}{2}\right\| \geq \sqrt{\frac{D}{\rho_{min}}} \tag{95}$$

then $\dot{H} \leq \dot{V} \leq 0$. Thus, we can obtain v_i, α_{iNN} and $\|\tilde{W}_i\|$ are bounded in the sets defined as follows

$$\Omega_{v_i} = \left(v_i \middle| |v_i| \leq |\sqrt{\frac{D}{c_{min}}} + \frac{\epsilon_{max}}{2c_{min}}\right) \tag{96}$$

$$\Omega_{\alpha_{iNN}} = \left(\alpha_{iNN} \middle| |\alpha_{iNN}| \leq |\sqrt{\frac{D}{\Gamma_{zmin}\beta_{min}}} + \frac{\epsilon_{max}}{2\beta_{min}}\right) \tag{97}$$

$$\Omega_{\tilde{W}_i} = \left(\tilde{W}_i \middle| \|\tilde{W}_i\| \leq |\sqrt{\frac{D}{\rho_{min}}} + \frac{W_{max}}{2}\right) \tag{98}$$

Moreover $\tilde{x}$ is bounded in the set defined as $\{\|\tilde{x}\| \mid \|\tilde{x}\| > v\}$ and $\tilde{W}$ is also bounded. Thus, all the signal in the system is bounded. If we choose the value of c_i and β_i big enough then the error of v_i and α_{iNN} is arbitrarily small.

4. SIMULATION

To demonstrate the effectiveness of the controller , we perform simulation studies with the following model of a robot manipulator. First of all, we select the parameters of the manipulator system as follows:

$L = 1m$, $M = 2kg$, $I = 2kg \cdot m^2$, $J = 0.5kg \cdot m^2$, $K = 10N \cdot m/rad$, $g = 9.8m/s^2$.

The reference trajectory is

$$\begin{cases} \dot{x}_{d1} = x_{d2} \\ \dot{x}_{d2} = 2sin(\frac{3}{2}t) - \frac{1}{2}x_{d1} - \frac{3}{2}x_{d2} \\ y_d = x_{d1} \end{cases} \tag{99}$$

Results of simulation are shown as follows.

First we remove the state observer to examine the performance of the controller.

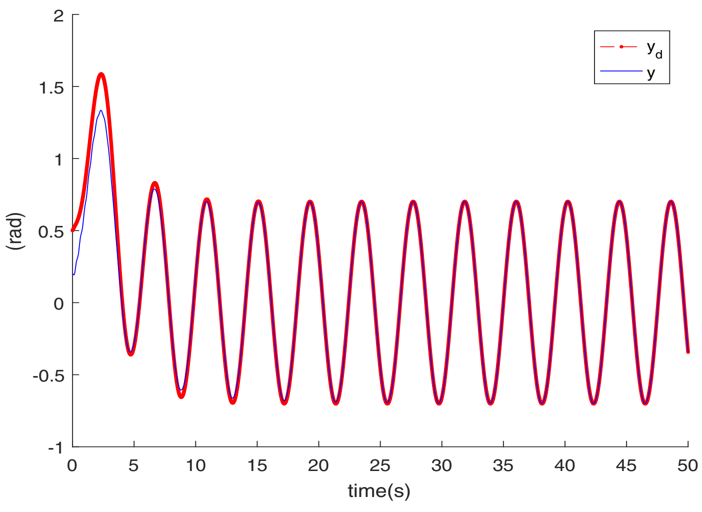


Figure 4: The relationship between reference trajectory y_d and actual output y

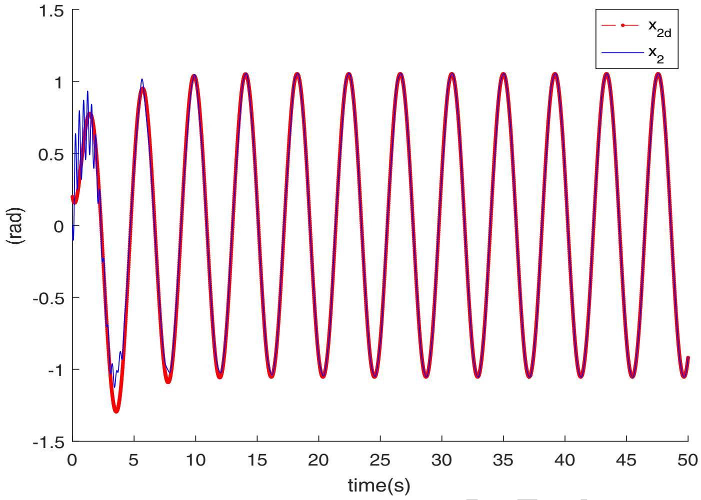


Figure 5: The relationship between reference trajectory x_{d2} and actual state variable x_2

Fig. 4 and 5 are the relationship between reference trajectory y_d and x_{d2} with actual variables y and x_2. It's clear that the actual output y and x_2 can track reference trajectory y_d and x_{d2} accurately and quickly.

Then we assume that x_3 and x_4 is not measurable and use the estimate $\hat{x}_3$ and $\hat{x}_4$ from state observer instead of x_3 and x_4. Results of simulation are shown as follows.

Figs. 6-9 is the relationship between the state variable x and estimated value $\hat{x}$. From Figs. 6-9 we can see the states of the observer follow those of the actual system accurately and quickly. Thus, the tracking performance of the observer is satisfactory. Fig. 10 and 11 are the relationship between reference trajectory y_d and x_{d2} with actual variables y and x_2. We can see the actual output y and x_2 can track reference trajectory y_d and x_{d2} as well as the simulation which running without the action of state observer. Thus, we can confirm the reliable performance of the observer and controller. Figs. 12-14 is the weights of neural network. It can be seen the weights of neural network converge to the optimal value in a short time. To sum up, the proposed controller based on neural network observer is able to achieve good transient

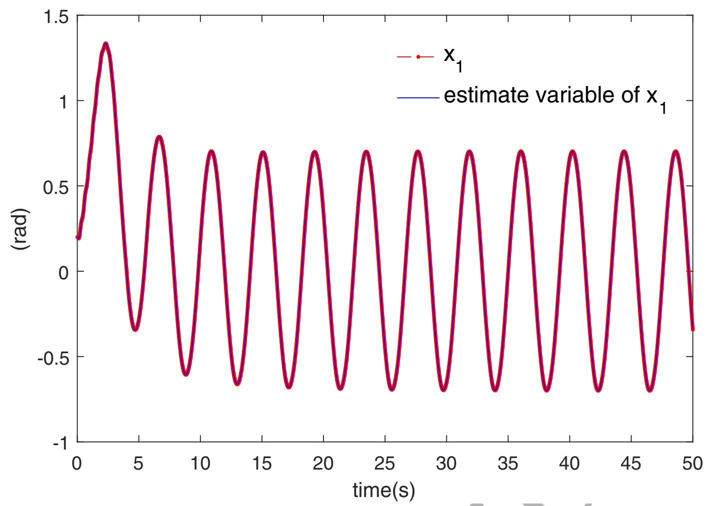


Figure 6: The relationship between state variable x_1 and estimated value $\hat{x}_1$

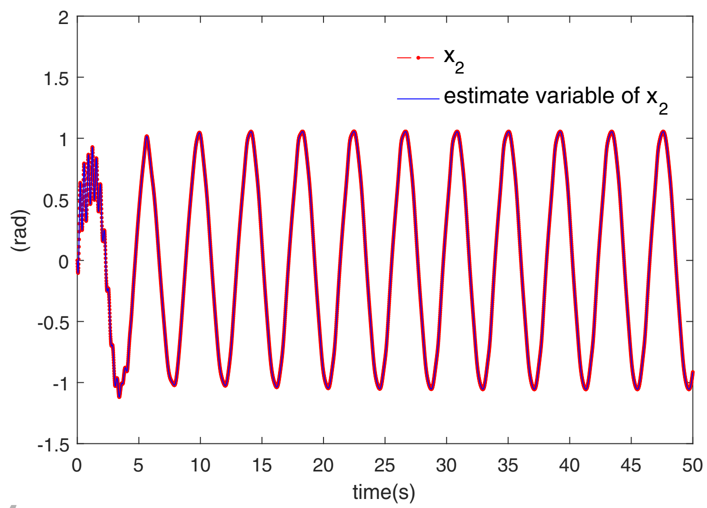


Figure 7: The relationship between state variable x_2 and estimated value $\hat{x}_2$

tracking performance of tracking errors in the presence of unknown model.

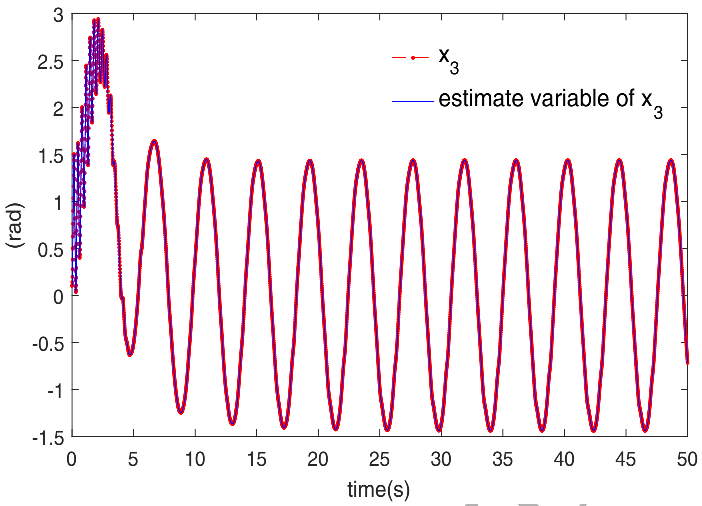


Figure 8: The relationship between state variable x_3 and estimated value $\hat{x}_3$

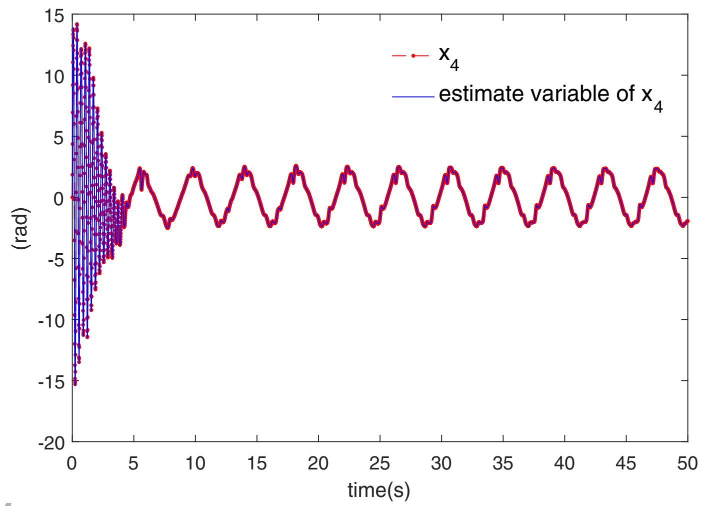


Figure 9: The relationship between state variable x_4 and estimated value $\hat{x}_4$

5. CONCLUSION

185 A controller for flexible joint manipulator with unknown model is designed in this paper. We design an observer based on RBF neural network to estimate the

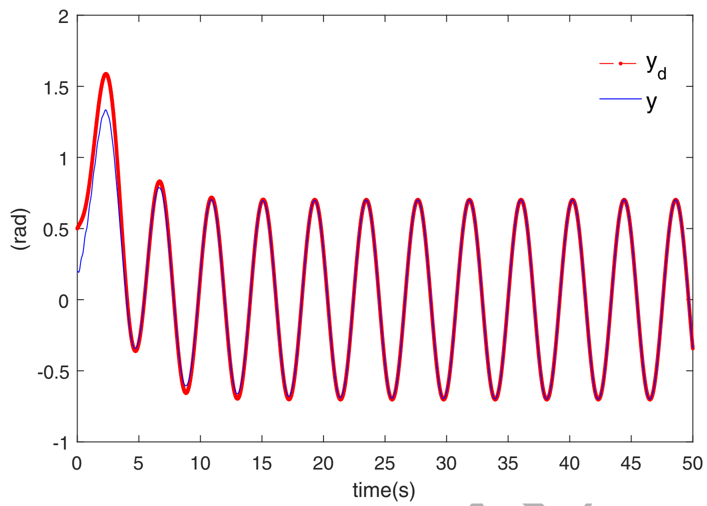


Figure 10: The relationship between reference trajectory y_d and actual output y

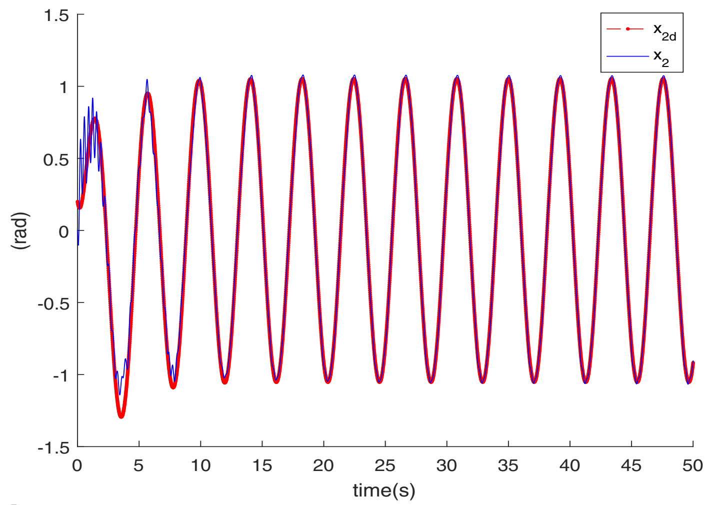


Figure 11: The relationship between reference trajectory x_{d2} and actual state variable x_2

state variables unmeasurable of the robot manipulator. We design the controller based on dynamic surface control method. RBF neural network is used to construct the unknown model. Stability and performance of the overall system combining observer and controller are rigorously established by the Lapunov

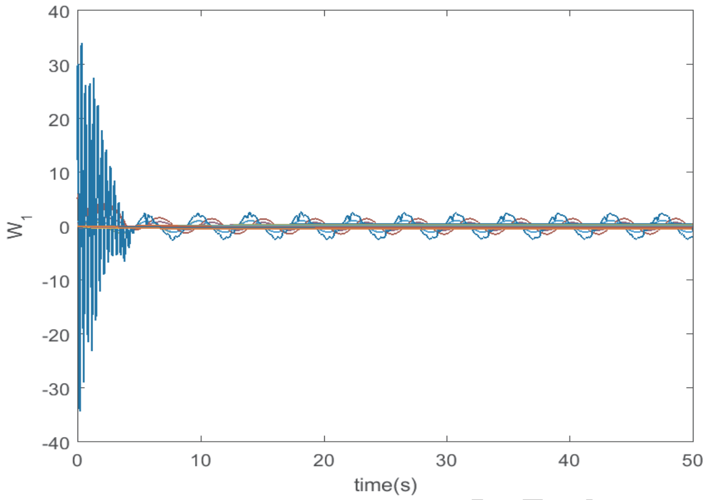


Figure 12: Weights of neural network W_1

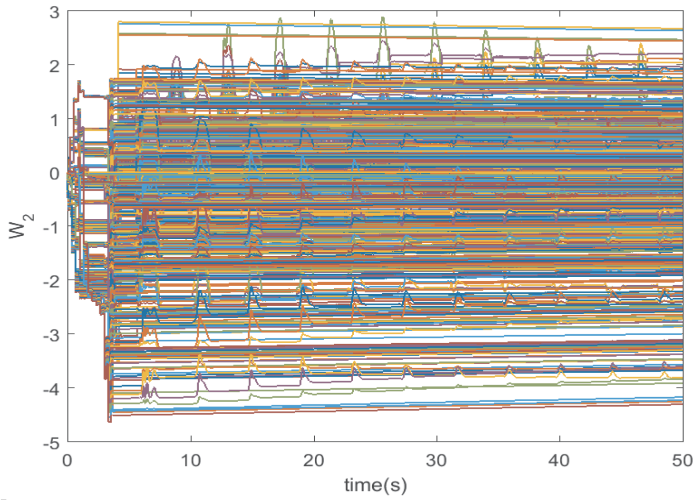


Figure 13: Weights of neural network W_2

method. Simulation studies are performed to test and verify the effectiveness of the proposed controller. Simulation results from an application to a single link flexible joint manipulator confirm the reliable performance of the proposed controller.

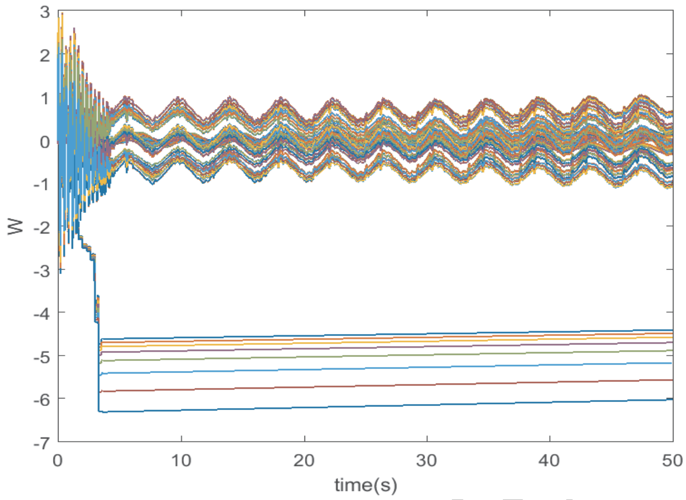


Figure 14: Weights of neural network W

References

[1] L. Wang, Collaborative robot monitoring and control for enhanced sustainability, The International Journal of Advanced Manufacturing Technology 81 (9) (2015) 1433–1445.

[2] C. Blackman, P. Desruelle, A helping hand for europe: the competitive outlook for the eu robotics industry, Economic Modelling 19 (1) (2010) 65–90.

[3] H. Ding, M. Schipper, B. Matthias, Collaborative behavior design of industrial robots for multiple human-robot collaboration, in: International Symposium on Robotics, 2013, pp. 1–6.

[4] A. Bicchi, G. Tonietti, Fast and soft-arm tactics [robot arm design], IEEE Robotics & Amp Amp Automation Magazine 11 (2) (2004) 22–33.

[5] J. Choi, S. Hong, W. Lee, S. Kang, M. Kim, A robot joint with variable stiffness using leaf springs, IEEE Transactions on Robotics 27 (2) (2011) 229–238.

[6] D. Navarro-Alarcon, Z. Wang, H. M. Yip, Y. H. Liu, A method to regulate the torque of flexible-joint manipulators with velocity control inputs, in: IEEE International Conference on Robotics and Biomimetics, 2014.

[7] W. He, Y. Chen, Z. Yin, Adaptive neural network control of an uncertain robot with full-state constraints, IEEE Transactions on Cybernetics 46 (3) (2016) 620–629.

[8] M. Chen, G. Tao, Adaptive fault-tolerant control of uncertain nonlinear large-scale systems with unknown dead zone., IEEE Transactions on Cybernetics 46 (8) (2015) 1851–1862.

[9] M. Chen, S. S. Ge, Adaptive neural output feedback control of uncertain nonlinear systems with unknown hysteresis using disturbance observer, IEEE Transactions on Industrial Electronics 62 (12) (2015) 7706–7716.

[10] M. Chen, P. Shi, C. C. Lim, Robust constrained control for mimo nonlinear systems based on disturbance observer, IEEE Transactions on Automatic Control 60 (12) (2015) 3281–3286.

[11] W. He, Y. Ouyang, J. Hong, Vibration control of a flexible robotic manipulator in the presence of input deadzone, IEEE Transactions on Industrial Informatics.

[12] W. He, S. Zhang, Control design for nonlinear flexible wings of a robotic aircraft, IEEE Transactions on Control Systems Technology 25 (1) (2017) 351–357.

[13] W. He, Y. Dong, Adaptive fuzzy neural network control for a constrained robot using impedance learning, IEEE Transactions on Neural Networks and Learning Systems.

[14] W. He, S. S. Ge, Cooperative control of a nonuniform gantry crane with constrained tension, Automatica 66 (2016) 146–154.

[15] D. Swaroop, J. C. Gerdes, P. P. Yip, J. K. Hedrick, Dynamic surface control of nonlinear systems, in: American Control Conference, 1997, pp. 3028–3034 vol.5.

[16] D. Swaroop, J. K. Hedrick, P. P. Yip, J. C. Gerdes, Dynamic surface control for a class of nonlinear systems, IEEE Transactions on Automatic Control 45 (10) (2000) 1893–1899.

[17] P. Goldsmith, B. Francis, A. Goldenberg, Stability of hybrid position/force control applied to manipulators with flexible joints, International Journal of Robotics and Automation 14 (4) (1999) 146–160.

[18] C. Yang, Y. Jiang, Z. Li, W. He, C.-Y. Su, Neural control of bimanual robots with guaranteed global stability and motion precision PP (99) (2016) 1–1.

[19] C. Yang, X. Wang, L. Cheng, H. Ma, Neural-learning-based telerobot control with guaranteed performance., IEEE Transactions on Cybernetics (2016) 1–12.

[20] C. Yang, Z. Li, R. Cui, B. Xu, Neural network-based motion control of an underactuated wheeled inverted pendulum model., IEEE Transactions on Neural Networks and Learning Systems 25 (11) (2014) 2004–2016.

[21] C. Yang, X. Wang, Z. Li, Y. Li, C. Y. Su, Teleoperation control based on combination of wave variable and neural networks, IEEE Transactions on Systems, Man, and Cybernetics: Systems PP (99) (2016) 1–12.

[22] M. Wang, A. Yang, Dynamic learning from adaptive neural control of robot manipulators with prescribed performance PP (99) (2017) 1–12.

[23] S. L. Dai, M. Wang, C. Wang, Neural learning control of marine surface vessels with guaranteed transient tracking performance, IEEE Transactions on Industrial Electronics 63 (3) (2015) 1–1.

[24] S. Nicosia, P. Tomei, A. Tornamb, An approximate observer for a class of nonlinear systems, Systems and Control Letters 13 (1) (1989) 43–51.

[25] S. Nicosia, A. Tornamb, High-gain observers in the state and parameter estimation of robots having elastic joints, Systems and Control Letters 13 (4) (1989) 331–337.

[26] M. S. Ahmed, S. H. Riyaz, Dynamic observersa neural net approach, Journal of Intelligent and Fuzzy Systems Applications in Engineering and Technology 9 (1,2) (2000) 113–127.

[27] Y. H. Kim, F. L. Lewis, C. T. Abdallah, Nonlinear observer design using dynamic recurrent neural networks, in: IEEE Conference on Decision and Control, 1997, pp. 949–954 vol.1.

[28] Q. Wu, M. Saif, Neural adaptive observer based fault detection and identification for satellite attitude control systems 2 (2005) 8–10.

[29] F. Abdollahi, H. A. Talebi, R. V. Patel, A stable neural network observer with application to flexible-joint manipulators, in: International Conference on Neural Information Processing, 2002, pp. 1910–1914 vol.4.

[30] X. Duan, Analysis of function approximation ability of neural, Fuzzy Systems and Mathematics (4) (1998) 79–84.

[31] J. Yi, Y. Hou, Intelligent Control Technology - Serialized Teaching Material of Control Courses in Higher Engineering Colleges, Beijing University of Technology Press, 1999.

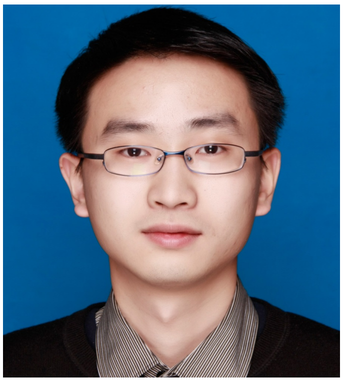
[32] X. Wang, J. Yuan, J. Shi, Y. Liu, A review of modeling theory and control methods for flexible manipulators, robot 24 (1) (2002) 86–91.

[33] Y. Gai, Design and research of flexible arm experiment system, Ph.D. thesis, Northeastern University (2009).

[34] A. C. Huang, Y. C. Chen, Adaptive sliding control for single-link flexible-joint robot with mismatched uncertainties, IEEE Transactions on Control Systems Technology 12 (5) (2004) 770–775.

[35] M. Wang, C. Wang, Learning from adaptive neural dynamic surface control of strict-feedback systems., IEEE Transactions on Neural Networks and Learning Systems 26 (6) (2015) 1247–1259.

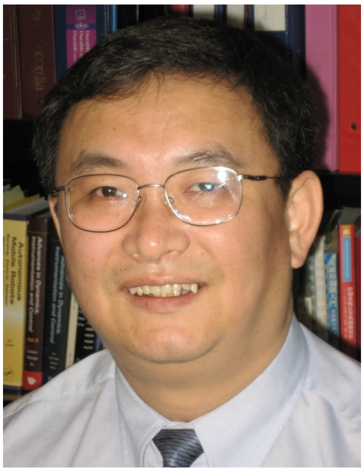
Xin Liu received the B.Eng. degree in automation from the South China University of Technology, Guangzhou, China, in 2016, and is currently pursuing the M.S. degree in the South China University of Technology, Guangzhou, China. His research interests include human-robot interaction, robotics and intelligent control.



Chenguang Yang received the B.Eng. degree in measurement and control from Northwestern Polytechnical University, Xian, China, in 2005, and the Ph.D. degree in control engineering from the National University of Singapore, Singapore, in 2010. He received postdoctoral training at Imperial College London, UK. He is a senior lecturer with Zienkiewicz Centre for Computational Engineering, Swansea University, UK. He is also with South China Univeristy of Technology, China. His research interests lie in robotics, automation and computational intelligence.

Zhiguang Chen received the B.Eng. degree in automation from the South China University of Technology, Guangzhou, China, in 2016, and is currently pursuing the M.S. degree in the South China University of Technology, Guangzhou, China. His research interests include human-robot interaction, robotics and intelligent control.

Min Wang (M09) received the B.Sc. and M.Sc. degrees from the Department of Mathematics, Bohai University, Jinzhou, China, in 2003 and 2006, and the Ph.D. degree from the Institute of Complexity Science, Qingdao University, Qingdao, China, in 2009. She was a Visiting Research Student with the Department of Electrical and Computer Engineering, National University of Singapore, Singapore, in 2008. She is currently an Associate Professor with the School of Automation Science and Engineering, South China University of Technology, Guangzhou, China. She has authored and co-authored about 20 papers in top international journals. Her current research interests include intelligent control, deterministic learning theory, and pattern-based control. Dr. Wang was a recipient of the Excellent Doctoral Dissertations Award of Shandong Province in 2010 and the Outstanding Graduate Award for Technological Innovation of Shandong Province in 2009.



Chun-Yi Su received the Ph.D. degree in control engineering from the South China University of Technology, Guangzhou, China, in 1990. He joined Concordia University, Montreal, QC, Canada, in 1998, after a seven-year stint with the University of Victoria, Victoria, BC, Canada. He is currently with the College of Automation Science and Engineering, South China University of Technology, on leave from Concordia University. His current research interests include the application of automatic control theory to mechanical systems. He is particularly interested in control of systems involving hysteresis nonlinearities. He has authored or co-authored over 300 publications in journals, book chapters, and conference proceedings. Dr. Su has served as an Associate Editor of the IEEE TRANSACTIONS ON AUTOMATIC CONTROL, the IEEE TRANSACTIONS ON CONTROL SYSTEMS TECHNOLOGY, and the Journal of Control Theory and Applications. He has been on the Editorial Board of 18 journals, including the IFAC Journal of Control Engineering Practice and Mechatronics.