Proceedings of the 17th World Congress
The International Federation of Automatic Control
Seoul, Korea, July 6-11, 2008

Adaptive Fuzzy Control Based on Fuzzy Neural Network for Uncertain Nonlinear Systems

Y. J. Huang* and T. C. Kuo**

** Department of Electrical Engineering, Yuan Ze University, Chungli, Taiwan (Tel: 886-3-4638800; e-mail: eeyjh@staurn.yzut.edu.tw).*
*** Department of Electrical Engineering, Ching Yun University, Taiwan (Tel: 886-3-4581196; e-mail: tck@mail.cyu.edu.tw)*

Abstract: In this paper, an adaptive fuzzy controller based on fuzzy neural network is proposed for uncertain nonlinear systems. The main advantages are the simple design, no requirement of system model, and release of fixed universal range of fuzzy output. A fuzzy neural network is applied to on-line identify the control system and provide sufficient information of the adaptive laws for the proposed fuzzy controller. Finally, experimental results of a two-link robotic arm are given to verify the effectiveness of the proposed approach.

1. INTRODUCTION

The fuzzy logic theory was first proposed by Zadeh in 1965 (Zadeh, 1965). Its most advantage is that it is very useful for analysis when there is no reliable system model. Because the dynamic equations of systems can not be easily found accurately, fuzzy logic control (FLC) has become popular. A study was presented in the design and stability of a fuzzy control system (Wang, 1997). Later, fuzzy logic control has found wide applied in various applications such as stepping motors, robot manipulators, and helicopters, etc(Lee, and Sul, 1998; Betin, *et al*., 2000; Guo, and Woo, 2003; Lower, 2005).

Conventionally, the fuzzy logic controller is based on expert knowledge to define input-output variables, linguistic variables, fuzzy rules, and universal ranges. The conventional fuzzy logic controller can ensure the system stability, but it can not ensure the global system robustness. In order to choose fuzzy sets, designs have to base on performance requirements and stability regions of control systems. Therefore, there are many studies had combined fuzzy control with adaptive or other robust algorithms to guarantee system robustness (Ham, and Johnson, 2000; Yoo and Ham, 2000; Er, and Chin, 2000; Khaber, *et al*., 2006; Guan, *et al*., 2005; Sharkawy, 2005). Additionally, an adaptive fuzzy controller was designed to be a main controller (Sharkawy, 2005). According to the Lyapunov stability theorem, the adaptive law was derived. However, the requirement of the system matrix is necessary.

In order to satisfy the no requirement of a system model and derive the adadptive laws of the fuzzy controler, a fuzzy neural netwok (FNN) is utlized in this paper. FNN comprises the ability of fuzzy on handling uncertain or unknown knowledge of system model and the ability of neural network in learning (Horikawa, *et al*., 1992; Chen, and Teng, 1995). In this study, we develop a novel adaptive fuzzy logic controller based on an on-line tuning FNN, which is named FNN-AFLC for uncertain nonlinear systems. The on-line tuning FNN can identify the nonlinear system in different circumstances without the knowledge of experts. The adaptive FLC can auto tune the centers of the membership function of output. In summary, with the varying universal range, the proposed FNN-AFLC has additional merits to the conventional FLC such as (a) faster tracking response, (b) better steady system performance, and (c) robustness against system uncertainties and external disturbances.

2. FNN-ADAPTIVE FUZZY LOGIC CONTROL DESIGN

2.1 Fundamental of a fuzzy system

The block diagram of a typical fuzzy system is shown in Fig. 1. There are four parts in a fuzzy system, including fuzzification, fuzzy rule base, fuzzy inference engine, and defuzzification. The system structure includes a fuzzy system with singleton fuzzification, if-then rule base, Mamdani implication, product inference engine, and center average defuzzification. The input variables are first translated to the input fuzzy sets. If-then rules are designated from experts' experience. Then, the fuzzy inference engine maps input fuzzy sets to output fuzzy sets according to the defined fuzzy rules. Finally, the output fuzzy set is translated to a crisp output value through the center average defuzzification. Fig. 2 shows the typical membership function with five rules.

For N input variables and M rules, the crisp output y can be written as

$$y = \frac{\sum_{j=1}^{M} \bar{y}^j \prod_{i=1}^{N} \mu_{A_i^j}(x_i)}{\sum_{j=1}^{M} \prod_{i=1}^{N} \mu_{A_i^j}(x_i)}, \tag{1}$$

where A_i^j is the fuzzy set of j th fuzzy rule of i th input variable, $\bar{y}^j$ is the center of the output membership function of the j th rule.

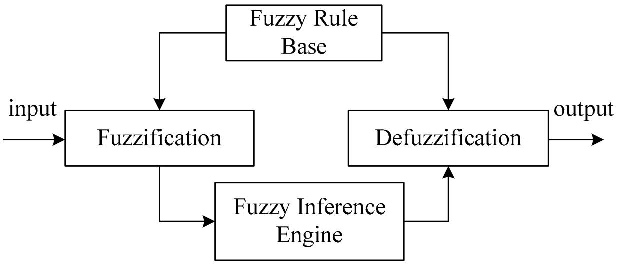

Fig. 1. Block diagram of a fuzzy system.

Note that the universal range of the output membership function is usually fixed and designated by experts in general. However, the set of universal range may not be optimal while the control system possesses uncertainties and external disturbances. Thus, the control performance is not guaranteed. In this paper, an adaptive universal range is proposed. Based on the proposed fuzzy neural network, the adaptive universal range can be tuned automatically until the tracking error approaches zero.

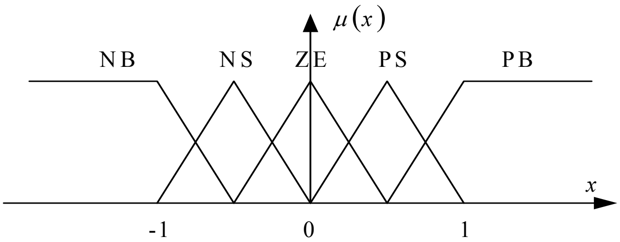

Fig. 2. Membership function (normalized).

2.2 Description of the FNN-AFLC control system

The block diagram of an FNN-ASMC controller for a nonlinear system with two inputs and two outputs is shown in Fig. 3. Generally, to derive proper adaptive control laws for unknown nonlinear systems is difficult. The goal here is to derive a fuzzy controller with adaptive laws without the knowledge of system model. Fuzzy control can be applied to systems without system models. However, fuzzy control is based on experts' experiences to establish fuzzy rules, membership functions, and universal ranges. To further guarantee tracking control performance is difficult while system uncertainties and external disturbances exist.

The proposed FNN-AFLC uses the fuzzy neural networks 1 and 2 to achieve system identification. The advantages of the on-line tuning FNN are the capability of fuzzy control in handling system uncertainties and the capability of neural networks in a learning procedure. Let the control input τ_i be defined as

$$\tau_i = \hat{r}_i \hat{\tau}_i \,,\ i = 1, 2. \tag{2}$$

The gain $\hat{r}_i$ decides the universal range of output under the normalized universal range of output. The FNN can on-line tune parameters through the gradient descent method and supply the optimal parameters to an adaptive regulator. Then, the proposed adaptive regulator can on-line provide the adaptive laws of the gain $\hat{r}_i$ to the i th FLC until the tracking error reaches to zero.

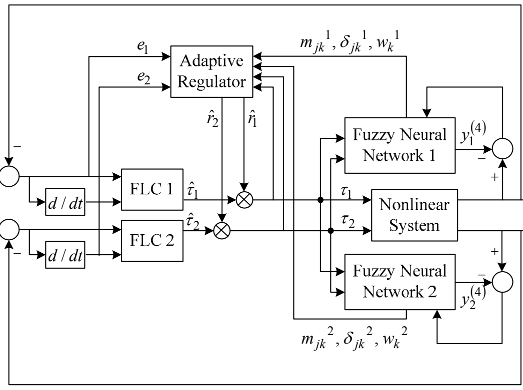

Fig. 3. Block diagram of the FNN-AFLC control system.

2.3 Description of fuzzy neural network

The structure of a four-layer fuzzy neural network which has two inputs and one output is shown in Fig. 4, where nodes in the input layer represent input linguistic variables, nodes in the membership layer represent membership functions, nodes in the rule layer is a fuzzy rule base, and node in the output layer is a crisp output. The output signals of the nodes in each layer are described as follows.

Layer 1: Input layer

In Layer 1, each node is an input node representing an input variable. The input nodes pass signals to the next layer without a computation process. The output of j th node in layer 1, $y_j^{(1)}$, is represented as

$$y_j^{(1)} = u_j,\ j = 1, 2, \tag{3}$$

where u_j is the j th input to the node of Layer 1.

Layer 2: Membership layer

Each node in Layer 2 represents a Gaussian-membership function. The output of k th node in Layer 2 with j th input, $y_{jk}^{(2)}$, is represented as

$$y_{jk}^{(2)} = \exp\left(\frac{-\left(u_j - m_{jk}\right)^2}{\delta_{jk}^{\ 2}}\right), \tag{4}$$

where δ_{jk} and m_{jk} are the standard deviation and mean of the Gaussian function of the k th node with j th input, respectively.

Layer 3: Rule layer

Each node in Layer 3 is a multiplication function, which performs a fuzzy logic rule. The output of k th node in layer 3, $y_k^{(3)}$, is obtained from multiplying incoming signals and represented as

$$y_k^{(3)} = \prod_j y_{jk}^{(2)} . \tag{5}$$

In other words, $y_k^{(3)}$ is the firing strength of k th rule node.

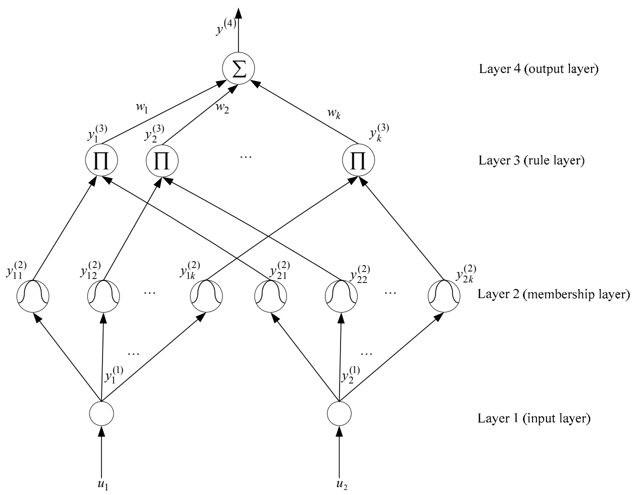

Fig. 4. Architecture of a fuzzy neural network.

Layer 4: Output layer

The node in Layer 4 is a summation function, which sums all incoming signals. The output in Layer 4, $y^{(4)}$, is obtained from summing incoming signals and represented as

$$y^{(4)} = \sum_k y_k^{(3)} w_k , \tag{6}$$

where w_k is the weight of k th link and represents output strength of the k th rule node.

2.4 FNN On-line learning algorithm

In Fig. 3, there are two FNNs. The purpose of FNN is to execute system identification of an unknown nonlinear system with two inputs and two outputs. Therefore, let the inputs of FNN be the control input, τ_1 and τ_2, and the output of FNN be $y_1^{(4)}$ and $y_2^{(4)}$. Thus, the output of i th FNN can be represented as

$$y_i^{(4)} = FNN_i(\tau_1, \tau_2), \quad i = 1, 2 . \tag{7}$$

To derive the on-line learning algorithm, the supervised gradient descent method is adopted. In the system identification phase, let the energy function for i th link be defined as

$$E_i = \frac{1}{2}\left(x_i - y_i^{(4)}\right)^2 . \tag{8}$$

Minimizing the given energy function (8), three parameters including the mean m_{jk}, the standard deviation δ_{jk} of Gaussian functions, and the link weight w_k, are to be tuned. The on-line learning algorithm can be described as follows.

The updating law of m_{jk} for i th FNN is

$$\Delta m_{jk}{}^i = -\eta_m{}^i \frac{\partial E_i}{\partial m_{jk}{}^i} , \tag{9}$$

where $\eta_m{}^i$ is the learning rate of the mean of the Gaussian function for i th FNN.

The updating law of δ_{jk} for i th FNN is

$$\Delta \delta_{jk}{}^i = -\eta_\delta{}^i \frac{\partial E_i}{\partial \delta_{jk}{}^i} , \tag{10}$$

where $\eta_\delta{}^i$ is the learning rate of the standard deviation of the Gaussian function for i th FNN.

The updating law of w_k for i th FNN is

$$\Delta w_k{}^i = -\eta_w{}^i \frac{\partial E_i}{\partial w_k{}^i} , \tag{11}$$

where $\eta_w{}^i$ is the learning rate of the weight for i th FNN.

The mean and standard deviation of the Gaussian function and the weight of i th FNN are updated as follows:

$$m_{jk}{}^i(n+1) = m_{jk}{}^i(n) + \Delta m_{jk}{}^i , \tag{12}$$

$$\delta_{jk}{}^i(n+1) = \delta_{jk}{}^i(n) + \Delta \delta_{jk}{}^i , \tag{13}$$

$$w_k{}^i(n+1) = w_k{}^i(n) + \Delta w_k{}^i . \tag{14}$$

Then, the two FNN can act an identified nonlinear system as long as the FNN is fulfilled through (9)-(14). Without the system model, an advantage of the identified FNN is that it can be behalf of a real system under system uncertainties and external disturbances. Thus, the adaptive laws of the FLC can be obtained easily through the identified FNN as the following descriptions.

2.5 Adaptive laws design

In the tracking phase, let the energy function for i th output be defined as

$$V_i = \frac{1}{2}(x_{id} - x_i)^2 . \tag{15}$$

According to the supervised gradient descent method, the adaptive law of $\hat{r}_i$ is

$$\Delta\hat{r}_i = -\eta_r{}^i \frac{\partial V_i}{\partial \hat{r}_i}, \tag{16}$$

where $\eta_r{}^i$ is the learning rate of the i th input gain. As long as the on-line tuning FNN with the updating laws (9)-(14) is fulfilled, the system identification will be completed, i.e., $y_i^{(4)} \to x_i$. One may rewrite (16) with the control law (2), i.e.,

$$\Delta\hat{r}_i = -\eta_r{}^i \frac{\partial V_i}{\partial x_i}\frac{\partial y_i^{(4)}}{\partial \tau_i}\frac{\partial \tau_i}{\partial \hat{r}_i}, \tag{17}$$

The control gain of the input τ_i are updated as follows,

$$\hat{r}_i(n+1) = \hat{r}_i(n) + \Delta\hat{r}_i . \tag{18}$$

Hence, the convergence of $\hat{r}_i$ is proven by (17). The control gain $\hat{r}_i$ will reach optimal values until the tracking error approximates to zero.

In summary, the adaptive fuzzy controller based on fuzzy neural network is proposed for nonlinear systems. In Fig. 3, the proposed on-line tuning FNN with the updating laws (9)-(11) provides the parameters $m_{jk}{}^i$, $\delta_{jk}{}^i$, and $w_k{}^i$ for the adaptive regulator. Then, the adaptive regulator adjusts the control gain $\hat{r}_i$ automatically using (17). Simultaneously, the universal range of output is also tuned automatically. In this approach, the adaptive universal range replaces the fixed universal range in conventional fuzzy control. Therefore, the proposed adaptive FLC has the optimal universal range according to the tracking error for control systems with uncertainties and external disturbances.

3. EXPERIMENTAL RESULTS

Figure 5 shows the photograph of a two-link robotic arm. Motors 1 and 2 drive the Links 1 and 2 to swing. Sensors 1 and 2 return the angles of links. Figure 6 shows the developed two-link robotic arm control system. A motion control card and the VisSim software are adopted on a personal computer. In the driver, the stopped value for the analog input is 2.5V. The lowering voltage will increase the speed in reverse; raising the voltage will increase the speed forward. To reduce the cost, a cheap varying resistance is used to be a feedback sensor. Applying the dividing voltage theorem, the angle of robotic arm can be calculated accordingly. The most advantages are that the method is cheap and simple. Oppositely, the shortcoming of a varying resistance is less accuracy than an encoder. In this practical system, we only consider one input e_i as input variable in the FLC. Although it is different from the conventional FLC with e_i and $\dot{e}_i$ as input variables, one input e_i can handle the whole control system enough.

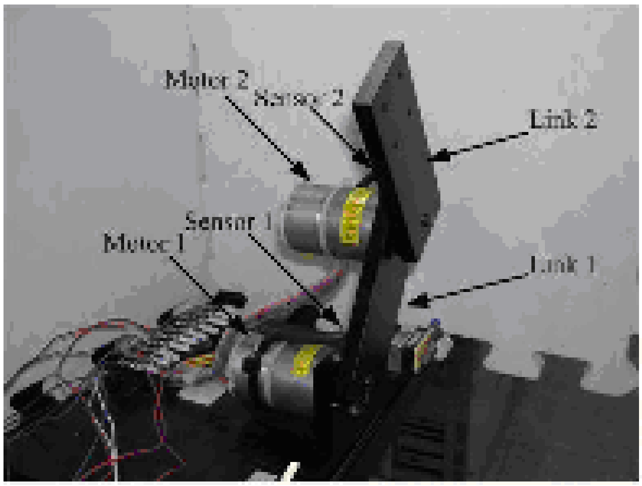

Fig. 5. Photograph of a two-link robotic arm.

In order to demonstrate the control performance, let the desired trajectory be $60\sin(2t)$. The fuzzy rules are set as following:

Rule 1: If e_i is NB, then $\hat{\tau}_i$ is NB

Rule 2: If e_i is NM, then $\hat{\tau}_i$ is NM

Rule 3: If e_i is ZE, then $\hat{\tau}_i$ is NB

Rule 4: If e_i is PM, then $\hat{\tau}_i$ is PM

Rule 5: If e_i is PB, then $\hat{\tau}_i$ is PB

The normalized membership functions in Fig. 2 are used. The universal range of input variable e_i is [-30, 30]. The initial universal range of output variable $\hat{\tau}_i$ is set as [-1, 1]. The initial $\hat{r}_1$ and $\hat{r}_2$ are 2 and 1.5, respectively. The sampling time is 0.01 seconds. The learning rates $\eta_m{}^i$, $\eta_\delta{}^i$, $\eta_w{}^i$, and $\eta_r{}^i$ are set to be 0.8. In the FNN structure, five nodes are used in Layer 3 (rule layer), i.e., $k = 5$. The initial parameters are $m_{11}{}^i = m_{21}{}^i = 0$, $m_{12}{}^i = m_{22}{}^i = 1.25$, $m_{13}{}^i = m_{23}{}^i = 2.5$,

$m_{14}{}^{i} = m_{24}{}^{i} = 4.75$, $m_{15}{}^{i} = m_{25}{}^{i} = 5$, $\delta_{11}{}^{i} = \delta_{12}{}^{i} = \cdots = \delta_{15}{}^{i} = \delta_{21}{}^{i} = \delta_{22}{}^{i} = \cdots = \delta_{25}{}^{i} = 0.75$, and $w_1{}^{i} = w_2{}^{i} = \cdots = w_5{}^{i} = 1$, respectively.

With the FLC (1), the updating laws (9-14), and the adaptive laws (17), a most advantage is that the proposed FNN-AFLC does not need the dynamic equation of the two-link robotic arm. Figures 7 and 8 show the tracking trajectories, respectively. Since the initial gains $\hat{r}_1$ and $\hat{r}_2$ are arbitrarily chosen, the tracking response is not good in the beginning. A few seconds later, the tracking response is more satisfactory as long as the adaptive gains reach their optimal values. Note that the maximum motor voltage is limited between 0V and 5V as shown in Figs. 9 and 10. Through the adaptive gains $\hat{r}_1$ and $\hat{r}_2$ can be tuned automatically to their optimal values until the tracking errors approach to zero theoretically, however, there are still small tracking errors which are caused by the defects of the too simple mechanism design. The gains $\hat{r}_1$ and $\hat{r}_2$ are blocked from 1 to 2.5 in order to protect the hardware. Finally, Figure 11 shows the changing curves of the adaptive gains. The experimental results reveal that the proposed FNN-AFLC is indeed effective in practice.

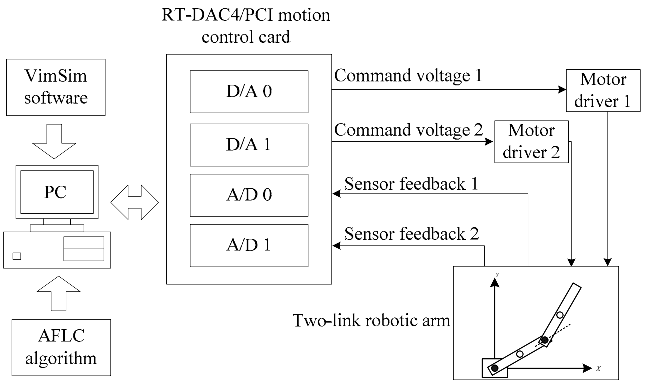

Fig. 6. A two-link robotic arm control system.

4. CONCLUSIONS

In this study, an adaptive and intelligent control method for unknown nonlinear systems is presented. The strategy involves an on-line tuning FNN and an adaptive FLC. According to the gradient descent method, the FNN can on-line identify the unknown nonlinear system and provide the optimal adaptive parameters for the adaptive FLC. The adaptive FLC can auto-tune the universal range according to the tracking error. The main contributions of this adaptive fuzzy controller include: 1) speeding up the tracking response, 2) improving the steady system performance, 3) increasing the system robustness, and 4) releasing the requirement of system model. Finally, the proposed control method was applied to the tracking control of a practical two-link robotic arm. In the experiment, the mechanism design was too simple, so there were still some small transient errors. However, the theoretical result was verified. The proposed method can be effectively applied to those control systems without system models.

5. ACKNOWLEDGEMENT

The authors would like to thank the National Science Council of Taiwan, for financially supporting this work under Grants NSC95-2221-E-231-013 and NSC95-2622-E-155-008-CC3.

REFERENCES

Betin, F., D. Pinchon, and G.A. Capolino (2000). Fuzzy logic applied to speed control of a stepping motor drive. *IEEE Transactions on Industrial Electronics*, **47**, pp. 610-622.

Chen, Y.C. and C.C. Teng (1995). A model reference control structure using a fuzzy neural network. *Fuzzy Sets Systems*, **73**, pp. 291-312.

Er, M.J. and S.H. Chin (2000). Hybrid adaptive fuzzy controllers of robot manipulators with bounds estimation. *IEEE Transactions on Industrial Electronics,* **47**, pp. 1151-1160.

Guan, P., X.J. Liu, and J.Z. Liu (2005). Adaptive fuzzy sliding mode control for flexible satellite. *Engineering Application of Artificial Intelligent*, **18**, pp. 451-459.

Guo, Y. and P.Y. Woo (2003). An adaptive fuzzy sliding mode controller for robotic manipulators. *IEEE Transactions on Systems*, *Man*, *Cybernetics, part A*, **33**, pp. 149-159.

Ham, C., Z. Qu, and R. Johnson (2000). Robust fuzzy control for robot manipulator. *IEE Proceeds- Control Theory Applications*, **147**, pp. 212-216.

Horikawa, S., T. Furuhashi, and Y. Uchikawa (1992). On fuzzy modeling using fuzzy neural networks with the backpropagation algorithm. *IEEE Transactions on Neural Network*, **3,** pp. 801-806.

Khaber, F., A. Hamzaoui, K. Zehar, and N. Essounbouli (2006). New stable adaptive fuzzy approach to control uncertain nonlinear SISO systems. *International Journal of Systems Science*, **37**, pp. 437-445.

Lee, H.D. and S.K. Sul (1998). Fuzzy-logic based torque control strategy for parallel-type hybrid electric vehicle. *IEEE Transactions on Industrial Electronics*, **45**, pp. 625-632.

Lower, M. (2005) Fuzzy flight control system for helicopter intelligence in hover," in *Proceeds of the 5th International Conference Intelligent Systems Design and Applications*, Poland, pp. 370-374.

Sharkawy, A.B. (2005). Fuzzy and adaptive fuzzy control for the automobiles' active suspension system. *Vehicle Systems Dynamics*, **43**, pp. 795-806.

Wang, L. (1997). *A course in fuzzy systems and control*. Prentice Hall, New Jersey.

Yoo, B.K. and W.C. Ham (2000). Adaptive control of robot manipulator using fuzzy compensator. *IEEE Transactions on Fuzzy Systems*, **8**, pp. 186-199.

Zadeh, L.A. (1965). Fuzzy sets. *Information Control*, **8**, pp. 338-353.

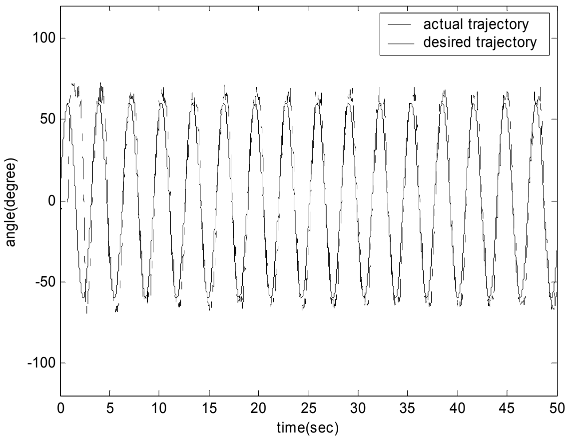

Fig. 7. Tracking control of first link.

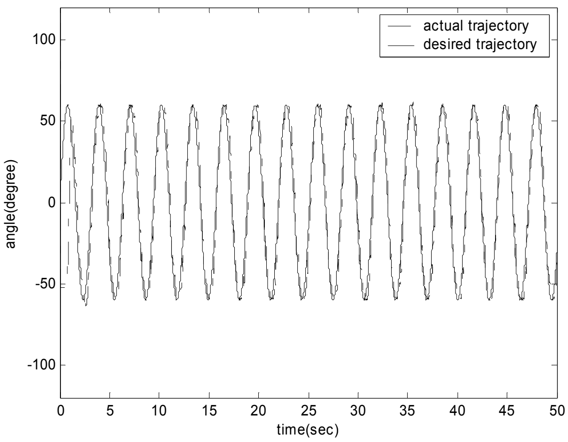

Fig. 8. Tracking control of second link.

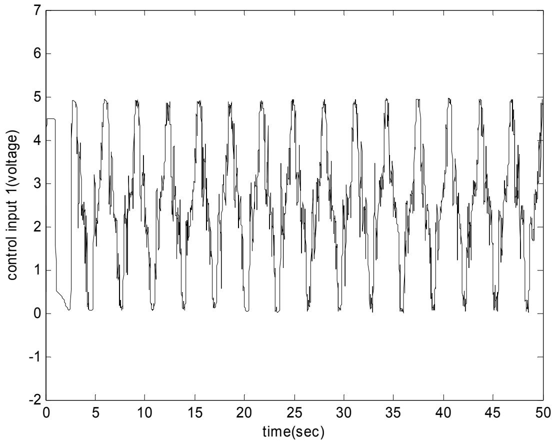

Fig. 9. Control input of first link.

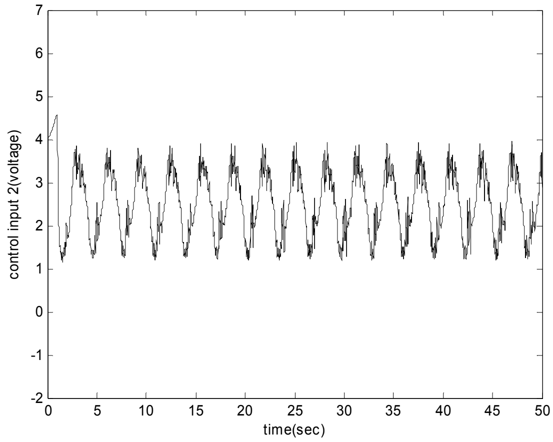

Fig.10. Control input of second link.

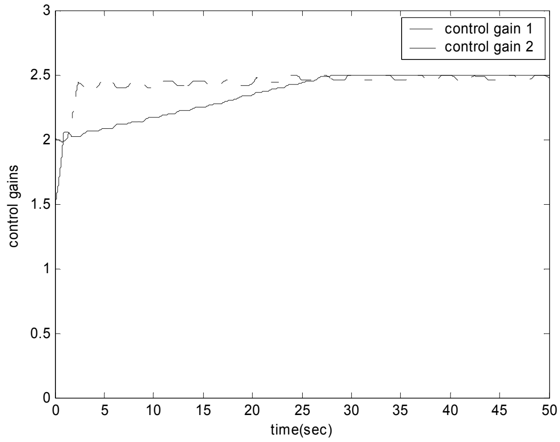

Fig. 11. Control gains.