A ROBUST ITERATIVE LEARNING CONTROL WITH NEURAL NETWORKS FOR ROBOT

Cheng Shao[1], Juan Nie[1], Furong Gao[2]

[1]*Research Center of Information and Control*
Dalian University of Technology, Dalian 116024, Liaoning, P.R. China
Email: cshao@dlut.edu.cn
[2]*Department of Chemical Engineering, Hong Kong University of Science and Technology,*
Clear Water Bay, Kowloon, Hong Kong

Abstract: Using identification of neural networks, a new robust iterative learning control algorithm is proposed in the paper. Combined with feedback control in real time, the neural network is employed to identify the nonlinear system online and to produce the feed-forward actions of iterative learning control algorithm to realize continuous trajectory tracking task for robot. Simulation results demonstrate that the algorithm can not only overcome uncertainties and external disturbances, but also meet the trajectory command with few iterative learning and network training, and thus possess better robustness and control performance.

Keywords: iterative learning control, neural network, system identification, disturbances rejection.

1. INTRODUCTION

Robot is a kind of high nonlinear, closely coupled and time-varying dynamic system, so that its exact dynamic model is difficultly established. In order to satisfy the requirement of high-precision motion control of robot, some of the literatures have proposed many new control methods, such as computed torque method, adaptive control method, varying-structured control method and iterative learning control method. Among these methods, iterative learning control has been aroused general interest. This scheme can utilize a prior knowledge regarding the controlled system, combining its output and desired signals, so as to make the controlled system yield the desired movement. Especially for high nonlinear and close coupled dynamical systems, meanwhile with high-precision requirement of position, like industrial robot and digital machine tool, iterative learning control has acquired some useful application results (Xie, Z.D et al., 2000) However, the complex industrial plant like robot not only possesses high nonlinear properties but also operates in an environment with external uncertainties in most cases. Therefore, it is more significant to investigate the robust learning control strategy for nonlinear system in the presence of uncertain disturbances.

Since neural network not only has the satisfactory capacity of approximating any nonlinear mapping but also can learn and adapt to the dynamical property of unknown system, neural network based control system has fairly strong adaptability and robustness (He, Y.B. and X.Z. Li, 2000). In resent years, neural network control considered as a new approach has been applied to robot control and obtained some research results. When introducing neural network to identify and control the nonlinear system, a double-neural network structure is to be used in most cases. One is to learn positive model of the controlled system as a identifying one itself, another is used to learn the inverse model as a controller. But the structure may lead to more parameters from controller to be adjusted, and stability and robustness of the closed loop system cannot be ensured. In (Li, M.Z. and F.L. Wang, 1998), on the basis of positive model identification of neural network, the control problem was converted to an optimizing one and then processed iterative solution. But it remains to be further studied to advance the precision of neural-identifying model and to select weight coefficients and step factors. A neural network controller with iterative learning algorithm is presented in (Wang, C.Q, 1998) incorporating feedback control actions, in order to overcome the uncertainties and load disturbances of model. The neural network based controller was to directly realize inverse-dynamic control, which means that the plant must be dynamically invertible, and thus the tracking precision lay on the precision of the inverse model. A case of existing uncertainties and parameter varieties was considered in (Ozaki T and Suzuki T, 1991) where two neural network controllers were employed to identify different parts of the model so as to compensate effects of uncertainties and parameter varieties. But this kind of neural network structure may induce many tuning parameters and need much more repetitive trials. Usually, there isn't standard procedure to select the structure of neural network and effective algorithm; the training numbers over hundreds of neural network and the low convergence rate become one of the primary open problems. Simulation results in (Wang, C.Q, 1998) and (Ozaki T and Suzuki T, 1991) illustrated that the learning numbers and tracking error performance of iterative learning control based on neural network would be modified greatly.

This paper presents a new neural network based iterative learning control algorithm, which combines iterative learning control with neural network identification for the purpose of trajectory tracking control of robot. As neural network has the ability of self-learning, that utilizes the prior output data of uncertain system to estimate iteratively the system static state property to achieve ideal approaching precision for identification of positive model, a robust iterative learning control scheme on the basis of the better positive model is designed. The neural network is used to identify the positive model of the nonlinear system on iterative axis, which can give feed forward actions of iterative learning controller to reduce the effects of nonlinearities and model uncertainties. Meanwhile, the feedback actions of iterative learning controller make joint movement follow the desired trajectory on time axis by using the control parameters derived by the neural network. That is, after obtaining better approaching precision of network training for model identification iteratively trail by trail, the feed-forward actions of iterative learning control law of the next trail are constructed by the output signals of the neural network, and then integrated with feedback control to track the desired trajectory of robot in real time. The feedback control is introduced to compensate effects of both errors of identification and iterative learning, so the controlled system can get better robustness and control precision. As there exist many architectures of neural network, the paper uses the most common multi-layered neural network to identify the positive model. Simulation results indicate that the method is very effective to robotic systems with unknown external disturbances, and it can also acquire satisfying tracking performance by fewer numbers of network training and iterative learning processes.

2. MODEL IDENTIFICATION BASED ON NEURAL NETWORK

System identification is a basic and important work for the control system design. But identification of the complex system is a more difficult and challengeable issue. Robot is a kind of high nonlinear, close coupled and time-varying dynamical system, with the effects of model uncertainties and external disturbances, so it is difficult to establish its precise dynamical model. Owing to complicated mechanism of robot and many unknown uncertainties including measurement errors, the conventional methods of identification would not suit to high precision control of robot. While the nonlinear approximating property and the high parallel operation ability of neural network provide a valid way for identification of complex systems, especially for nonlinear system.

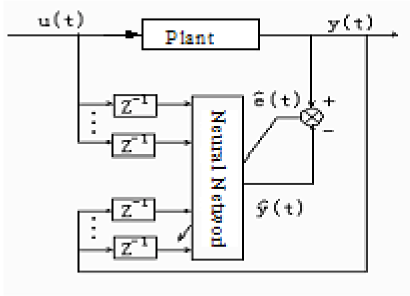


Fig.1. Architecture of the neural network based positive-model

Using multi-layered static network to get the positive model of nonlinear system, the controlled plant can be considered as a "black-box", which means that it is unnecessary to analyze exactly the internal structure of unknown process or plant. As an approximate model of the actual system, if the neural network based model can give sufficiently small identification error, the output signals of the model can be considered as the output estimates of the actual plant. Therefore the following conventional MIMO nonlinear input-output discrete time plant is considered

$$\begin{aligned} y(t) = f(&y(t-1), y(t-2), \cdots, y(t-n); \\ &u(t-1), u(t-2), \cdots, u(t-m)) \end{aligned} \tag{1}$$

where

$$u(t) = [u_1(t), u_2(t), \cdots, u_p(t)]^T \in R^p,$$
$$y(t) = [y_1(t), y_2(t), \cdots, y_q(t)]^T \in R^q$$

are the plant inputs and the plant outputs vectors of dimensions *p* and *q*, respectively, *m, n* are called as model orders and assumed to be known, and *f* is allowed as an unknown nonlinear input/output vector function of dimension *q*, *i.e.*

$$f(x) = [f_1(x), f_2(x), \cdots, f_q(x)]^T.$$

Eqn. (1) can be simplified as

$$y(t) = f(I(t-1)) \tag{2}$$

where

$$\begin{aligned} I(t-1) = [&y(t-1)^T, \cdots, y(t-n)^T, \\ &u(t-1)^T, \cdots, u(t-m)^T] \in R^{nq+mp} \end{aligned}.$$

It is pointed out in (He, Y.B. and X.Z. Li, 2000) that a feed-forward neural network with simple hidden layer has the capability of approximating arbitrary nonlinear function if there are enough nodes on the hidden layer of the neural network. A neural network based positive model structure of the plant is illustrated in Fig.1, where the used neural network is a three-layered back propagation network showed in Fig.2. Then the neural network based identifying model can be described as follows

$$\hat{y}(t+1) = N(I(t), W) \tag{3}$$

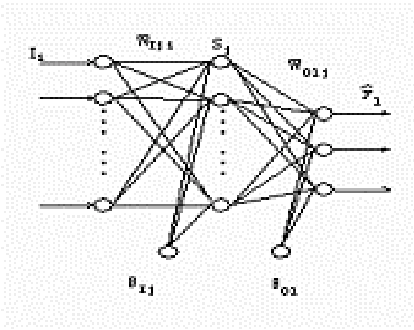


Fig.2. Architecture of the three-layered BP neural network

where W is the synaptic weights vector, N is the input/output mapping function of the neural network; $I(t)\in R^{nq+mp}$ represents the inputs of the neural network and $\hat{y}(t+1)\in R^{q}$ represents the neural network outputs composed of N_O output neurons. So the number of the output neurons of the neural network can be determined easily, *i.e.* $N_O = q$. In the architecture of the three-layered BP neural network showed in Fig.2, a nonlinear Sigmoid function in the hidden layer and a linear function in the output layer will be employed, and the outputs of the hidden layer and the output layer can be expressed as follows

$$S_j = f_s(net_j) = f_s(\sum_{i=1}^{N_I} W_{Iji} I_i + \theta_{Ij}),$$
$$j = 1,2,\cdots,N_H \tag{4}$$

$$\hat{y}_l = (\sum_{j=1}^{N_H} W_{Olj} S_j + \theta_{Ol}), \quad l = 1,2,\cdots,N_O \tag{5}$$

where $f_s(x) = \dfrac{1}{1+e^{-x}}$ is a sigmoid function, W_{Iji}, W_{Olj}, θ_{Ij}, θ_{Ol} are the connection weights and threshold values responding to the input layer to the hidden layer and the hidden layer to the output layer, respectively, I_i denotes the input of the input layer, net_j is the input of the hidden layer, S_j represents the output of the hidden layer, $\hat{y}$ is the outputs of the output layer, N_I, N_H, N_O represent the neuron number of the input, hidden and output layer, respectively. Define identification error of the neural network as

$$\begin{aligned} e_n(t+1) &= y(t+1) - \hat{y}(t+1) \\ &= f(I(t)) - N(I(t),W) \end{aligned} \tag{6}$$

When the neural network being trained sufficiently, the optimal weighting value W^* can be obtained and it holds that

$$\begin{aligned} &\left\| f(I(t) - N(I(t),W^*) \right\| = \left\| e_n(t+1) \right\| \le \varepsilon, \\ &\forall I(t) \in D \end{aligned} \tag{7}$$

where ε is a sufficiently small positive constant standing for the given approximating precision, D is a strict compact set in R^{nq+mp}. However, in this paper the network training is integrated with iterative learning control. In the kth trial, minimize the following quadratic cost function to get an optimal weighting value W_k^* firstly by using all input-output data of this trial

$$J_k = \frac{1}{2}\sum_{t=0}^{T}[y_k(t) - \hat{y}_k(t)]^2, \quad k = 1,2,\cdots \tag{8}$$

where T is the period of the trials, $y_k(t)$ and $\hat{y}_k(t)$ represents the system outputs and the network outputs of the kth trial, respectively. To solve W_k^* from (8), the steepest descent algorithm is employed in the paper.

$$W_{r+1} = W_r - \beta\frac{\partial J}{\partial W_r} + \alpha\Delta W_{r-1}, r = 1,2,\cdots \tag{9}$$

where β is a learning rate and α is a momentum factor. The function of the momentum factor is to memorize the changing direction of connection weights in previous training procedure and restrain vibration of the system that may be produced called a smooth action. During the kth trial, the weight W_r can be modified recursively along optimal direction as the training number r increases, so one hopes that the identification error of model can be reduced gradually. When themodel satisfies the given approximating precision,the training process of (9) will be completed andthe final weight obtained from (9) will be set as W_k^*. Then using W_k^* calculates the network outputs defined by $\hat{y}_k^*(t)$ $(t\in[0,T])$ based on (4)and (5) for each trial that will be applied to construct the feed-forward actions of the k+1th iterative learning control law, and combined with feedback control in real time to produce the control inputs u_{k+1} that will be described in next section.

3. NEURAL NETWORKS BASED ITERATIVE LEARNING CONTROL FOR ROBOT

The dynamics equation of an n-degree-of –freedom robot can be described in the following

$$\begin{aligned} &M(\theta(t))\ddot{\theta}(t) + V(\dot{\theta}(t),\theta(t)) \\ &\qquad + G(\theta(t)) + T_a(t) = \tau(t) \end{aligned} \tag{10}$$

where $\theta(t)\in R^p$ is the vector of generalized joint position, $M(\theta(t))\in R^{p\times p}$ is a symmetrical positive inertia matrix; $V(\dot{\theta}(t),\theta(t))\in R^p$ is the vector representing centrifugal and Coriolis; $G(\theta(t))\in R^p$ is the vector of gravitational term; $\tau(t)\in R^p$ is the vector of joint torques supplied by the actuators and $T_a(t)\in R^p$ is an unknown term arising from bounded disturbances. Due to the uncertainties and external disturbances of the robotic dynamic model, it is impossible to get the exact value of generalized joint position. The paper coordinates a P-type iterative learning controller with identification model of neural network. Regarding the disturbances as a part of the system itself, the neural network is employed to identify the whole nonlinear system so as to make the outputs of network approach the actual outputs of the system infinitely. As an identified model of the controlled plant, if the model error ε is small enough, the outputs of neural network can be considered as the actual outputs of the controlled system, *i.e.* $y(t) \approx \hat{y}(t)$. In order to improve robustness of the controlled system and reduce the influence of nonlinear uncertainties and disturbances to control performance, a feed-forward compensation action is firstly introduced based on the iterative learning controller that may be either a conventional controller such as PID, PI, P-type controller or an intelligent controller like fuzzy controller and expert controller, but a simple PD-type controller is used in the paper. However an extension to other type of controllers is easily made. Suppose that u_{fb} is output of feedback controller and u_{ff} is the one of feed-forward controller based on the neural network identification. Then a compound control law $u(t)$ composed of u_{fb} and u_{ff} will be derived. The block diagram of architecture of control system is illustrated in Fig.3.

In the *k*th iterative learning control process, it can be known from the diagram that the control law of robot trajectory tracking is,

$$u^k(t) = u_{fb}^k(t) + u_{ff}^k(t) \tag{11}$$

where $u_{fb}^k(t) = k_p e_p^k(t) + k_d \dot{e}_p^k(t)$ is the feedback control action, k_p, k_d are the positive matrixes of position and velocity gains, respectively, $e_p^k(t) = y_d(t) - y_p^k(t)$, $y_d(t)$ is the desired trajectory of the system, $y_p^k(t)$ is the actual output including model uncertainties and external disturbances in the *k*th trial. But u_{ff} is obtained by

$$u_{ff}^{k+1}(t) = u_{ff}^k(t) + k_{ILC} e_n^k(t) \tag{12}$$

which is of the iterative learning controller, where k_{ILC} is the learning control gain matrix, $e_n^k(t) = y_d(t) - y_n^k(t)$, $y_n^k(t)$ is the output of neural network in the *k*th trial. To guarantee the convergence of iterative learning control algorithm (Pi, D.Y. and Y.X. Sun, 1999), the selection of k_{ILC} should satisfy

$$\rho(I - k_{ILC} D(0)) < 1 \tag{13}$$

where D is the close-loop transfer function matrix of the system, $\rho(\cdot)$ represents the corresponding spectrum radius.

For a two-degree-of-freedom robot, the orders *p* and *q* of the system are both 2, and maximum differential degree of its mathematical model is 2 also. The input signals of neural network are fed by vectors of the plant input signals and the desired trajectory signals with delay degrees 0,1 and 2. The neuron numbers of the input layer, hidden layer and output layer are $N_I = 12, N_H = 10$ and $N_O = 2$, respectively. The training algorithm of the network is described by (9). It is observed from next section that rather short learning time is needed in general.

In conclusion, the iterative learning control scheme proposed in this paper can be summarized as follows:

1) For $k = 0$, give an initial weight W_0^* of W_k^* to produce $\hat{y}_n^k(t)\ (t \in [0,T])$ based on (4) and (5), and only feedback control action is considered in (11).

2) For $k \geq 1$, use (9) to derive W_k^* and then (12) to calculate the feed forward action $u_{ff}^k(t)$.

Furthermore an iterative learning control law u_{k+1} resulting from (11) will be available. The process of iterative learning control is detached from neural network training. When the *k*th procedure of iterative learning control is completed, the neural network is

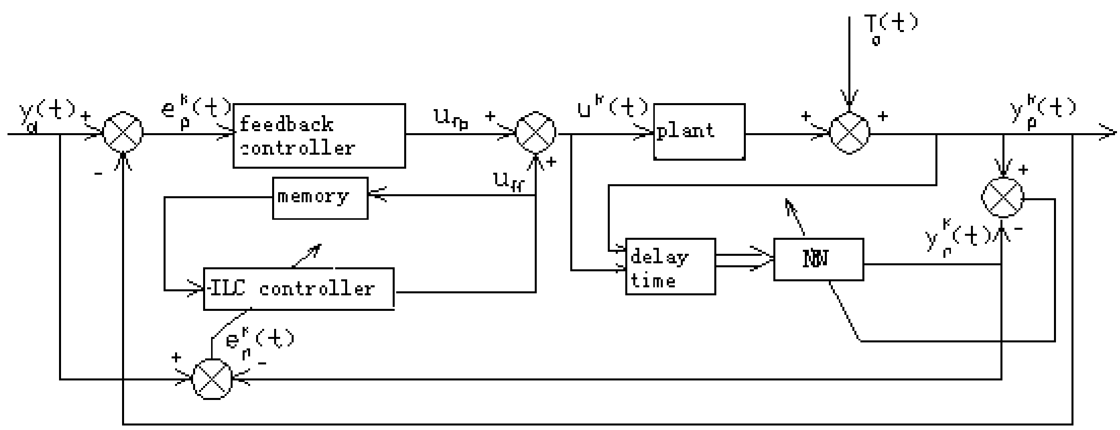


Fig.3. The block diagram of architecture of control system

then trained by (9) using all inputs and outputs in the kth trial.

4. SIMULATION AND ANALYSIS

The robot used in simulation is a two-degree-of-freedom SCARA-type one given in (Ozaki T, Suzuki T, 1991). The expression of each term in the robotic dynamic equation is shown as follows

$$M_{11} = m_1 K_1^2 + m_2(L_1^2 + K_2^2 + 2L_1 K_2 \cos(\theta_2)) + I_1 + I_2$$

$$M_{12} = m_2(K_2^2 + L_1 K_2 \cos(\theta_2)) + I_2$$

$$M_{21} = M_{12}$$

$$M_{22} = m_2 K_2^2 + I_2$$

$$V_1 = -m_2 L_1 K_2 \sin(\theta_2)(2\dot{\theta}_1 + \dot{\theta}_2)\dot{\theta}_2 + D_{m1}\dot{\theta}_1$$

$$V_2 = m_2 L_1 K_2 \sin(\theta_2)\dot{\theta}_1^2 + D_{m2}\dot{\theta}_2$$

$$G_1 = g((m_1 K_1 + m_2 L_1)\cos(\theta_1) + m_2 K_2 \cdot \cos(\theta_1 + \theta_2))$$

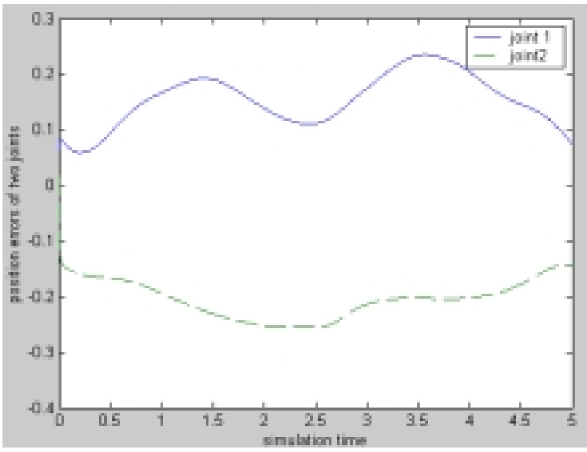
$$G_2 = g m_2 K_2 \cos(\theta_1 + \theta_2)$$

where the following physical parameters of the robot with two links are, arm length L_1=0.25m, L_2=0.16m; link centers of gravity K_1=0.2m, K_2=0.14m; mass m_1=9.5kg, m_2=5.0kg; inertia $I_1 = 4.3\times10^{-3} kg\cdot m^2$, $I_2 = 6.1\times10^{-3} kg\cdot m^2$; motor damping coefficients $D_{m1} = 3.85\times10^{-3} N\cdot s\cdot m^{-1}$, $D_{m2} = 1.39\times10^{-3} N\cdot s\cdot m^{-1}$; gravitational acceleration $g = 9.81 m\cdot s^{-2}$.

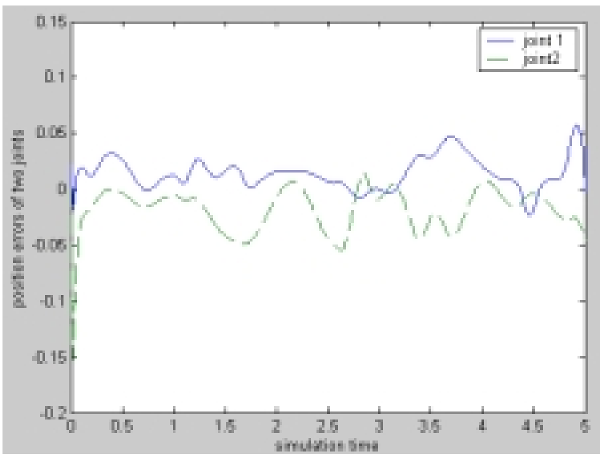
The desired trajectories of two joints are $\theta_{1d} = -\frac{\pi}{2}\cos(\pi t/2.5), \theta_{2d} = \frac{\pi}{2}\sin(\pi t/2.5)$; the gain matrixes of the feedback controller are set at k_p=diag[300, 300], k_d= diag[20, 20]; the external disturbance was $T_a = [0.13\cos(\pi t/10 + 5), 0.23\sin(\pi t/10 + 4)]^T$; the gain matrix of learning controller is K_{ILC}=diag[100,100]; the connection weights of the neural network are randomly initialized between（-0.5,0.5）; the momentum factor is α=0.9, the learning rate is β=0.01. It takes about 5s for simulation, and the sampling period is 0.01s. The numbers of iterative learning and neural network training for one iterative learning procedure are both 20. As convergence of the BP network depends on the initial weights of its learning mode, we would reinitialize the connection weights at the outset of each learning trial. Fig.4 showed the tracking error of the manipulator with two joints, and (a)—(d) illustrate the error curves of the first, 7th,14th, and 20th iterative learning trial. At the 20th iterative learning control process, the index curve of training performance for the neural network identifying model was shown in Fig.5. It is easy to see from Fig.4(a)—(d) that the error curves of two joints possess clearly convergent trend as iterative learning times increase. At the 20th iterative learning control process, the error satisfied the requirement of better tracking precision. From Fig.5, it is clear that the performance criterion of the BP network training attain to 1e-6 level. Whereas the simulation results in (Wang, C.Q, 1998) presented that there existed certain error between the actual and desired trajectory of the joints when only feedback control was operated. At the 85th iterative learning control procedure, the square sum of tracking errors on two joints are 0.0059 and 0.064, respectively. As far as other kinds of external disturbances are concerned, such as a noise signal, an impulse at any time, simulations are also performed in the paper. The results show that the proposed scheme can also get rather good requirement of tracking precision.

5. CONCLUSION

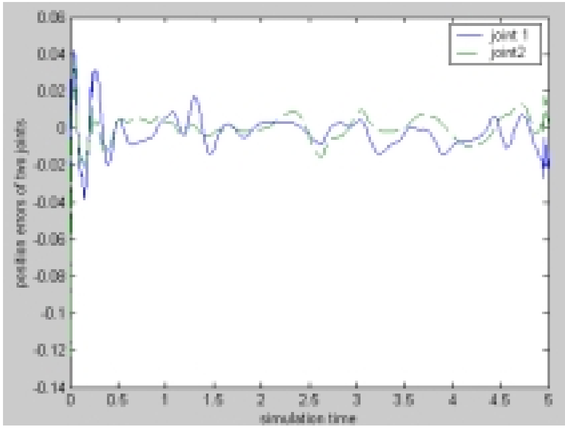
The paper presents a method of iterative learning control combining with identifying model of neural network. The BP neural network is employed to identify the nonlinear system and to produce the feed-forward action of iterative learning control algorithm, and it is integrated with feedback control in real time to form the neural network based robust iterative learning control algorithm. The scheme



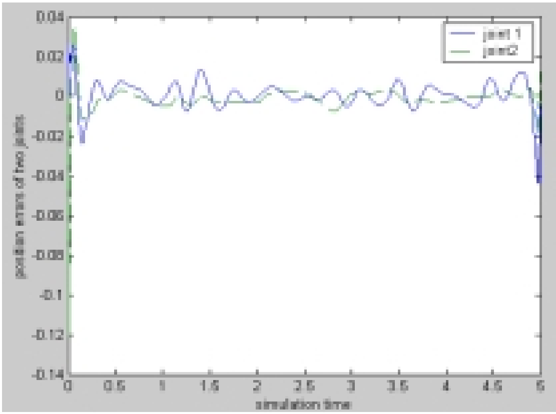
(a)



(b)



(c)



(d)

Fig.4. The tracking error curves of two joints:
(a) First trial; (b) 7th trial; (c) 14th trial; (d) 20th trial.;

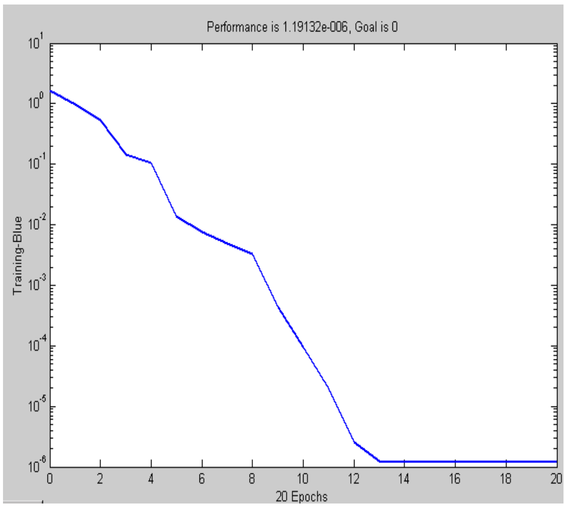


Fig.5. The performance index curve of the neural network identifying model

makes the robotic controller have the ability of self-learning and eliminate the influences of uncertainties and external disturbances of the dynamical model. Further analysis performed in the paper indicates that the control system can realize high-precision tracking to any trajectory on the condition that the identification precision of neural network is good enough. Moreover, the simulation investigation shows that the neural network based control strategy can be used better for the complex industrial processes.

6. ACKNOWLEDGEMENT

The authors would like to acknowledge the supports of Foundation for University Key Teacher by the Ministry of Education of China, the State Science and Technology Pursuit Plan Project of China (2001BA204B01), and the Hong Kong Research Grant Council (HK UST RIG95/96. EG03).

REFERENCES

Li, X.Z., L.K. Jian and Y. He (1997). Iterative Learning Control in Continual Trajectory Tracking of Nonlinear System. *Machine Tool and Hydraulic Pressure*, **5**, 22-23.

Li, M.Z. and F.L. Wang (1998). A Neural Network-Based Iterative Optimal Controller for Nonlinear Systems. *Journal of Northeastern University (Natural Science)*, **19(2)**, 191-194.

Wang, C.Q (1998). Iterative Learning Control for Robot Manipulators Using Neural Networks. *Journal of Nanjing University of Aeronautics & Astronautics*, **30(4)**, 395-399.

Ozaki T, Suzuki T (1991). Trajectory control of robotic manipulators using neural networks. *IEEE Trans on Industrial Electronics*, **38(3)**, 195-202.

He, Y.B. and X.Z. Li (2000). *Technology and Ap plications of Neural Network Control*, Science Publisher.

Xie, Z.D., S.L. Xie and Y.Q. Liu (2000). Development and Expectation for Learning Control Theory of Nonlinear System. *Control Theory and Application*, **17(1)**, 4-8.

Pi, D.Y. and Y.X. Sun (1999). The Convergence of Iterative Learning Control with Open-closed-loop P-type Scheme for Nonlinear Time-varying System. *Automatica*, **25(3)**, 351-354.

Shao, H.H (1997). *Advanced Control of Industrial Process*, ShangHai Jiaotong University Publisher.