Proceedings of the 12th IEEE International Symposium on Intelligent Control 16-18 July 1997, Istanbul, Turkey

Neuro-Based Adaptive Torque Control of a Flexible Arm

Toshio TSUJI, Bing Hong XU and Makoto KANEKO
Faculty of Engineering, Hiroshima University
Higashi-Hiroshima, 739 JAPAN

Abstract

This paper presents a Neuro-Based Adaptive Control (NBAC) for torque control of a flexible arm with structural uncertainties. In the NBAC, a neural network (NN) is connected in parallel with a linearized plant model, so that the NN is expected to identify the uncertainties included in the plant. The NN works as an adaptive controller simultaneously that can compensate for the uncertainties. The NBAC is applied to the torque control of a flexible arm that includes linear and nonlinear uncertainties. Experimental results illustrate effectiveness and applicability of the NBAC.

1 Introduction

In recent years, applications of the neural network to adaptive control have been intensively conducted. The methods which have been proposed so far may be classified into two approaches according to the number of the neural networks to be used.

The first approach utilizes a single neural network in order to obtain a kind of the inverse model of the controlled system, such as Yabuta and Yamada [1], Kraft and Campagne [2], Carelli et al. [3] and Akhyar and Omatsu [4]. In this approach, even if the adaptive controller of the controlled system is obtained by learning of the neural network, the uncertainty included in the controlled system cannot be obtained. When the forward model of the controlled system is necessary, the controlled system must be identified again by using other identification techniques.

The second approach of the neural adaptive control utilizes multiple neural networks, such as Narendra and Parthasarathy [5], Ku and Lee [6] and Iiguni et al. [7]. In this approach, one neural network is dedicated to the forward model for identifying the uncertainties of the controlled system, and the other neural networks may compensate for the effect of the uncertainties based on the trained forward model. However, multiple neural networks must be trained and stability of the control system is quite difficult to be assured.

In this paper, we propose a new neural adaptive control scheme called *Neuro-Based Adaptive Control (NBAC)* that can realize adaptive control and identification for a class of controlled systems including uncertainties using only a single neural network. The neural network can identify the uncertainties included in the controlled system and adaptively modify the control input computed by a pre-designed conventional feedback controller.

Then the NBAC is applied to a torque control problem of a single-joint flexible arm. While the flexible arm contacts with a fixed object, we would like to control the joint torque of the flexible arm. In this case, the dynamic characteristics of the flexible arm under consideration nonlinearly depend on the material and shape of the arm, contact force, contact friction and so on. It is very difficult to obtain the exact dynamic model of the flexible arm beforehand, and a precise torque control of the flexible arm cannot be achieved by a conventional adaptive control technique. In this paper, the control performance and identification ability of the NBAC are shown with the comparison of the experimental results using a model reference adaptive control and other neural adaptive control methods.

2 Neuro-Based Adaptive Control

2.1 Plant model

Consider a nonlinear discrete-time plant described as the following single-input-single-output form

$$y(k) = H_L(z^{-1})u(k) + f(u(k)), \tag{1}$$

where $u(k)$ and $y(k)$ are the input and the output of the plant, respectively; $f(u(k))$ represents the nonlinear part; and $H_L(z^{-1})u(k)$ represents the linear part of the plant. The linearized model $H_L(z^{-1})$ is further supposed to be divided into a nominal model $H_{La}(z^{-1})$ and uncertainties $\Delta_{La}(z^{-1})$:

$$H_L(z^{-1}) = H_{La}(z^{-1}) + \Delta_{La}(z^{-1}), \tag{2}$$

where z^{-1} denotes for the delay operator.

For the nonlinear part $f(u(k))$, it is assumed that its linear approximation is given by

$$f(u(k)) \approx H^*(z^{-1})u(k) \tag{3}$$

0-7803-4116-3 / 97 / $10.00

From (2) and (3), the plant dynamics (1) can be rewritten as

$$\begin{aligned} y(k) &\approx H_L(z^{-1})u(k) + H^*(z^{-1})u(k) \\ &= H(z^{-1})u(k), && (4) \\ H(z^{-1}) &= H_n(z^{-1})[1+\Delta_H(z^{-1})], && (5) \\ H_n(z^{-1}) &= H_{La}(z^{-1}) + H^*(z^{-1}), && (6) \\ \Delta_H(z^{-1}) &= \frac{\Delta_{La}(z^{-1})}{H_n(z^{-1})}, && (7) \end{aligned}$$

where $H_n(z^{-1})$, $\Delta_H(z^{-1})$ are the known nominal model and the uncertainty, respectively. We assume that the structure of $H_n(z^{-1})$ and $\Delta_H(z^{-1})$ can be described by the following expressions:

$$\begin{aligned} H_n(z^{-1}) &= \frac{B_n(z^{-1})}{A_n(z^{-1})}, && (8) \\ A_n(z^{-1}) &= 1+\sum_{j=1}^{n} a_j z^{-j}, && (9) \\ B_n(z^{-1}) &= \sum_{i=0}^{m} b_i z^{-i} \quad (n \geq m), && (10) \\ \Delta_H(z^{-1}) &= \frac{\Delta_B(z^{-1})}{\Delta_A(z^{-1})}, && (11) \\ \Delta_A(z^{-1}) &= 1+\sum_{j=1}^{h} \alpha_j z^{-j}, && (12) \\ \Delta_B(z^{-1}) &= \sum_{i=0}^{l} \beta_i z^{-i} \quad (h \geq l), && (13) \end{aligned}$$

where $H_n(z^{-1})$ is controllable; $\Delta_A(z^{-1})$, $\Delta_B(z^{-1})$ include unknown coefficients α_j, β_i; and $l(\leq n)$, $h(\leq m)$ are unknown orders of the polynomials $\Delta_A(z^{-1})$, $\Delta_B(z^{-1})$. Note that appearance of the uncertainties in the plant dynamics is changed from the additive form (1) to the multiplicative one (5).

2.2 Adaptive Control Scheme

The block diagram of the NBAC of the discrete-time plant with uncertainties (1) is shown in Fig. 1, where $r(k)$ is the reference signal and $e(k) = r(k) - y(k)$ is the error between the reference signal and the output. The output $\hat{y}(k)$ of the identification model is the sum of the nominal model output $y_n(k)$ and the identified output $y_{ID}(k)$ that is the NN's output passed through the model $H_n(z^{-1})$ as shown in Fig. 1. The NN is trained using the identified error $\epsilon(k)$. Thus, the input $u(k)$ is defined as

$$u(k) = u_n(k) - y_{NN}(k). \quad (14)$$

Now, the behavior of the NBAC system is explained as follows. First, consider a special case when there are no uncertainties in the plant (1), that is, $\Delta_u(k) = 0$. The controller $G_n(z^{-1})$ for the model $H_n(z^{-1})$ is designed to realize a desirable response.

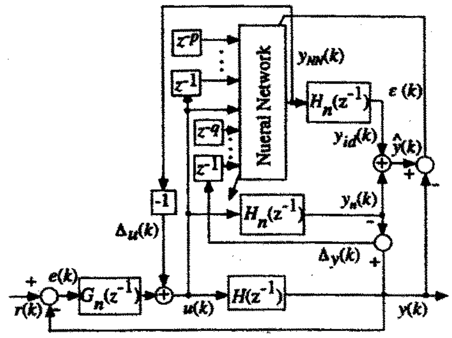


Figure 1: Block diagram of the NBAC

The closed loop transfer function $F_n(z^{-1})$ is described as

$$F_n(z^{-1}) = \frac{y(k)}{r(k)} = \frac{G_n(z^{-1})H_n(z^{-1})}{1+G_n(z^{-1})H_n(z^{-1})}. \quad (15)$$

Next, consider the general case of $\Delta_u(k) \neq 0$. The controller $G(z^{-1})$ is defined as

$$G(z^{-1}) = G_n(z^{-1})[1+\Delta_G(z^{-1})], \quad (16)$$

where $\Delta_G(z^{-1})$ represents the modified value of the controller $G(z^{-1})$. Thus, the closed loop transfer function $F(z^{-1})$ that consists of $H(z^{-1})$ and $G(z^{-1})$ is given as

$$F(z^{-1}) = \frac{G(z^{-1})H(z^{-1})}{1+G(z^{-1})H(z^{-1})}. \quad (17)$$

If (15) and (17) are equivalent, the response of $H(z^{-1})$ by $G(z^{-1})$ should accord with the desired response. In this case we have the following relation from (7) and (15):

$$\Delta_G(z^{-1}) = -\frac{\Delta_H(z^{-1})}{1+\Delta_H(z^{-1})}. \quad (18)$$

On the other hand, by (16) the input $u(k) = G(z^{-1})e(k)$ can be represented as

$$\begin{aligned} u(k) &= G_n(z^{-1})e(k) + \Delta_u(k), && (19) \\ \Delta_u(k) &= \Delta_G(z^{-1})G_n(z^{-1})e(k). && (20) \end{aligned}$$

Also, from (5), the output $y(k)$ becomes

$$\begin{aligned} y(k) &= H_n(z^{-1})u(k) + H_n(z^{-1})\Delta_y(k), && (21) \\ \Delta_y(k) &= \Delta_H(z^{-1})u(k), && (22) \end{aligned}$$

where $\Delta_u(k)$, $\Delta_y(k)$ are the modified value of the input and the output of the uncertainties, respectively.

Substituting (19), (20), (22) into (18), we have

$$\Delta_u(k) = -\Delta_y(k). \quad (23)$$

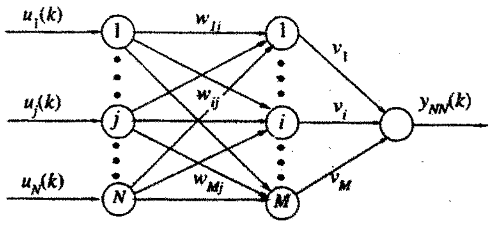


Figure 2: Neural network used in the NBAC

In the training process of the NN, the identified error $\epsilon(k) = \widehat{y}(k) - y(k)$ can be described by

$$\epsilon(k) = H_n(z^{-1})\{y_{NN}(k) - \Delta_y(k)\}. \quad (24)$$

If the NN is well trained, we can expect $y_{NN}(k) \approx \Delta_y(k)$. Consequently, we can see that the response of the control system shown in Fig. 1 agrees with the desired response from (14).

2.3 Neural Network

The multi-layer NN is used in the NBAC as shown in Fig. 2. The number of units of the input layer and the hidden layer are N and M, respectively. The number of units of the output layer is one. In Fig. 2, $w_{ij}(k)$ represents the weight that connects the unit j of the input layer to the unit i of the hidden layer; $v_i(k)$ represents the weight that connects the unit i of the hidden layer to the output layer's unit; $\boldsymbol{W}(k) \in R^{M\times N}$, $\boldsymbol{V}(k) \in R^{M\times 1}$ are the weight matrices of the hidden layer and the weight vector of the output layer, respectively. The NN's input vector $\boldsymbol{U}_{IN}(k) = [u_1(k), u_2(k), \cdots, u_N(k)]^T \in R^{N\times 1}$ is defined as

$$\begin{aligned} \boldsymbol{U}_{IN}(k) &= [u(k), u(k-1), \cdots, u(k-l), \\ &\quad \Delta_y(k-1), \cdots, \Delta_y(k-h)]^T, \quad (25) \end{aligned}$$

where $N = l + h + 1$.

Also, the unit j's output of the input layer is defined as $I_j = u_j(k)$ $(j = 1, \cdots, N)$, the unit i's output of the hidden layer as

$$H_i = \sigma(s_i), \quad (26)$$

$$s_i = \sum_{j=1}^{N} w_{ij} I_j, \quad (27)$$

and the sigmoid function as

$$\sigma(x) \equiv \frac{1}{\gamma} tanh(\gamma x), \quad (28)$$

where γ is the positive parameter related with the shape of the sigmoid function. Moreover, the output of the output layer is defined as

$$O_k = \sigma(\kappa), \quad (29)$$

$$\kappa = \sum_{i=1}^{M} v_i H_i. \quad (30)$$

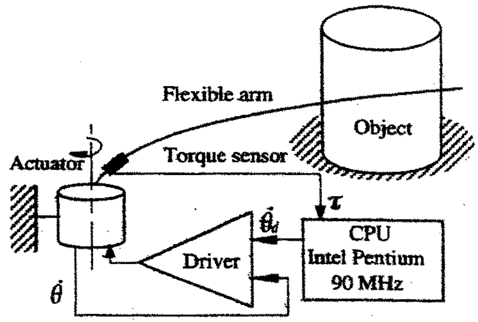


Figure 3: Flexible arm in contact with an object

The energy function for the learning of NN is

$$J(k) = \frac{1}{2}\epsilon^2(k), \quad (31)$$

which is minimized by changing the weights w_{ij} and v_i. According to the error back propagation algorithm [8], the weight updating rules at one sampling time can be described as

$$\boldsymbol{V}(k+1) = \boldsymbol{V}(k) - \eta\epsilon(k)H_n(z^{-1})\frac{\partial y_{NN}(k)}{\partial \boldsymbol{V}(k)}, \quad (32)$$

$$\boldsymbol{W}(k+1) = \boldsymbol{W}(k) - \eta\epsilon(k)H_n(z^{-1})\frac{\partial y_{NN}(k)}{\partial \boldsymbol{W}(k)}, \quad (33)$$

where $\eta > 0$ is the learning rate. The reference [9] analyzes the stability of the proposed system.

3 Torque Control of a Flexible Arm

In this section, we apply the proposed scheme to torque control of a flexible arm as shown in Fig.3. While the flexible arm contacts with a fixed object, we would like to control the joint torque of the flexible arm in accordance with a reference signal.

The rotational stiffness of the joint is largely changed depending on the position of the contact point [10]. When the distance from the joint to the contact point is small, the rotational stiffness should be large. When the contact point goes away from the joint, the joint becomes less stiffer. Thus, the dynamic characteristics of the flexible arm under consideration nonlinearly depend on the contact point as well as the material and shape of the flexible arm, contact force, and contact friction.

3.1 Experimental Device

An experimental device for the torque control of the flexible arm is shown in Fig. 3. The arm is steel, 0.32 m in length and 0.8 mm in diameter. The torque sensor of a semiconductor gauge is glued on an aluminum sheet. When the arm contacts with a fixed object, the torque $\tau(k)$ at the joint of the arm can be measured by the torque sensor. The actuator is

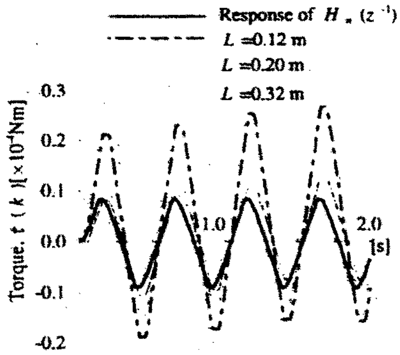


Figure 4: Measured and predicted torque of the flexible arm

velocity-controlled with the desired angular velocity $\dot{\theta}_d(k)$ of the joint. It should be noted that the driving torque of the actuator cannot be controlled directly.

For this experimental device, we consider a plant described by

$$\tau(k) = H_n(z^{-1})[1 + \Delta_H(z^{-1})]u(k), \qquad (34)$$

where $u(k)$ and $\tau(k)$ are the input to the actuator and the joint torque of the flexible arm, respectively; and $H_n(z^{-1})$ represents the linear nominal model that is estimated from measured data by using a conventional identification technique as follows.

The desired angular velocity $\dot{\theta}_d(k)$ is considered as the input to the flexible arm, so that the transfer function from $\dot{\theta}_d(k)$ to the torque $\tau(k)$ can be approximately described by

$$H_n(s) = \frac{K_s K_b}{s(t_s s + 1)}, \qquad (35)$$

where K_s is the gain and t_s is the time constant of the velocity-controlled system, and K_b is the elastic constant of the arm. The discrete form of (35) is given as

$$H_n(z^{-1}) = \frac{b_1 z^{-1} + b_2 z^{-2}}{1 + a_1 z^{-1} + a_1 z^{-2}}. \qquad (36)$$

In order to identify the parameters of (36), the contact point L= 0.20 m is chosen and the arm is fixed to the environment. The rectangular input signal with its amplitude of 2.0×10^{-4} rad/s and a period of 0.5 s is used as $\dot{\theta}_d$ after passing through the first-order low-pass filter with a cut-off frequency 5 Hz. The joint torque is measured with the sampling frequency 100 Hz.

The identified values of the model parameters are determined as $\hat{a}_1$=-1.20296, $\hat{a}_2$=0.20121, $\hat{b}_1$=0.03063, $\hat{b}_2$=0.07341. The response of the nominal model with the identified parameters is shown in Fig. 4 as the thick line. From Fig. 4 we can see that the error between the output of $H_n(z^{-1})$ and the measured torque of the flexible arm with $L = 0.2$ m is increasing with time.

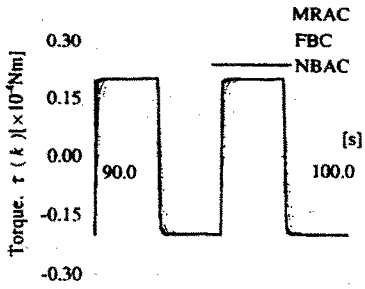


Figure 5: Experimental results of the torque control of the flexible arm with L=0.20 m

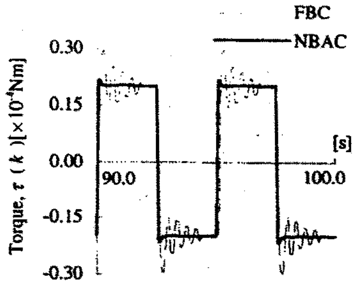
Using the same experimental device, the fixed position L of the arm is changed. The measured results are also shown in Fig. 4. Here, the alternate long and short dashed line represents the result with L= 0.12 m and the dashed line represents the result with L= 0.32 m. When the contact position L is varied, the joint torque becomes significantly different from the output of the nominal model with L= 0.20 m. In the next subsection, the proposed scheme using the identified parameters for L= 0.20 m is applied to the torque control of the flexible arm with different contact positions.

3.2 Control Performance

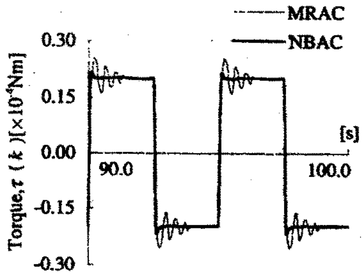
In the neural network used in the experiment, the initial value of the weight is set as an uniform random number in $[-1.0 \times 10^{-3},\ 1.0 \times 10^{-3}]$. The learning rate is η=0.05 and the parameter γ of the sigmoid function is γ=1. In order to cover the maximum order (p=2, q=2) of the uncertainties $\Delta_H(z^{-1})$, the neural network consists of five units in the input layer, and ten units in the hidden layer, and one unit in the output layer. Also, the reference signal $r(k)$ is of a rectangular form with its amplitude of 2.0×10^{-4} Nm and a period of 5 s. The feedback controller $G_n(z^{-1})$ is $G_n(z^{-1})$=1, and the sampling time and the control duration are 0.01 s and 100 s, respectively. The proposed scheme is applied to 6 different contact points that are L =0.12, 0.16, 0.20, 0.24, 0.28, 0.32 m.

Figures 5, 6, 7 show the experimental results corresponding to the case of L=0.20 m, L=0.12 m, L=0.32 m, respectively. In all cases, the dashed lines represent the results corresponding to the use of the feedback controller (FBC) $G_n(z^{-1})$, the thick lines the results by the proposed scheme (NBAC), the thin lines the results obtained with the model reference adaptive control (MRAC) [11]. In the MRAC, as the reference model and the controller, the nominal model $H_n(z^{-1})$ of (36) with the identified parameters (L=0.2 m) and $G_n(z^{-1}) = 1$ are used.

In Fig. 5, due to the fact that the same value of



(a) NBAC and the feedback control by using $G_n(z^{-1}) = 1$



(b) NBAC and the model reference adaptive control

Figure 6: Experimental results of the torque control of the flexible arm with L=0.12 m

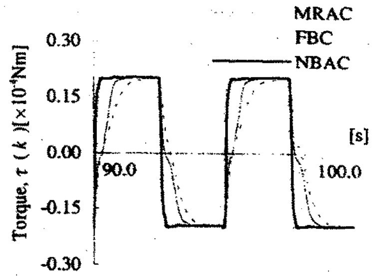


Figure 7: Experimental results of the torque control of the flexible arm with L=0.32 m

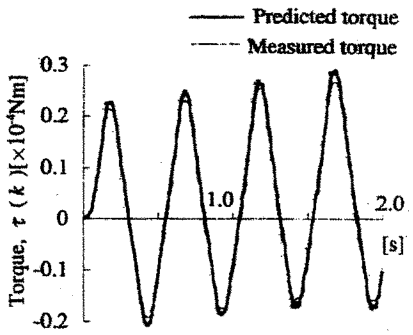


Figure 8: Identification ability of the NBAC

L=0.20 m is used for identifying the nominal model, the experimental results obtained under three control methods are not obviously different.

However, when L is varied as shown in Fig. 6 and Fig. 7, the feedback control by using $G_n(z^{-1})$ produces significant overshoot or undershoot. The model reference adaptive control works well for linear parameter perturbation, so that it can improve the control performance slightly. On the other hand, the proposed scheme always produces stable responses. It should be noted that the identified parameters of the nominal model for L=0.20 m are used for all cases.

3.3 Identification Ability

The other feature of the proposed scheme is that it can construct the identification model for the controlled plant (see $\widehat{y}(k)$ shown in Fig. 1). So we investigate the identification ability of the proposed method for the contact point L=0.12 m after learning of 100 s which is corresponding to Fig. 7.

Figure 8 shows the identification model's output $\widehat{y}(k)$ for the same input signal as shown in Fig. 4. It should be noted that the identified parameters for L=0.20 m are used as the nominal model $H_n(z^{-1})$. We can see that the adaptive control and identification for the controlled plant can be achieved using only a single neural network simultaneously.

3.4 Comparison

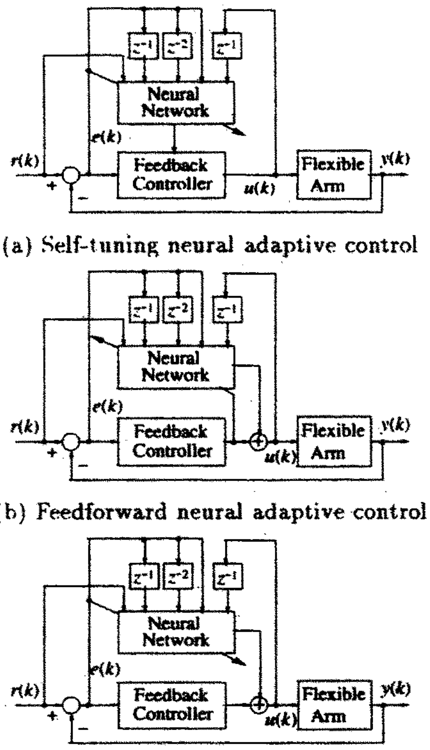
In this subsection, the proposed method is compared with other neural adaptive control methods, which are the self-tuning neural adaptive control (abbreviated as STNC) [4], the feedforward neural adaptive control (FNAC) [3], and the parallel neural adaptive control (PNAC) [2]. Figure 9 shows three block diagrams of STNC, FNAC and PNAC, respectively. It should be noted that the direct neural adaptive control [1] did not result any stable learning in our experiments.

Experimental conditions are the same as the ones described in the subsection 3.2 except for the control duration 60 s. The mean square error E_n during the one period of the reference signal, that is

$$E_n = \frac{1}{5}\sum_{k=1}^{500} e^2[500(n-1)+k] \quad (n = 1, 2, \cdots, 12), \tag{37}$$

is computed for each control method. The sampling frequency is 100 Hz.

Figure 10 shows comparisons of the learning history. The learning speed of the proposed method is faster than those of other control methods. STNC, PNAC and FNAC need to learn the inverse model, while the proposed scheme requires to learn only the uncertainty included in the forward model of the controlled plant. Thus, the learning load of the proposed scheme is much less than the ones of other control methods.



(a) Self-tuning neural adaptive control

(b) Feedforward neural adaptive control

(c) Parallel neural adaptive control

Figure 9: Block diagrams of the adaptive control systems using neural networks

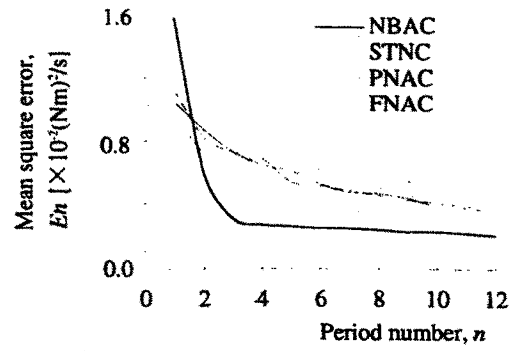


Figure 10: Comparison of the learning history

4 Conclusion

In this paper, a Neuro-Based Adaptive Control of a discrete-time plant with uncertainties has been proposed. Then, the NBAC was applied to the torque control of a flexible arm in contact with the environment. Even though the parameters of the flexible arm were largely varied, the joint torque was controlled adaptively. Experimental results illustrated effectiveness and applicability of the NBAC. In future, we plan to extend the NBAC method for a nonlinear plant.

The authors would like to thank Mr. Naohiro Ueno of Hiroshima University for providing us his experimental devices.

References

[1] T. Yabuta and T. Yamada, "Neural Network Controller Characteristics with Regard to Adaptive Control," *IEEE Trans. On Systems, Man, and Cybernetics*, vol.22, no.1, pp.170–176, 1992

[2] L. G. Kraft, III and D. P. Campagna, "A Summary Comparison of CMAC Neural Network and Traditional Adaptive Control Systems," In W.Thomas Miller, III, Richard S. Sutton, and Paul J. Werbos, *Neural Networks for Control*, MIT Press, Cambridge, Massachusetts, 1990

[3] R. Carelli, E. F. Camacho and D. Patiño, "A Neural Network Based Feedforward Adaptive Controller for Robots," *IEEE Trans. On Systems, Man, and Cybernetics*, vol.25, no.9, pp.1281–1288, 1995

[4] S. Akhyar and S. Omatsu, "Self-tuning PID Control by Neural Networks," *Proc. of International Joint Conference on Neural Networks*, pp.2749–2752, 1993

[5] K. S. Narendra and K. Parthasarathy, "Identification and Control of Dynamical Systems Using Neural Networks," *IEEE Trans. on Neural Networks*, vol.1, no.1, pp.4–27, 1990

[6] C. Ku and K. Y. Lee, "Diagonal Recurrent Neural Networks for Dynamic Systems Control," *IEEE Trans. on Neural Networks*, vol.6, no.1, pp.144–156, 1995

[7] Y. Iiguni, H. Sakai and H. Tokumaru, "A Nonlinear Regulator Design in the Presence of System Uncertainties Using Multilayered Neural Network," *IEEE Trans. On Neural Networks*, vol.2, no.4, pp.410–417, July 1991

[8] D. E. Rumelhart, G. E. Hinton and R. J. Williams, "Learning Representations by Error Propagation," in D. E. Rumelhart, J. L. McClelland and PDP Research Group: *Parallel Distributed Processing*, vol.1, pp.318–362, MIT Press, 1986

[9] T. Tsuji , B. H. Xu and M. Kaneko, "Identification and Control of Dynamical System by One Neural Network," *Proceedings of the IEEE International Conference on Industrial Technology*, pp.701–705, 1996

[10] M. Kaneko, "Active Antenna," *Proc. IEEE int. Conference on Robotics and Automation*, pp.2665–2671, 1994

[11] K. J. Åstrom and W. Wittenmark "Adaptive Control," Addison-Wesley, 1989