

# Motion Control of Two-Link Flexible-Joint Robot, Using Backstepping, Neural Networks, and Indirect Method

Withit Chatlatanagulchai and Peter H. Meckl, *Member, IEEE*

**Abstract**—We present a state-feedback control of a two-link flexible-joint robot. The control algorithm does not require the mathematical model representing the robot. Three-layer neural networks approximate the unknown plant functions. The neural network weights are adapted on-line. We use backstepping control structure together with variable structure control to provide robustness to all uncertainties. We have included experimental results to show the effectiveness of the control algorithm.

## I. INTRODUCTION

WE are interested in motion control of a flexible-joint robot for several reasons. First, the joint flexibility exists in most robots. It arises from driving components such as actuators, gear teeth, or transmission belts. In some applications, the designers incorporate flexible joints into their products intentionally to absorb impact force and to reduce damage to the parts from accidental collision. Second, control designers should explicitly include joint flexibility in their design because joint resonant frequencies, which are located within the control bandwidth, can be excited and cause severe oscillations. The experiment in [1] suggested that the designers should consider joint flexibility in both modeling and control design.

Controller design of two-link flexible-joint robot is challenging due to two main reasons. First, its Euler-Lagrange model is much more complicated than those of rigid-joint or one-link flexible-joint robot. Second, the number of degrees of freedom is twice the number of control inputs. The control inputs do not directly apply to the links. Instead, the control inputs directly apply to the motors that connect to the links via flexible-joint dynamics. This results in the loss of some important structural properties that apply for rigid-joint robots, such as matching property between nonlinearities and the inputs, passivity from inputs to link velocities.

There exist some well-established control designs for flexible-joint robots. In [2], they transform the dynamic model of the flexible-joint robot into the standard singular perturbation model, by using link position as slow variable and joint torque as fast variable. Controller is a composite of slow and fast control. Slow-control input, which adds damping to the system, drives the closed-loop system to a

quasi-steady state system that has the structure of a rigid-joint robot. Then, fast-control input can be designed using available techniques for the rigid-joint robot. To avoid having to measure the joint torque signal, you may consult [3] for the design of an observer. Reference [4] shows an alternative singular perturbation model by using tracking error of motor shaft as fast variable. Reference [5] extends the work in [2] to the case where model uncertainties exist in the system. They use radial basis function networks to estimate unknown functions, and use discontinuous variable-structure controller to provide the closed-loop system with robustness for the estimation errors.

In this paper, we consider the trajectory-tracking task of a two-link flexible-joint robot in the horizontal plane. The controller algorithm does not require a closed-form mathematical model of the robot. We use a three-layer neural network to estimate each unknown function. Uncertainties come from the estimation of plant functions, actuator nonlinearities, and external disturbance. We use a smooth version of variable structure controller to provide robustness to the system, under the assumption that all uncertainties are bounded by unknown bounds. Backstepping structure provides a way to insert control effort to each subsystem.

We are able to control the trajectory of the robot effectively using link angular position, link angular velocity, motor angular position, and motor angular velocity. Existing work usually requires, in addition to the quantities above, link angular acceleration and jerk, or flexible-joint torque.

Throughout this paper, unless otherwise specified, we have the following definition.

*Definition 1:* We denote by  $\|\bullet\|$  any suitable norm. When it is required to be specific, we denote any p-norm by  $\|\bullet\|_p$ . The symbol  $\|\bullet\|_F$  denotes the Frobenius norm, i.e. given a matrix  $A$ , the Frobenius norm is given by  $\|A\|_F^2 = \text{tr}(A^T A) = \sum a_{ij}^2$ .

We organize this paper as follows. Section II contains some information on the robot and the experiment setup. Section III contains three-layer neural network background and controller design. Section IV contains experimental results. Section V is a conclusion of the paper.

## II. A TWO-LINK FLEXIBLE-JOINT ROBOT

Fig. 1 depicts a robot, for which we are designing the controller. The robot operates in the horizontal plane, has two links and two motors. Input torque  $u_1$  is applied to the first motor, which drives the first sprocket through a chain. The sprocket is attached to the first link via the first torsional spring that provides joint flexibility. The second motor is

Manuscript received January 31, 2005.

The authors are with Motion and Vibration Control Laboratory, School of Mechanical Engineering, Purdue University, West Lafayette, IN 47907-2088 USA (e-mail: chatlata@purdue.edu, meckl@purdue.edu; phone: 765-494-0539; fax: 765-494-5686).

situated on the first link. Input torque  $u_2$  is applied to the second motor which drives the second sprocket. The second sprocket is attached to the second link via the second torsional spring. Note that the second motor's shaft does not share the same axis with the axis of rotation of the second link. This setting is more practical than the shared-axis cases commonly treated in existing literature.

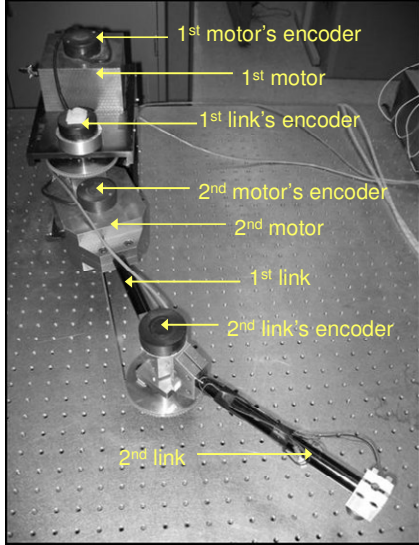


Fig. 1. Photograph of the two-link flexible-joint robot in our laboratory.

There are four optical encoders; each measures angular positions of the two links and the two motors. Angular velocities are obtained from numerical differentiation of the position signals. Two current amplifiers supply current to the two motors.

Fig. 2 depicts the overall experiment setup. We use Labview 7.1, Labview Real-Time Module, and Labview FPGA Module to perform hardware-in-the-loop experiment. The data acquisition board is National Instruments' PCI-7831R.

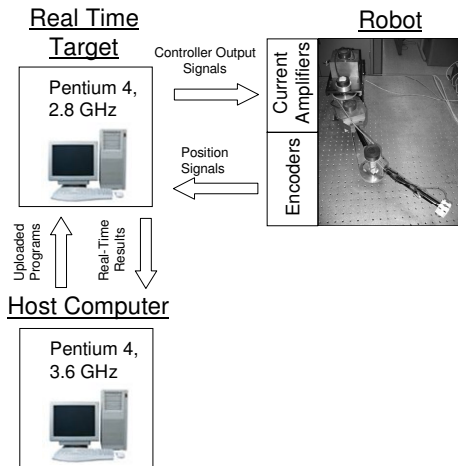


Fig. 2. Diagram showing overall experiment setup.

We let  $\theta_1$  be absolute angular position of the first link,  $\theta_2$  be relative angular position of the second link,  $\theta_3$  be absolute angular position of the first motor, and  $\theta_4$  be

relative angular position of the second motor. If we let  $x_1 = [x_{11}, x_{12}]^T = [\theta_1, \theta_2]^T$  and  $x_2 = [x_{21}, x_{22}]^T = [\dot{\theta}_1, \dot{\theta}_2]^T$  be state vectors representing link position and velocity,  $x_3 = [x_{31}, x_{32}]^T = [\theta_3, \theta_4]^T$  and  $x_4 = [x_{41}, x_{42}]^T = [\dot{\theta}_3, \dot{\theta}_4]^T$  be state vectors representing motor position and velocity, and  $u = [u_1, u_2]^T$  be input vector representing input torque, the equations of motion of this robot can be put in the following state space form:

$$\begin{aligned} \dot{x}_1 &= x_2 + d_{a1}(\bar{x}_4), \\ \dot{x}_2 &= f_2(\bar{x}_2) + g_2(\bar{x}_2)(x_3 + d_{a2}(\bar{x}_4)), \\ \dot{x}_3 &= x_4 + d_{a3}(\bar{x}_4), \\ \dot{x}_4 &= f_4(\bar{x}_4) + g_4(\bar{x}_4)(u + d_{a4}(\bar{x}_4)), \\ y &= x_1, \end{aligned} \quad (1)$$

where  $\bar{x}_i = \{x_{i1}, x_{i2}, \dots, x_{i1}, x_{i2}\}$ ;  $f_2, f_4 \in \mathbb{R}^2$  and  $g_2, g_4 \in \mathbb{R}^{2 \times 2}$  are vectors and matrices that contain smooth functions;  $d_{ai}(\bar{x}_4) = [d_{ai1}, d_{ai2}]^T$  is an additive disturbance vector that may depend on all states.

In the next section, we will design a controller for the system in the form of (1), where  $f_2, f_4, g_2$ , and  $g_4$  are unknown. The disturbance  $d_{ai}$  is bounded by an unknown constant.

### III. CONTROLLER DESIGN

#### A. Three-Layer Neural Network

Each unknown scalar function is estimated by a three-layer neural network in Fig. 3.

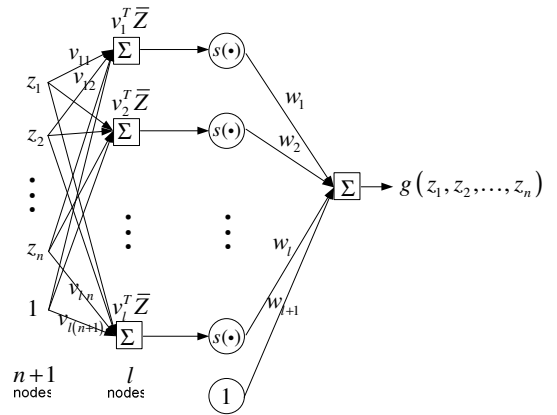


Fig. 3. A three-layer neural network. The square represents node whose output contains adjustable parameters.

Variables in the network can be defined as follows:

$$\begin{aligned} \bar{Z} &= [z_1, z_2, \dots, z_n, 1]^T \in \mathbb{R}^{n+1}, \\ V &= [v_1, v_2, \dots, v_l] \in \mathbb{R}^{(n+1) \times l}, \\ v_i &= [v_{i1}, v_{i2}, \dots, v_{i(n+1)}]^T \in \mathbb{R}^{n+1}, i = 1, 2, \dots, l, \\ S(V^T \bar{Z}) &= [s(v_1^T \bar{Z}), s(v_2^T \bar{Z}), \dots, s(v_l^T \bar{Z}), 1]^T \in \mathbb{R}^{l+1}, \end{aligned}$$

$$W = [w_1, w_2, \dots, w_l, w_{l+1}]^T \in \mathbb{R}^{l+1},$$

$$g(W, V, z_1, z_2, \dots, z_n) = W^T S(V^T \bar{Z}) \in \mathbb{R}.$$

$s(\cdot)$  can be any appropriate activation function that is a non-constant, bounded and monotone increasing continuous function. We use a sigmoid function

$$s(x) = 1/(1 + e^{-x}), \forall x \in \mathbb{R}.$$

This network can uniformly approximate any scalar-valued continuous function to any arbitrary accuracy with some constant ideal weights  $W^*, V^*$ , and some appropriate number of hidden-layer nodes,  $l^*$ , as was proved in [6]. From the universal approximation property, we have

$$g(z_1, z_2, \dots, z_n) = W^{*T} S(V^{*T} \bar{Z}) + \varepsilon, \quad (2)$$

where  $\|\varepsilon\| < \varepsilon_U$  is approximation error with unknown  $\varepsilon_U > 0$  provided that  $g(\cdot)$  is defined on a compact set  $\Omega_z$ . Note that the foregoing statement is only to assure the existence of ideal weights and ideal number of hidden-layer nodes. The appropriate number of hidden-layer nodes, in practice, can be found from trial and error with the problem in consideration. The ideal weights generally are unknown. However, in system identification application, the ideal weights are typically assumed constant and bounded as in the following Assumption.

*Assumption 1:* On the compact set  $\Omega_z$ , the ideal neural network weights  $W^*, V^*$  are constant and bounded by  $\|W^*\| \leq W_U, \|V^*\|_F \leq V_U, i = 1, \dots, m$ , where  $W_U$  and  $V_U$  are unknown.

Note that the neural network weight  $V$  appears nonlinearly. According to [7], approximators that are nonlinear in their parameters can achieve the same level of approximation accuracy as those that are linear and usually require fewer number of adjusting parameters. However, since the parameter appears nonlinearly, the parameter-tuning law is usually more complicated.

Since ideal weights are unknown, let  $\hat{W}$  and  $\hat{V}$  be the estimates of  $W^*$  and  $V^*$ , respectively. The estimate of the function  $g$  is given by

$$\hat{g}(z_1, z_2, \dots, z_n) = \hat{W}^T S(\hat{V}^T \bar{Z}). \quad (3)$$

By using Lemma 3.6 in [8], the differences between neural network outputs with ideal and estimated weights are given by

$$\begin{aligned} \hat{W}^T S(\hat{V}^T \bar{Z}) - W^{*T} S(V^{*T} \bar{Z}) &= \tilde{W}^T (\hat{S} - \hat{S}' \hat{V}^T \bar{Z}) \\ &\quad + \hat{W}^T \hat{S}' \tilde{V}^T \bar{Z} + d_u, \end{aligned} \quad (4)$$

where  $\tilde{W} = \hat{W} - W^*, \tilde{V} = \hat{V} - V^*, \hat{S} = S(\hat{V}^T \bar{Z}) \in \mathbb{R}^{l+1}$ ,

$$\hat{S}' = \text{diag}\{\hat{s}'_1, \hat{s}'_2, \dots, \hat{s}'_l, 0\} \in \mathbb{R}^{(l+1) \times (l+1)},$$

$$\hat{s}'_i = s'(z_a) = \left. \frac{d[s(z_a)]}{dz_a} \right|_{z_a = \hat{v}_i^T \bar{z}} \in \mathbb{R}, i = 1, 2, \dots, l,$$

$$s(x) = 1/(1 + e^{-x}), \forall x \in \mathbb{R}.$$

The residual term  $d_u$  is bounded by

$$|d_u| \leq \|V^*\|_F \|\bar{Z}\| \|\hat{W}^T \hat{S}'\| + \|W^*\| \|\hat{S}' \hat{V}^T \bar{Z}\| + \|W^*\|_1. \quad (5)$$

Note that  $\tilde{W}$  and  $\tilde{V}$  appear linearly in (4). This is

important since from Assumption 1,  $\dot{\tilde{W}} = \dot{\hat{W}}$  and  $\dot{\tilde{V}} = \dot{\hat{V}}$ , therefore the weight adaptation laws can be easily designed using this linear structure.

### B. Backstepping and Variable Structure Controller

In this section, we design a controller that can achieve globally uniformly ultimately bounded tracking error; the ultimate bound can be made arbitrarily small by adjusting design parameters. We require the following assumptions.

*Assumption 2:* The additive disturbance  $d_{aik}(\bar{x}_4)$ , where  $i = 1, \dots, 4, k = 1, 2$ , is bounded by  $\|d_{aik}(\bar{x}_4)\| < d_{aikU}$ , where  $d_{aikU}$  is an unknown constant.

*Assumption 3:* There exists a known constant  $g_{ijkU} > 0$  such that  $\|g_{ijk}(\cdot)\| \leq g_{ijkU}, \forall i = 2, \forall j = 1, 2, \forall k = 1, 2$ .

*Assumption 4:* The desired trajectory  $x_{1d}$  is sufficiently smooth.

The control objective is to make output  $y = x_1$  follow desired trajectory  $x_{1d}$  as closely as possible, while all the signals in the closed-loop system are bounded. For convenience, we drop arguments of some functions where appropriate.

In backstepping design, we try to reduce the error between actual state and desired state of each subsystem – having the tracking error be the error of the first subsystem. Let  $e_i = [e_{i1}, e_{i2}]^T = x_i - x_{id}, i = 1, \dots, 4$  be those errors.

*Step 1:*

Let the virtual control law of the first subsystem be

$$x_{2d} = -c_1 e_1 + \dot{x}_{1d} + u_{2dvsc} = [x_{21d}, x_{22d}]^T,$$

where  $c_1$  is a design parameter,  $u_{2dvsc}$  is variable structure control law to be designed. From Assumption 2, we have the following inequality

$$\sum_{k=1}^2 \{\|d_{aik}\|\} \leq \sum_{k=1}^2 \{\|d_{aikU}\|\}.$$

We let the variable structure control law be in the form  $u_{2dvsc} = [u_{2dvsc1}, u_{2dvsc2}]^T$ , where

$$u_{2dvscj} = -\hat{K}_{1j} \left( \frac{2}{\pi} \arctan \left( \frac{e_{1j}}{\mu_{1j}} \right) \right),$$

$\mu_{1j}$  is a small positive design parameter,  $\hat{K}_{1j}$  approximates

$$\sum_{k=1}^2 \{\|d_{aikU}\|\}.$$

The time derivative of the error of the first subsystem becomes

$$\dot{e}_1 = \dot{x}_1 - \dot{x}_{1d} = (x_2 + d_{a1}) - \dot{x}_{1d} = e_2 - c_1 e_1 + u_{2dvsc} + d_{a1}.$$

*Step 2:*

Let the virtual control law of the second subsystem be

$$x_{3d} = -\hat{g}_2^{-1} [e_1 + c_2 e_2 + \hat{f}_2 - \dot{x}_{2d} - u_{3dvsc}] = [x_{31d}, x_{32d}]^T.$$

From (2), (5), Assumption 2, and Assumption 3, we have

$$\begin{aligned} |d_{uf2j}| + |e_{f2j}| + \sum_{k=1}^2 \{ |d_{ug2jk} x_{3dk}| \} \\ + \sum_{k=1}^2 \{ |e_{g2jk} x_{3dk}| \} + \sum_{k=1}^2 \{ |g_{2jk} d_{a2k}| \} \end{aligned} \leq K_{2j}^{*T} \phi_{2j},$$

where

$$K_{2j}^* = \left[ \left\| V_{f_{2j}}^* \right\|_F, \left\| W_{f_{2j}}^* \right\|, \left\| W_{f_{2j}}^* \right\|_1 + \varepsilon_{f_{2jU}} + \sum_{k=1}^2 \{g_{2jkU} d_{a2kU}\}, \right. \\ \left. \sum_{k=1}^2 \left\{ \left\| V_{g_{2jk}}^* \right\|_F \right\}, \sum_{k=1}^2 \left\{ \left\| W_{g_{2jk}}^* \right\| \right\}, \sum_{k=1}^2 \left\{ \left\| W_{g_{2jk}}^* \right\|_1 \right\} + \sum_{k=1}^2 \left\{ \varepsilon_{g_{2jkU}} \right\} \right]^T, \\ \phi_{2j} = \left[ \left\| \bar{Z}_{f_{2j}} \hat{W}_{f_{2j}}^T \hat{S}'_{f_{2j}} \right\|_F, \left\| \hat{S}'_{f_{2j}} \hat{V}_{f_{2j}}^T \bar{Z}_{f_{2j}} \right\|, 1, \right. \\ \left. \sum_{k=1}^2 \left\{ \left\| \bar{Z}_{g_{2jk}} \hat{W}_{g_{2jk}}^T \hat{S}'_{g_{2jk}} x_{3dk} \right\|_F \right\}, \sum_{k=1}^2 \left\{ \left\| \hat{S}'_{g_{2jk}} \hat{V}_{g_{2jk}}^T \bar{Z}_{g_{2jk}} x_{3dk} \right\| \right\}, \right. \\ \left. \sum_{k=1}^2 \left\{ \left\| x_{3dk} \right\| \right\} \right].$$

The variable structure control law is given by

$$u_{3dvsc} = [u_{3dvsc1}, u_{3dvsc2}]^T \in \mathbb{R}^2,$$

where

$$\bar{\phi}_{2j} = -\hat{K}_{2j}^T \bar{\phi}_{2j}, \\ \left[ \begin{array}{c} \left\| \bar{Z}_{f_{2j}} \hat{W}_{f_{2j}}^T \hat{S}'_{f_{2j}} \right\|_F \frac{2}{\pi} \arctan \left( \frac{e_{2j}}{\mu_{2j}} \left\| \bar{Z}_{f_{2j}} \hat{W}_{f_{2j}}^T \hat{S}'_{f_{2j}} \right\|_F \right) \\ \left\| \hat{S}'_{f_{2j}} \hat{V}_{f_{2j}}^T \bar{Z}_{f_{2j}} \right\| \frac{2}{\pi} \arctan \left( \frac{e_{2j}}{\mu_{2j}} \left\| \hat{S}'_{f_{2j}} \hat{V}_{f_{2j}}^T \bar{Z}_{f_{2j}} \right\| \right) \\ \frac{2}{\pi} \arctan \left( \frac{e_{2j}}{\mu_{2j}} \right) \\ \sum_{k=1}^2 \left\{ \left\| \bar{Z}_{g_{2jk}} \hat{W}_{g_{2jk}}^T \hat{S}'_{g_{2jk}} x_{3dk} \right\|_F \right\} \frac{2}{\pi} \\ \bullet \arctan \left( \frac{e_{2j}}{\mu_{2j}} \sum_{k=1}^2 \left\{ \left\| \bar{Z}_{g_{2jk}} \hat{W}_{g_{2jk}}^T \hat{S}'_{g_{2jk}} x_{3dk} \right\|_F \right\} \right) \\ \sum_{k=1}^2 \left\{ \left\| \hat{S}'_{g_{2jk}} \hat{V}_{g_{2jk}}^T \bar{Z}_{g_{2jk}} x_{3dk} \right\| \right\} \frac{2}{\pi} \\ \bullet \arctan \left( \frac{e_{2j}}{\mu_{2j}} \sum_{k=1}^2 \left\{ \left\| \hat{S}'_{g_{2jk}} \hat{V}_{g_{2jk}}^T \bar{Z}_{g_{2jk}} x_{3dk} \right\| \right\} \right) \\ \sum_{k=1}^2 \left\{ \left\| x_{3dk} \right\| \right\} \frac{2}{\pi} \arctan \left( \frac{e_{2j}}{\mu_{2j}} \sum_{k=1}^2 \left\{ \left\| x_{3dk} \right\| \right\} \right) \end{array} \right]$$

$\mu_{2j}$  is a small positive design parameter,  $\hat{K}_{2j}$  approximates  $K_{2j}^*$ . The time derivative of the error of the second subsystem becomes

$$\dot{e}_2 = \dot{x}_2 - \dot{x}_{2d} \\ = (f_2 + g_2(x_3 + d_{a2})) - \dot{x}_{2d} + \hat{g}_2 x_{3d} - \hat{g}_2 x_{3d} + g_2 x_{3d} - g_2 x_{3d} \\ = f_2 - \dot{x}_{2d} + \hat{g}_2 [-\hat{g}_2^{-1} [e_1 + c_2 e_2 + \hat{f}_2 - \dot{x}_{2d} - u_{3dvsc}]] + g_2 d_{a2} \\ + (g_2 - \hat{g}_2) x_{3d} + g_2 (x_3 - x_{3d}) \\ = \begin{bmatrix} \varepsilon_{f_{21}} - \tilde{W}_{f_{21}}^T (\hat{S}_{f_{21}} - \hat{S}'_{f_{21}} \hat{V}_{f_{21}}^T \bar{Z}_{f_{21}}) - \hat{W}_{f_{21}}^T \hat{S}'_{f_{21}} \hat{V}_{f_{21}}^T \bar{Z}_{f_{21}} \\ -d_{uf_{21}} \\ \varepsilon_{f_{22}} - \tilde{W}_{f_{22}}^T (\hat{S}_{f_{22}} - \hat{S}'_{f_{22}} \hat{V}_{f_{22}}^T \bar{Z}_{f_{22}}) - \hat{W}_{f_{22}}^T \hat{S}'_{f_{22}} \hat{V}_{f_{22}}^T \bar{Z}_{f_{22}} \\ -d_{uf_{22}} \end{bmatrix}$$

$$+ \begin{bmatrix} \varepsilon_{g_{211}} - \tilde{W}_{g_{211}}^T (\hat{S}_{g_{211}} & \varepsilon_{g_{112}} - \tilde{W}_{g_{112}}^T (\hat{S}_{g_{112}} \\ -\hat{S}'_{g_{211}} \hat{V}_{g_{211}}^T \bar{Z}_{g_{211}}) & -\hat{S}'_{g_{112}} \hat{V}_{g_{112}}^T \bar{Z}_{g_{112}}) \\ -\hat{W}_{g_{211}}^T \hat{S}'_{g_{211}} \hat{V}_{g_{211}}^T \bar{Z}_{g_{211}} & -\hat{W}_{g_{112}}^T \hat{S}'_{g_{112}} \hat{V}_{g_{112}}^T \bar{Z}_{g_{112}} \\ -d_{ug_{211}} & -d_{ug_{112}} \\ \varepsilon_{g_{121}} - \tilde{W}_{g_{121}}^T (\hat{S}_{g_{121}} & \varepsilon_{g_{122}} - \tilde{W}_{g_{122}}^T (\hat{S}_{g_{122}} \\ -\hat{S}'_{g_{121}} \hat{V}_{g_{121}}^T \bar{Z}_{g_{121}}) & -\hat{S}'_{g_{122}} \hat{V}_{g_{122}}^T \bar{Z}_{g_{122}}) \\ -\hat{W}_{g_{121}}^T \hat{S}'_{g_{121}} \hat{V}_{g_{121}}^T \bar{Z}_{g_{121}} & -\hat{W}_{g_{122}}^T \hat{S}'_{g_{122}} \hat{V}_{g_{122}}^T \bar{Z}_{g_{122}} \\ -d_{ug_{121}} & -d_{ug_{122}} \end{bmatrix} x_{3d} \\ -e_1 - c_2 e_2 + u_{3dvsc} + g_2 d_{a2} + g_2 (x_3 - x_{3d}).$$

Step 3:

Let the virtual control law of the third subsystem be

$$x_{4d} = -g_{2U} e_2 - c_3 e_3 + \dot{x}_{3d} + u_{4dvsc} = [x_{41d}, x_{42d}]^T,$$

$$\text{where } g_{2U} = \begin{bmatrix} g_{21U} & g_{212U} \\ g_{221U} & g_{222U} \end{bmatrix}.$$

Similar to Step 1, we let the variable structure control be in the form  $u_{4dvsc} = [u_{4dvsc1}, u_{4dvsc2}]^T$ , where

$$u_{4dvscj} = -\hat{K}_{3j} \left( \frac{2}{\pi} \arctan \left( \frac{e_{3j}}{\mu_{3j}} \right) \right),$$

The time derivative of the error of the third subsystem is given by

$$\dot{e}_3 = -g_{2U} e_2 - c_3 e_3 + e_4 + u_{4dvsc} + d_{a3}.$$

Step 4:

Let the desired control law be

$$u = -\hat{g}_4^{-1} [e_3 + c_4 e_4 + \hat{f}_4 - \dot{x}_{4d} - u_{5dvsc}] = [u_1, u_2]^T.$$

The time derivative of the error of the last subsystem,  $\dot{e}_4$ , can be derived similar to Step 2.

The variable structure control law,  $u_{5dvsc}$ , remains similar to that in Step 2 by replacing  $x_{3dk}$  with  $u_k$ .

We use the following  $\sigma$ -modification weight updating laws:

$$\dot{\hat{W}}_{fij} = \Gamma_{wfij} [(\hat{S}_{fij} - \hat{S}'_{fij} \hat{V}_{fij}^T \bar{Z}_{fij}) z_{ij} - \sigma_{wfij} \hat{W}_{fij}],$$

$$\dot{\hat{V}}_{fij} = \Gamma_{vfij} [\bar{Z}_{fij} \hat{W}_{fij}^T \hat{S}'_{fij} z_{ij} - \sigma_{vfij} \hat{V}_{fij}],$$

$$\dot{\hat{W}}_{gijk} = \Gamma_{wgijk} [(\hat{S}_{gijk} - \hat{S}'_{gijk} \hat{V}_{gijk}^T \bar{Z}_{gijk}) x_{(i+1)dk} z_{ij} - \sigma_{wgijk} \hat{W}_{gijk}],$$

$$\dot{\hat{V}}_{gijk} = \Gamma_{vgijk} [\bar{Z}_{gijk} \hat{W}_{gijk}^T \hat{S}'_{gijk} x_{(i+1)dk} z_{ij} - \sigma_{vgijk} \hat{V}_{gijk}],$$

$$\dot{\hat{K}}_{ij} = \Gamma_{kij} [\bar{\phi}_{ij} z_{ij} - \sigma_{kij} \hat{K}_{ij}],$$

where  $\Gamma_{wfij}, \Gamma_{vfij}, \Gamma_{wgijk}, \Gamma_{vgijk}, \Gamma_{kij} > 0$ ;  $i = 2, 4$ ;  $j = 1, 2$ ;  $k = 1, 2$ . The  $\sigma > 0$  terms in the update laws are used to prevent  $\hat{W}, \hat{V}, \hat{K}$  from growing unboundedly by maintaining their values around their initial values.

**Definition 2:** We define  $(\hat{\cdot}) = (\hat{\cdot}) - (\tilde{\cdot})$  where  $(\hat{\cdot})$  is the actual or ideal value,  $(\tilde{\cdot})$  is estimated error, and  $(\hat{\cdot})$  is the estimated value.

Using a Lyapunov function

$$V = \sum_{i=1}^4 \left\{ \frac{1}{2} e_i^T e_i + \sum_{j=1}^2 \sum_{k=1}^2 \left[ \frac{1}{2} \tilde{W}_{gijk}^T \Gamma_{wgijk}^{-1} \tilde{W}_{gijk} + \frac{1}{2} \text{tr}(\tilde{V}_{gijk}^T \Gamma_{vgijk}^{-1} \tilde{V}_{gijk}) \right] + \sum_{j=1}^2 \left[ \frac{1}{2} \tilde{W}_{fj}^T \Gamma_{wfj}^{-1} \tilde{W}_{fj} + \frac{1}{2} \text{tr}(\tilde{V}_{fj}^T \Gamma_{vfj}^{-1} \tilde{V}_{fj}) \right] + \frac{1}{2} \tilde{K}_{ij}^T \Gamma_{kij}^{-1} \tilde{K}_{ij} \right\},$$

and the following facts

$$2\tilde{W}^T \hat{W} = \|\tilde{W}\|^2 + \|\hat{W}\|^2 - \|W^*\|^2 \geq \|\tilde{W}\|^2 - \|W^*\|^2,$$

$$2\text{tr}\{\tilde{V}^T \hat{V}\} = \|\tilde{V}\|_F^2 + \|\hat{V}\|_F^2 - \|V^*\|_F^2 \geq \|\tilde{V}\|_F^2 - \|V^*\|_F^2,$$

$$0 \leq |\alpha| - \alpha \frac{2}{\pi} \arctan\left(\frac{\alpha}{\mu}\right) \leq 0.2785\mu, \forall \alpha \in \mathbb{R},$$

and after some straightforward but lengthy derivation, we obtain the derivative of the Lyapunov function as

$$\dot{V} \leq -\zeta V + \delta,$$

where  $\zeta > 0$  and  $\delta \geq 0$ .

From this point on, it can be shown using standard nonlinear analysis that the error trajectories  $e, \tilde{K}, \tilde{W}, \tilde{V}$  are globally uniformly ultimately bounded. The ultimate bound, all-time exponential-decay upper bound, as well as the time when the trajectories enter the ultimate bound can also be found, but are omitted here.

#### IV. EXPERIMENTAL RESULTS

All design parameters are as follows:

$$l = 3, \Gamma_{wf} = \Gamma_{vf} = \Gamma_{wg} = \Gamma_{vg} = 10, \Gamma_k = 1, c_i = 5,$$

$$\sigma_{wf} = \sigma_{vf} = \sigma_{wg} = \sigma_{vg} = \sigma_k = 0.1, \mu = 1.$$

All initial values are set to 0.1. Sampling period is 10 ms. The desired trajectory is obtained from passing square wave signal of amplitude 3 and 40-second period into the filter  $1/(s+2)^3$ .

Experimental results are given in Fig. 4. The control system achieves good overall tracking performance as can be seen from the results in parts (a) and (b). Both link angular positions  $\theta_1$  and  $\theta_2$  are able to follow their desired trajectories quite closely. Parts (c) to (f) show estimated values of the unknown functions. Parts (g) and (h) are control inputs to the two current amplifiers. Their values can be converted to voltage by multiplying with  $10/2^{16}/2$ .

#### V. CONCLUSION

We present a state-feedback control of a two-link flexible joint robot. The control system consists of three-layer neural networks, backstepping controller, and variable structure controller.

The controller achieves good tracking performance. However, we designed the controller based on the assumption that the actual plant is in a nonlinear form (1). The actual robot may not belong exactly to this form, which may degrade the controller performance. The future work includes finding the effect of this situation.

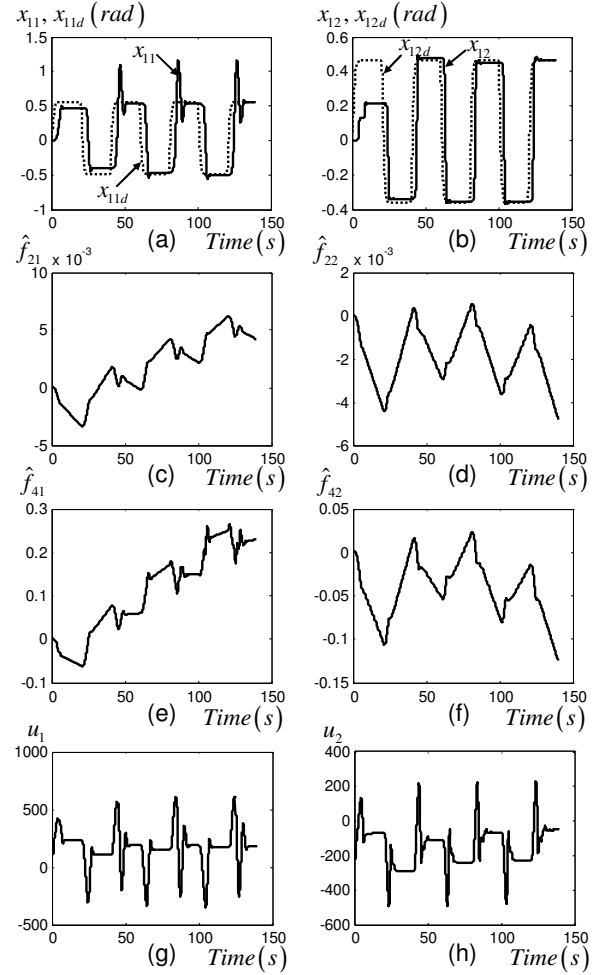


Fig. 4. Experiment result in 140 seconds. (a)  $\theta_1$  versus its desired trajectory  $\theta_{1d}$ . (b)  $\theta_2$  versus its desired trajectory  $\theta_{2d}$ . (c) Estimated function  $\hat{f}_{21}$ . (d) Estimated function  $\hat{f}_{22}$ . (e) Estimated function  $\hat{f}_{41}$ . (f) Estimated function  $\hat{f}_{42}$ . (g) Control input  $u_1$ . (h) Control input  $u_2$ .

#### REFERENCES

- [1] L. M. Sweet and M. C. Good, "Re-definition of the robot motion control problem: effects of plant dynamics drive system constraints, and user requirements," *Proc. of 23<sup>rd</sup> IEEE Conf. on Decision and Control*, Las Vegas, NV, 1984, pp. 724-731.
- [2] M. W. Spong, "Modeling and control of elastic joint robots," *Trans. ASME J. Dynamic Systems, Measurement and Control*, vol. 109, no. 4, pp. 310-319, 1987.
- [3] S. Nicosia, P. Tomei, and A. Tornambe, "A nonlinear observer for elastic robots," *IEEE J. Robot. Automat.*, vol. 4, no. 1, pp. 45-52, 1988.
- [4] S. S. Ge, "Adaptive control design for flexible joint manipulators," *Automatica*, vol. 32, no. 2, pp. 273-278, 1996.
- [5] S. S. Ge, T. H. Lee, and C. J. Harris, *Adaptive Neural Network Control of Robotic Manipulators*. Singapore: World Scientific Publishing, 1998, ch. 7.
- [6] K. I. Funahashi, "On the approximate realization of continuous mappings by neural networks," *Neural Networks*, vol. 2, pp. 183-192, 1989.
- [7] A. R. Barron, "Universal approximation bounds for superpositions of a sigmoid function," *IEEE Trans. Inform. Theory*, vol. 39, no. 3, pp. 930-945, 1993.
- [8] S. S. Ge, C. C. Hang, T. H. Lee, and T. Zhang, *Stable Adaptive Neural Network Control*. The Netherlands: Kluwer, 2002.