International Journal of Fuzzy Logic and Intelligent Systems, vol. 7, no. 4, December 2007 2007 pp. 236-241

A study on the Adaptive Controller with Chaotic Dynamic Neural Networks

Sang-Hee Kim*, Hee Wook Ahn* and Hua O. Wang**

***School of Electronic Engineering, Kumoh National Instute of Technology**
Gumi Gyungbuk, Korea 730-701
**** Dept. of Aerospace/Mechanical Engineering, Boston University**
Boston, MA 02215, USA

Abstract

This paper presents an adaptive controller using chaotic dynamic neural networks(CDNN) for nonlinear dynamic system. A new dynamic backpropagation learning method of the proposed chaotic dynamic neural networks is developed for efficient learning, and this learning method includes the convergence for improving the stability of chaotic neural networks. The proposed CDNN is applied to the system identification of chaotic system and the adaptive controller. The simulation results show good performances in the identification of Lorenz equation and the adaptive control of nonlinear system, since the CDNN has the fast learning characteristics and the robust adaptability to nonlinear dynamic system.

Key Words : Chaotic Dynamic Neural Network, Adaptive Control, Dynamic Backpropagation, System Identification

1. Introduction

Recently, the chaotic neural networks(CNNs) have been studied in application to nonlinear dynamic systems because of it's highly nonlinear dynamic characteristics. Biological neurons generally have chaotic characteristics permanently or transiently. The chaotic responses of biological neurons have been modeled quantitatively by many researchers. The primitive model was the Hodgkin-Huxley equation. Caianiello and Nagumo-Sato modified this model to make chaotic neural networks.[1] Aihara et al. proposed a discrete time model with continuous output, and applied this model to chaotic neural networks.[2] They showed that the neural networks could be applied to solve optimization problems such as traveling salesman problem(TSP). The effects of chaotic response have not verified yet by analytical methods. The chaotic characteristics of neuron model generally gives adverse effects on optimization problems, but the transient chaos of neuron model could be beneficial to overcome the local minimum problem. Aihara proposed that the transient chaotic characteristics of neuron could be helpful for global optimization.[3] Even though some modifications performed on chaotic neuron model, those previously proposed chaotic neuron models are still complicate, and need more dynamic characteristics in neuron itself and learning algorithm[4].

Manuscript received Oct. 7, 2007; revised Dec. 12, 2007.
This paper was supported by Research Fund, Kumoh National Institute of Technology

Since conventional CNNs' structure and learning rule does not proper to system identification and control application, a modified chaotic dynamic neuron model is presented to simplify the model and to enforce dynamic characteristics. The chaotic dynamic neural networks consist of the modified chaotic dynamic neurons. A dynamic backpropagation the learning algorithm is developed for the proposed CDNN. This structure is very compatible with highly nonlinear dynamic system in the neural network structure and learning rule. Chaotic dynamic neural networks could be substituted with feed-forward neural networks and recurrent neural networks because of it's complex nonlinearity in chaotic neurons.

L. Jin et al proposed absolute stability conditions for discrete-time recurrent neural networks and stable dynamic backpropagation learning in recurrent neural networks[6,7]. The stability of CDNN is more important factor than recurrent neural networks. Even though the CDNN has the fast adaptabilities and highly nonlinear dynamic characteristic, this network has a problem in stability. To improve the stability of CDNN, the previously induced convergence conditions are applied to the dynamic backpropagation learning method.

In this paper, a dynamic backpropagation learning rule for CDNN with stability conditions are proposed and applied to the system identification of a chaos system and an indirect adaptive controller. The simulation results show good performances, since the CDNN has the robust adaptability to nonlinear dynamic system.

2. Dynamic Backpropagation Learning of CDNN

2.1 Modified Chaotic Dynamic Neural Networks

Although the chaotic neuron model inherently has robust dynamic characteristics, the traditional chaotic neural networks(CNN), proposed by Aihara et al, decrease the dynamic characteristics in the structure and the learning rules. They used the backpropagation learning rule for the forward inputs between layer and used the time progressing learning rule(the continuous Hopfield learning algorithm) for the recurrent inputs in inter layer. These learning rules may be appropriate to the static patterns but not to the dynamic system applications such as forecasting, identifications, signal processing and dynamic system control. In this paper, the structure of CNN is modified, and the new learning rule is proposed for enhancing the dynamic characteristics.

Modified chaotic dynamic neural network is a globally coupled neural networks. Each chaotic neuron unit is globally coupled with present and past outputs of chaotic neuron units. Modified chaotic dynamic neural networks in Fig. 1 have two different coupling coefficients (weights) for both directions among the neurons of interlayer, and forward direction between layers.

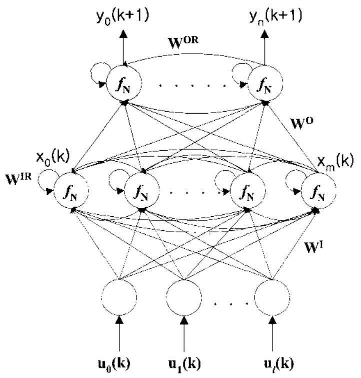


Fig. 1. Chaotic Dynamic Neural Networks

This connection weights in interlayer is defined as nonsymmetric form, $w_{ij}^{OR} \neq w_{ji}^{OR}$, $w_{ii}^{OR} \neq 0$ and $w_{ij}^{IR} \neq w_{ji}^{IR}$, $w_{ii}^{IR} \neq 0$. This structure is similar with fully recurrent neural networks.

The modified chaotic neural network is as follows,

$$x_i(k+1) = K \cdot x_i(k) + \sum_{j=1}^{n} w_{ij}^F \cdot I_j(k) + \sum_{j=1}^{m} w_{ij}^R \cdot y_i(k) \quad (1)$$

$$y_i(k+1) = f_N[x_i(k+1)] \quad (2)$$

$$f_N[x_i(k+1)] = \frac{1}{1+e^{-x_i(k+1)/\varepsilon}} \quad (3)$$

where ε is slope of sigmoid function.

To increase dynamic characteristics, the nonsymmetric weights are applied to recurrent inputs such as $w_{ij}^R \neq w_{ji}^R$, $w_{ii}^R \neq 0$. The chaotic neuron sums three inputs; the refractoriness, $K \cdot x_i(k)$, the activation, $\sum_{j=1}^{n} w_{ij}^F \cdot I_j(k)$, and the recurrent input, $\sum_{j=1}^{m} w_{ij}^R \cdot f_N(x_i(k))$. The summation passes through the nonlinear sigmoid function.

2.2 Dynamic backpropagation learning

Consider fig. 1, $u_i(k)$ is the i th input for each discrete time k, $S_j^H(k)$ is the weighted sum of inputs and refractory input to j th neuron in hidden layer, $x_j(k)$ is the output of j th neuron in hidden layer, K is refractory parameter of chaotic neuron, and $f_N(\cdot)$ is nonlinear sigmoid function. $\mathbf{W}^I$ and $\mathbf{W}^{IR}$ represent the weight vector between input and hidden layer and inter-connecting weight vector in the hidden layer. The weighted sum of j th neurons in hidden layer is as follows:

$$S_j^H(k) = \sum_{i=1}^{l} w_{ij}^I u_i(k) + \sum_{q=1}^{m} w_{qj}^{IR} x_q(k-1) + K \cdot S_j^H(k-1) \quad (4).$$

The j th neuron's output of hidden layer is as follow:

$$x_j(k) = f_N[S_j^H(k)] \quad (5).$$

Consider Fig. 1, $y_p(k)$ is p th output of output neuron for each discrete time k, $S_p^O(k)$ is the weighted sum of inputs and refractory input to p th output neuron in output layer, $x_j(k)$ is the output of j th neuron in hidden layer, K is refractory parameter of chaotic neuron, and $f_N(\cdot)$ is nonlinear sigmoid function. $\mathbf{W}^O$ and $\mathbf{W}^{OR}$ represent the weight vector between hidden and output and inter connecting weight vector in the output layer. The weighted sum of p th neurons in hidden layer is as follows:

$$S_p^O(k) = \sum_{j=1}^{m} w_{jp}^O x_j(k) + \sum_{r=1}^{n} w_{rp}^{OR} y_r(k-1) + K S_p^O(k-1) \quad (6)$$

$$y_p(k) = f_N[S_p^O(k)] \quad (7)$$

Using Eq. (4)(5), the weighted sum of neuron in output layer(Eq. 6) can define as follows:

$$\begin{aligned} S_p^O(k) = & \sum_{j=1}^{m} w_{jp}^O f_N[\sum_{i=1}^{l} w_{ij}^I u_i(k) + \sum_{q=1}^{m} w_{qj}^{IR} x_q(k-1) \\ & + K S_j^H(k-1)] + \sum_{r=1}^{n} w_{rp}^{OR} y_r(k-1) + K S_p^O(k-1) \end{aligned} \quad (8)$$

$$\begin{aligned} = & \sum_{j=1}^{m} w_{jp}^O f_N[\sum_{i=1}^{l} w_{ij}^I u_i(k) + \sum_{q=1}^{m} w_{qj}^{IR} f_N[\sum_{i'=1}^{l} u_{i'}(k-1) \\ & + \sum_{j=1}^{m} w_{jq}^{IR} x_j(k-2) \quad + K S_q^H(k-2)]] \end{aligned} \quad (9)$$

$$+ \sum_{r=1}^{n} w_{rp}^{OR} f_N[\sum_{j'=1}^{m} w_{j'r}^O x_{j'}(k-1) + \sum_{r'=1}^{n} w_{r'r}^{OR} y_{r'}(k-2)$$

$$+kS_r^O(k-2)]+KS_p^O(k-1)$$

$$O_p(k)=NF(u(l),x(l),y(l); \quad l\le k) \tag{10}$$

where $O_p(k)$ is the p th output of chaotic neural network, $NF(\cdot)$ is a nonlinear function which represents a nonlinear dynamic mapping chaotic neural networks.

This neural network model in Eq. (10) is a globally coupled with present and past inputs and outputs of all neurons. Therefore, this model could simulate any complex nonlinear dynamic system.

The dynamic learning process may be formulated as:

$$\boldsymbol{W}(k+1)=\boldsymbol{W}(k)-\eta\cdot\nabla_w\boldsymbol{E}(k) \tag{11}$$

where $\boldsymbol{W}(k)$ is an estimated weight vector at time k and η is a step size parameter, which affects the rate of convergence of the weights during learning.

The error index $\boldsymbol{E}(k)$ should be defined as

$$\boldsymbol{E}(k)=\tfrac{1}{2}\sum_{i=1}^{n}[y_i^d(k)-y_i^m(k)]^2 \tag{12}$$

$$=\tfrac{1}{2}\sum_{i=1}^{n}e_i^2(k)$$

where $e_i(k)=y_i^d(k)-y_i^m(k)$ is a learning error of i th neuron between the desired and network output at time k.

The gradient of error index with respect to an arbitrary weight vector W is represented by

$$\nabla_w\boldsymbol{E}(k)=-e(k)\nabla_w\boldsymbol{y}^m(k)=-e(k)\nabla_w\boldsymbol{O}(k) \tag{13}$$

where $\boldsymbol{O}(k)$ is output vector of neural network, and $\boldsymbol{y}^m(k)=\boldsymbol{O}(k)$ in case simple identification task. The output gradient $\nabla_w\boldsymbol{O}(k)$ with respect to output weights, interconnecting of output, interconnecting of hidden, and input weight in Eq. (13) are given by

$$\frac{\partial\boldsymbol{O}(k)}{\partial w_{ij}^O}=f_N'[S_j^O(k)]A_{ij}^O(k) \tag{14}$$

$$\frac{\partial\boldsymbol{O}(k)}{\partial w_{ij}^{OR}}=f_N'[S_j^O(k)]A_{ij}^{OR}(k) \tag{15}$$

$$\frac{\partial\boldsymbol{O}(k)}{\partial w_{ij}^{IR}}=[\sum_{r=1}^{n}f_N'(S_r^O(k))\cdot w_{jr}^O]\cdot f_N'(S_j^H(k))\cdot A_{ij}^{IR}(k) \tag{16}$$

$$\frac{\partial\boldsymbol{O}(k)}{\partial w_{ij}^{I}}=[\sum_{r=1}^{n}f_N'(S_r^O(k))\cdot w_{jr}^O]\cdot f_N'(S_j^H(k))\cdot A_{ij}^{I}(k) \tag{17}$$

where

$$A_{ij}^O(k)=O_i(k)+[w_{jj}^{OR}f_N'(S_j^O(k-1))+K]A_{ij}^O(k-1) \quad ,$$
$$A_{ij}^O(0)=0, \tag{18}$$

$$A_{ij}^{OR}(k)=O_i(k-1)+[w_{jj}^{OR}f_N'(S_j^O(k-1))+K]A_{ij}^{OR}(k-1) \quad ,$$
$$A_{ij}^{OR}(0)=0, \tag{19}$$

$$A_{ij}^{IR}(k)=x_i(k-1)+[w_{jj}^{IR}f_N'(S_j^H(k-1))+K]\cdot A_{ij}^{IR}(k-1) \quad ,$$
$$A_{ij}^{IR}(0)=0, \tag{20}$$

$$A_{ij}^{I}(k)=u_i(k)+[w_{jj}^{IR}f_N'(S_j^H(k-1))+K]\cdot A_{ij}^{I}(k-1) \quad ,$$
$$A_{ij}^{I}(0)=0, \tag{21}.$$

2.3 Convergence of dynamic backpropagation learning

The globally asymptotical stability condition of CDNN is applied for dynamic backpropagation learning. The convergence condition is derived in the previous paper of "A study on the Convergence Condition of Chaotic Neural Networks". The convergence is guaranteed to only one fixed point $y^*(k)$ if w^R has the conditions as

$$-(1-K)\cdot S_{f^*}<w^R\le\frac{1-K}{S_{f^*}} \tag{22}$$

Where $S_{f^*}=\max\left|\frac{\partial f_N(y^*(k))}{\partial y^*(k)}\right|$ (23)

The w^R is the recurrent weight for the internal state of CDNN $y(k)$ and S_{f^*} be defined as where $f_N(y(k))$ is sigmoid function. During the backpropagation learning process of CDNN, the absolute stable condition in eq. (22) are maintained in every step of learning.

3. Simulation results with CDNN

3.1 System identification of chaotic system

An approach for system identification using chaotic dynamic neural networks is presented for verifying the performance of CDNN. This example is identify chaotic system (Lorenz equation) that is described by the equation

$$\begin{aligned}\dot{x}&=\sigma(y-x)\\ \dot{y}&=rx-y-xz\\ \dot{z}&=xy-bz\end{aligned} \tag{24}$$

Where $\sigma,\ r,\quad b>0$ are parameters.

The results in Fig 2 show the identification results using the proposed dynamic backpropagation learning method in Eq (14)-(21). In this example, one CDNN is used to identify the chaotic MIMO dynamics. The structure of CDNN identifier consists of three inputs, 30 neurons in hidden layer, and 3 identified outputs. Just one CDNN is used for chaotic identifier for simulating the interconnection between neurons. The learning rate is selected ad 0.35, and refractory rate is 0.7. The figure 2 shows the result after 878 iterations of learning, and the final error was 0.002.

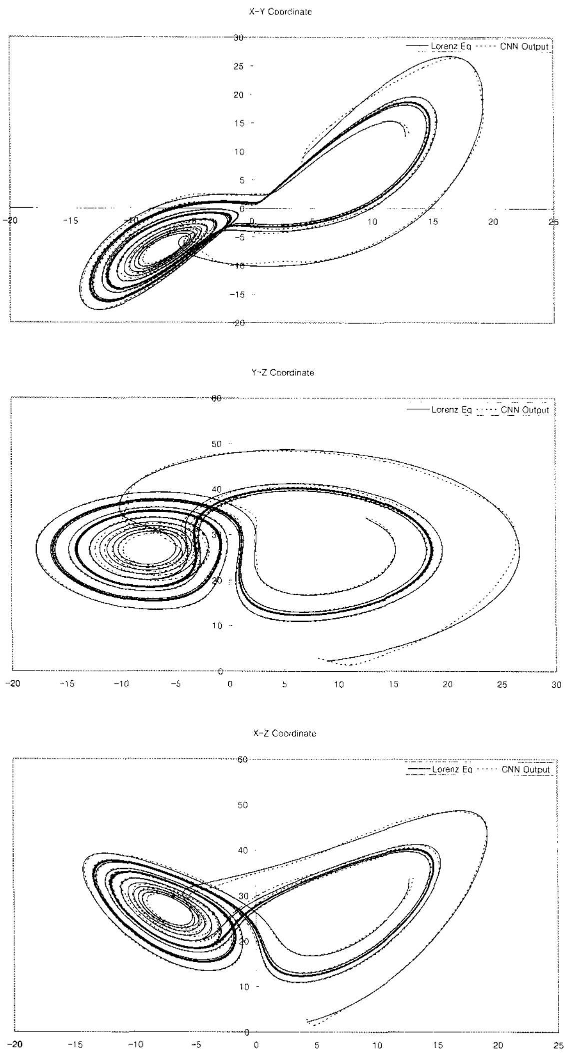


Fig. 2 Identification results of Lorenz Eq

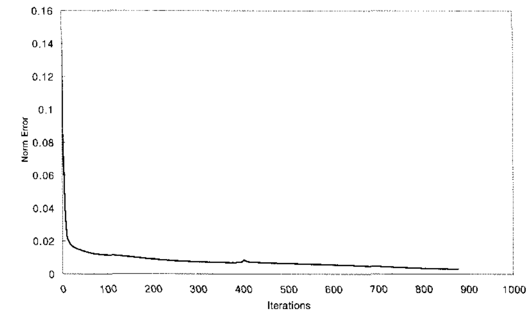


Fig 3. Normalized Error with iterations

3.2 Adaptive control with CDNN

An approach for indirect adaptive control using modified chaotic neural networks is presented. The indirect adaptive control system consists of system identifier and controller. The system identifier with a chaotic neural network, called chaotic neural network identifier(CNNI), identifies an unknown plant for providing unknown plant information to the controller with a chaotic neural network. Both neural networks identifier and controller use dynamic backpropagation algorithm. In identifier, the generalized dynamic backpropagation algorithm could be adopted for adjusting weights of CNNI. In controller, the relationship between the activation value of plant and the plant output should be constructed for adjusting the weights of a chaotic neural network controller(CNNC) in figure 4.

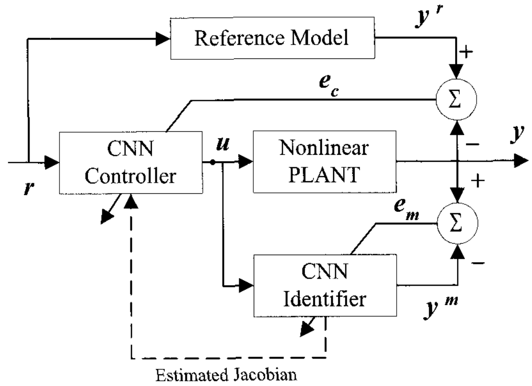


Fig 4. Structure of Adaptive Controller

The gradient of error index with respect to an arbitrary weight vector $\boldsymbol{W}$ of controller should be redefined. The dynamic learning process for CNNC may be formulated as:

$$\boldsymbol{W}(k+1) = \boldsymbol{W}(k) - \eta_c \cdot \nabla_w \boldsymbol{E}_c(k) \tag{25}$$

where $\boldsymbol{W}(k)$ is an estimated weight vector for controller at time k, and η_c is a step size parameter for CNNC which affects the rate of convergence of the weights during learning. The error index $\boldsymbol{E}_c(k)$ should be defined as:

$$\boldsymbol{E}_c(k) = \tfrac{1}{2}\sum_{i=1}^{n}[y_i^r(k) - y_i(k)]^2 = \tfrac{1}{2}\sum_{i=1}^{n} e_{ci}^2(k) \tag{26}$$

where n is number of output in plant, and $e_{ci}(k) = y_i^r(k) - y_i(k)$ is a learning error between the reference model and the plant output at time k. The gradient of error index with respect to an arbitrary weight vector W is represented by

$$\nabla_W \boldsymbol{E}_c(k) = -\boldsymbol{e}_c(k)\nabla_W \boldsymbol{y}(k) = -\boldsymbol{e}_c(k)\nabla_{u(k)}\boldsymbol{y}(k)\nabla_W \boldsymbol{u}(k) = -\boldsymbol{e}_c(k)\nabla_{u(k)}\boldsymbol{y}(k)\nabla_W \boldsymbol{O}^c(k) \tag{27}$$

where $\boldsymbol{e}_c(k)$ is learning error vector at time k, and the plant input vector $\boldsymbol{u}(k)$ is defined as the output vector of CNNC $\boldsymbol{O}^c(k)$.

Since the plant is normally unknown, the sensitivity term $\nabla_{u(k)}\boldsymbol{y}(k)$ could not be defined. After sufficient learning procedure, the learning error of CNNI could approximate to zero. Progressing the learning procedure of CNNI, the outputs of

CNNI is close to the plant output, i.e., $y(k) \approx y^m(k)$. The sensitivity term could be redefined as

$$\nabla_{u(k)} y(k) \approx \nabla_{u(k)} y^m(k) = \nabla_{u(k)} \boldsymbol{O}(k) \tag{28}$$

where $y^m(k) \equiv \boldsymbol{O}(k)$ and $\nabla_{u(k)} \boldsymbol{O}(k) \equiv \frac{\partial \boldsymbol{O}(k)}{\partial \boldsymbol{u}(k)}$.

The jacobian matrix could be defined as

$$J_{ij}(k) = \frac{\partial O_i(k)}{\partial u_j(k)} = \frac{\partial O_i(k)}{\partial S_i^O(k)} \sum_{p=1}^{m} \frac{\partial S_i^O(k)}{\partial x_p(k)} \frac{\partial x_p(k)}{\partial S_p^H(k)} \frac{\partial S_p^H(k)}{\partial u_j(k)} \tag{29}$$

where $J_{ij}(k)$ is an element of jacobian matrix which represents the sensitivity of plant output for input.

Consider Eq. (6)-(9), the partial derivatives can be defined as

$$\frac{\partial S_i^O(k)}{\partial x_p(k)} = w_{pi}^O, \quad \frac{\partial S_p^H(k)}{\partial u_j(k)} = w_{jp}^I.$$

$$J_{ij}(k) = \frac{\partial O_i(k)}{\partial u_j(k)} = f_N'(S_i^O(k)) \sum_{p=1}^{m} w_{pi}^O f_N'(S_p^H(k)) w_{jp}^I \tag{30}.$$

Eq. (27) could be redefined as

$$\nabla_{W^c} \boldsymbol{E}_c(k) = -\boldsymbol{e}_c(k) J(k) \nabla_{W^c} \boldsymbol{O}^c(k) \tag{31}$$

Using negative gradient in (31), the weights for CNNC can be adjusted in eq. (25). The equations (16)-(23) define the dynamic backpropagation learning algorithms for CNNC.

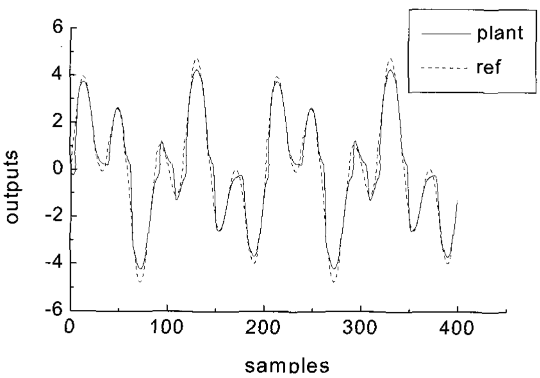
Example 2: The plant is described by the difference equation[4]

$$y^P(k+1) = \frac{y^P(k)}{1 + y^P(k)^2} + u^3(k) \tag{32}.$$

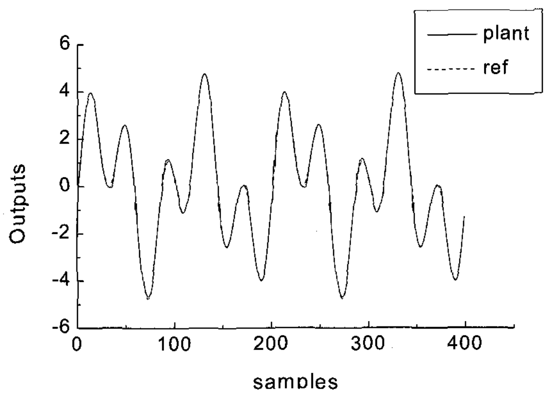
The reference model is described by difference equation

$$y^m(k+1) = 0.6 y^m(k) + r(k) \tag{33}$$

where $r(k) = \sin(2\pi k/25) + \sin(2\pi k/10)$.



(a) After 10 iterations



(b) After 1000 iterations

Fig. 5 Response of Controller

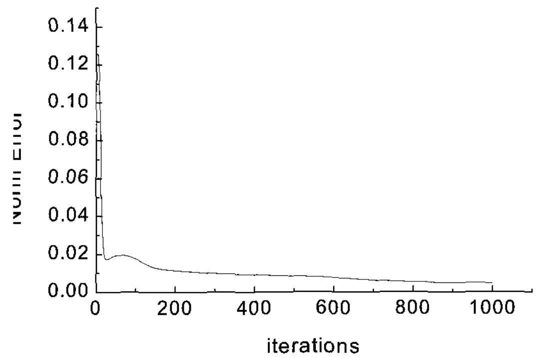


Fig. 6 Normalized Error with iterations

The results in Fig. 5 and 6 show the indirect adaptive neuro-control results using the proposed dynamic backpropagation learning method in Eq. (14)-(21) for CNNI and Eq.(30) and (31) for CNNC. The structure of CNNI consists of two inputs, 7 neurons in hidden layer, and 2 outputs, and CNNC has also same structure. The learning rates are selected as 0.3, and refractory rate is 0.15 for the CNNI and CNNC. Since CDNN has fast adapting characteristics, the CNNI identifies the plant model as on-line learning method.

4. Conclusion

This paper presented a dynamic backpropagation learning rule for CDNN with stability, and the proposed CDNN applied to the system identification of a chaos system and an indirect adaptive controller. The indirect adaptive controller consists of two CDNNs: a CNNI and a CNNC. Traditional CNN was modified to simplify the model and to enforce the dynamic characteristics. The performance of CDNN was tested for two examples: one of them was a nonlinear MIMO system identification for chaotic system, the second was an indirect neuro-adaptive controller. The simulation results show good performances, since the CDNN has the robust adaptability to nonlinear dynamic system.

References

[1] J. Nagumo and S. Sato, "On response characteristics of a mathematical neuron model," *Kybernetik*, vol. 10, pp. 155-164, 1972.

[2] L. Chen and K. Aihara, "Chaotic simulated annealing by a neural network model with transient chaos," *Neural Networks*, vol. 8, no. 6, pp. 915-930, 1995.

[3] L. Chen and K. Aihara, "Global Searching Ablity of Chaotic Neural Networks", *IEEE Trans. Circuits and System-I*, vol 46, No 8, pp. 974-993, 1999.

[4] K. S. Narendra and K. Parthasarathy, "Gradient methods for optimization of dynamical systems containing neural networks," *IEEE Trans Neural Networks*, vol. 2, no. 2, pp 252-262, 1991.

[5] H. K. Khalil, "*Nonlinear System*", Prentice Hall.

[6] L. Jin, N. Nikiforuk, and M. Gupta, "Absolute Stability Conditions for Discrete-Time Recurrent Neural Networks", *IEEE Trans. Neural Networks*, Vol. 5, No. 6, pp. 954-964, 1994.

[7] L. Jin and M. Gupta, "Stable Dynamic Backpropagation Learning in Recurrent Neural Networks", *IEEE Trans. Neural Networks*, Vol. 10, No. 5, pp. 1321-1334, 2000.

Sang Hee Kim
received his M.S degrees in Electrical Engineering from Yonsei University in 1985, and Ph.D degree from Texas A&M University 1992. He is professor at the school of electronic engineering at Kumoh National Institute of Technology from 1993. He performed research at M.I.T. and at Boston University as as research fellow and visiting professor in 2005-2006. His research interests are intelligent control, neural networks, nonlinear dynamics and chaos theory and Digital television.

Phone : 054) 478-7430
Fax : 054) 478-7449
E-mail : shkim@kumoh.ac.kr

Hee Wook Ahn
received his M.S. and Ph.D. degrees in Electrical Engineering from Korea Advanced Institute of Science and Technology(KAIST) in 1990 and 1997 respectively. He is professor at the school of electronic engineering at Kumoh National Institute of Technology from 1998. He performed research at Pennsylvania State University as visiting professor in 2004. His research interests are power system control and power electronics.

Phone : 054)478-7431
Fax : 054)478-7449
E-mail : hwahn@kumoh.ac.kr

Hua O. Wang
received his Ph.D. at University of Maryland. He is Associate Professor of Aerospace and Mechanical Engineering Department, College of Engineering, Intelligent Mechatronics Laboratory. He is an expert in nonlinear control and in particular in the control of systems having complex nonlinear dynamics. He is also affiliated with Boston University's Center for Information and Systems Engineering (CISE). His research interests are Control of nonlinear phenomena and bifurcations.

Phone : (617)353-8860
Fax : (617) 353-5866
Email : wangh@bu.edu