International Journal of
Mathematical Modelling & Computations
Vol. 04, No. 03, 2014, 267 - 276

An Improved Controlled Chaotic Neural Network for Pattern Recognition

Maryam Nahvi[a], Majid Amirfakhrian[b] and Alireza Vasiq[b]

[a]*Department of Computer Engineering,Payam e Noor University, Shahr e Rey, Iran*
[b]*Department of Mathematics, Islamic Azad University, Central Tehran Branch, Tehran, Iran.*

Abstract: A sigmoid function is necessary for creation a chaotic neural network (CNN).In this paper, a new function for CNN is proposed that it can increase the speed of convergence. In the proposed method, we use a novel signal for controlling chaos. Both the theory analysis and computer simulation results show that the performance of CNN can be improved remarkably by using our method. By means of this control method, the outputs of the controlled CNN converge to the stored patterns and they are dependent on the initial patterns. We observed that the controlled CNN can distinguish two initial patterns even if they are slightly different. These characteristics imply that the controlled CNN can be used for pattern recognition.

Received: 29 December 2013; Revised: 15 June 2014; Accepted: 2 August 2014.

Keywords: Chaotic Neural Network, Controlling Chaos, Associative Memory

Index to information contained in this paper

1. Introduction

Recently, chaotic dynamical behavior has attracted a great deal of attention in many research fields, and a great deal of progress in chaotic study has been made [1-5]. People are believe that chaotic dynamic behavior plays an important role in neural networks [6,7]. Many researchers have attempted to model artificial neural networks with chaotic dynamics on the basis of deterministic differential equations or stochastic models. For example, Aihara et al. [5] has proposed deterministic difference equations, which describe an artificial neural network model, composed of chaotic neurons. This model has advantages in terms of computational time and memory for numerical analysis due to the

*Corresponding author. Email: nahvi_maral@yahoo.com.

http://www.ijm2c.ir

spatiotemporal complex dynamics of the neurons. Adachi et al. [1] proposed a system based on model proposed by Aihara et al. Adachi et al. [2] have shown that the CNN can generate chaotic associative memory dynamics in several parameter regions. In particular, Kuroiwa et al. [7] have investigated the dynamical properties of a single chaotic neuron in response to stochastic inputs. On the other hand, Freeman's studies on the olfactory bulb of a rabbit revealed that the dynamics of the neural system in the basic state are chaotic, but in the case that a familiar scent is presented, the system rapidly simplifies its behavior and its dynamics become more ordered and nearly periodic [12]. This suggests an interesting model of the recognition and learning process in biological neural systems. In other words, self-induced chaos control might play a key role in the recognition and learning processes in biological neural systems. To employ this function in an artificial neural system, it is important to study how chaos can be controlled. In other words, it is the realization of the periodic dynamical behavior from the chaotic behavior that results when a stimulus is imparted. In addition, a study on controlling chaos in CNNs is also necessary to understand the mechanism of recognition in a real brain.

Several control methods have been adopted for CNNs. Mizutani et al. [11] proposed a chaos control method in which the chaos in the CNNs was controlled by employing an exponential feedback control signal to change the threshold value of a neuron. Nakamura et al. [12] and Kushiba et al. [8] controlled chaos by changing the parameters of the CNN. The CNN therefore reduces to a Hopfield network in which the outputs of the controlled CNN are fixed points; consequently, pattern recognition and memory search can be achieved. However, in all their methods, the control targets should have been referred to at the beginning of the recognition task. To apply the CNNs to information processing, a self-adaptable chaos control method in which the control targets need not be referred to as a priori is required [6]. In this paper, a new function for CNN is proposed that it can increase the speed of convergence. In the proposed method, we use a novel signal for controlling chaos in CNNs, in which the chaos is controlled in a self-adaptive manner by perturbing parameters of the system with a delay feedback control signal. Then we employ the controlled CNNs to perform pattern recognition. Both the theory analysis and computer simulation results show that the performance of CNN can be improved greatly by using our method.

The rest of this paper is as following. In Section 2, a model of the controlled CNN is reviewed briefly. In Section 3, we scrutinized the controlled CNN that mentioned at section 2. In Section 4, computer simulation results of solving CNN problem, by using the controlled CNN and improved controlled CNN, respectively, are presented. Some conclusions are given in Section 5.

2. **A Controlled Chaotic Neural Network Model**

A chaotic neural network (CNN) constructed with chaotic neurons. In this network, the behavior of associative dynamics for some values of the parameters is chaotic. The structure of a chaotic neural network is shown in Fig. 1 [10].

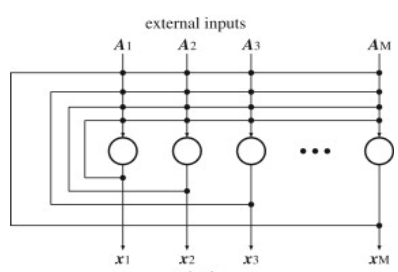

Fig. 1. Structure of chaotic neural network

The dynamics of the simplest chaotic neuron model can be described by the following equations [6]:

$$y(t+1) = ky(t) - \alpha f\big(y(t)\big) + a, \tag{1}$$

$$x(t+1) = f\,(y(t+1)), \tag{2}$$

$$f(x) = \frac{1}{1+e^{\frac{-x}{\varepsilon}}} \tag{3}$$

According to the data mentioned above, $y(t)$ is the internal state of the neuron at time t, $x(t)$ is the output of the neuron at time t, k is the decay parameter of the refractoriness, and α is the refractory scaling parameter. α is a parameter that is based on the external input and the threshold of the neuron; $f(x)$, the sigmoid function, is the output function of the neuron; and ε is the steepness parameter of the sigmoid function.For some values of the parameters, the internal state $y(t)$ and the output $x(t)$ exhibit chaotic behavior. The bifurcation diagram of $y(t)$ is shown in Fig. 2. The other parameters are fixed at $k = 0.5$, $\varepsilon = 0.04$, and $a = 1.0$.

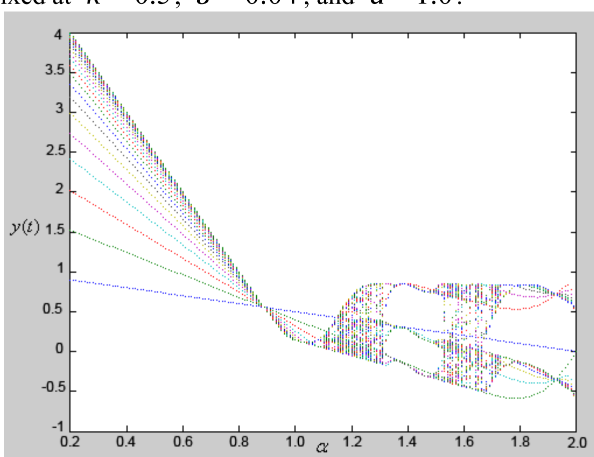

Fig. 2.The bifurcation diagram by using sigmoid function.

We notice that when the value of α is increased from 0.2 to 2.0, there exist two areas in which the neuron is chaotic. One chaotic area exists for a ranging from 1.066 to 1.111, while the other exists for ranging from 1.572 to 1.606, as shown in Fig. 2. The dynamics of the i th chaotic neuron in the CNN at time t can be described simply as follows [6]:

$$x_i(t+1) = f\big(\eta_i(t+1) + \zeta_i(t+1)\big), \tag{4}$$

$$\eta_i(t+1) = k_f \eta_i(t) + \sum_j^N w_{ij} x_j(t), \tag{5}$$

$$\zeta_i(t+1) = k_r \zeta_i(t) - \alpha x_i(t) + a_i \tag{6}$$

Where $x_i(t)$ is the output of the *i*th chaotic neuron for a time step t, and $\eta_i(t)$ and $\zeta_i(t)$ are the internal state variable of the feedback inputs from the constituent neurons in the network and the refractoriness of the *i*th chaotic neuron at time t, respectively. N is the number of neurons in the network. k_f and k_r are the decay parameters of the feedback inputs and the refractoriness, respectively. The parameter a_i is the threshold of the *i*th neuron. As in the case of the chaotic neuron model, the parameter α is the refractory scaling parameter of a neuron, and the output function of the neuron $f(x)$ is a sigmoid function with a steepness parameter ε, as described in Eq. (3). ω_{ij} is the synaptic weight to the *i*th constituent neuron from the *j*th constituent neuron. A neuron does not have the synaptic connection from itself so $\omega_{ii} = 0$. The weights are defined according to the symmetric auto-associative matrix of *n* binary patterns given as follows [6]:

$$w_{ij} = \frac{1}{n}\sum\left(2x_i^p - 1\right)\left(2x_j^p - 1\right), \tag{7}$$

Where x_i^p is the i th component of the P th binary pattern with a discrete value of either 0 or 1.

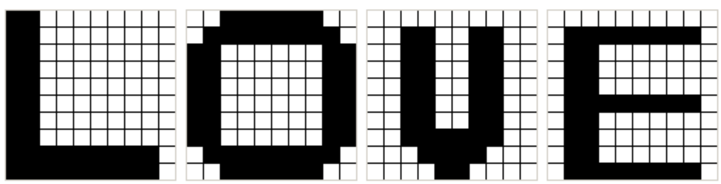

Fig. 3.Patterns to be stored. A pattern comprises 10 × 10 binary pixels.

In accordance with this kind of experiment, the binary patterns can be stored as basal memory patterns in the network. n is the total number of stored memory patterns. The four patterns shown in Fig. 3 are employed as the stored patterns. Each pattern comprises 10×10 binary pixels. Correspondingly, the network is constructed with 100 neurons. A neuron will be represented by a black block ‘‘■’’ when its output, x_i, is equal to 1, which indicates that the neuron is ‘‘excited,’’ while a neuron is denoted by a white block ’’ □’’ when its output is equal to 0, which indicates that the neuron is ‘‘resting’’. According to the synaptic weights learning rule given in Eq. (7), the four stored patterns shown in Fig. 3 are embedded in the CNN. We observe that in Fig. 2, a neuron can create chaotic behavior for certain special parameters and the chaotic dynamics of the neuron depends on α. We therefore assume that chaos would be controlled if the value of α of the chaotic neurons is perturbed by a control signal. If we consider a delay feedback signal to be the control signal in the chaos control process, the control targets do not have to be assigned. This is the main idea of self-adaptable control method in this paper. A controlled chaotic neuron is described as follows [6]:

$$y(t+1) = ky(t) - \alpha\beta^{k_c u(t)} f(y(t)) + a, \tag{8}$$

$$x(t+1) = f(y(t+1)), \tag{9}$$

$$f(x)=\frac{1}{1+e^{-x/\varepsilon}} \tag{10}$$

$$u(x)=\left|x(t)-x(t-\tau)\right| \tag{11}$$

In the case that $u(t)$ is a control signal determined by the difference in the output of the chaotic neuron at different times with a delay τ and $\beta^{\kappa_c u(t)}$ is the perturbation term that multiplies the refractory scaling parameter α. When the output of a chaotic neuron is chaotic, the control signal $u(t)$ is not zero and $\alpha\beta^{\kappa_c u(t)}$ is less than α when $0<\beta<1$. Thus, the dynamics of the chaotic neuron will be changed [6].

We observe the dynamics of the networks by calculating a quasi-energy function defined in Eq. (12) [10]. The quasi-energy function is one type of observation functions for CNN, which is not a true energy function. Energy function is usually used to analyze the dynamics of the conventional associative network. However, there is no energy function for chaotic dynamics, and such analogy is not appropriate for the quasi-energy function of the network because the network behavior cannot be characterized by gradient descent dynamics of the function. But, it is useful to analyze the long-term temporal behavior of the neurons in a network as a whole with the scalar quasi-energy function [6].

$$QE(t)=-\frac{1}{2}\sum_i\sum_{j\neq i} w_{ij}x_i(t)x_j(t)-\sum_i a_i x_i(t). \tag{12}$$

The quasi-energy function denotes that the outputs of the controlled CNN are stable and periodic.

3. Improvement of the Controlled CNN

Studies show that one of the disadvantages of the controlled CNN is that the recall speed is slow. Additionally, another of the disadvantages of the CNN is that the output of the CNN cannot be stabilized to one of the stored patterns or a periodic orbit because of its chaotic behavior [6]. To increase the recall speed and decrease chaos of CNN, we propose the new function instead of the sigmoid function. A sigmoid function is necessary for creation a CNN. Sigmoid function is shown in Eq. (10). For the CNN, we proposed the following new function:

$$f_1(x)=\frac{1}{1+e^{-x^2/\varepsilon}} \tag{13}$$

According to our new function, the bifurcation diagram of $y(t)$ for different values of α is shown in Fig. 4. Additionally, will prove new function contrast with sigmoid functionhave more speed convergence.

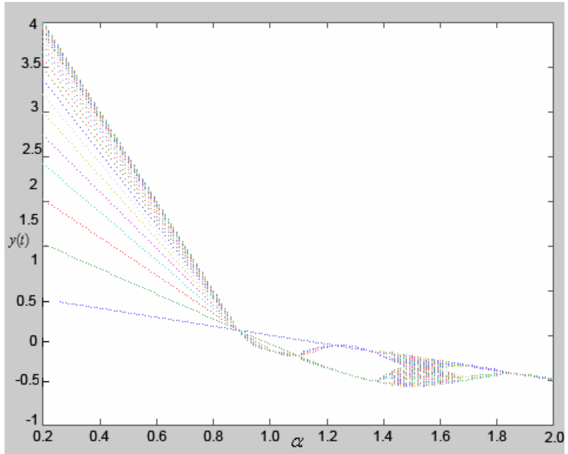

Fig. 4.The bifurcation diagram by using our new function.

- The first status ($x > 1$): $\frac{1}{1+e^{-x^2}} > \frac{1}{1+e^{-x}}$
- The second status($0 < x < 1$): $\frac{1}{1+e^{-x^2}} < \frac{1}{1+e^{-x}}$

As a result, it is seen that the behaviour of this function is like sigmoid function but the speed of its convergence is more than sigmoid function which causes to decrease of chaos. To employ this function in the CNN, it is important to study how chaos can be controlled. One of the disadvantages of the controlled CNN is that the recall speed is slow. Therefore mentioned control method for the CNN described in Eq. (8)-(11) will vary change. The core of this method is that using a new control signal instead of mentioned control signal for the CNN.A self-adaptable chaos control method is proposed as follows:

$$y_1(t+1) = ky(t) - \alpha u(t) f((y(t)) + a, \tag{14}$$

$$x(t+1) = f(y(t+1)), \tag{15}$$

$$f_1(x) = \frac{1}{1+e^{-x^2/\varepsilon}} \tag{16}$$

$$u_1(t) = e^{|x(t)-x(t-\tau)|} \tag{17}$$

The results of this comparison show the performance of CNN is improved greatly with the new method in chaos control. In the control method proposed in this paper, the control target needs not to be assigned and prior knowledge of the system is not required. This control technique is a type of adaptive control method. As a result, the outputs of the controlled CNN converge to the periodic orbits and are dependent on the initial patterns. Additionally, we will prove this new method contrast with mentioned method Have more speed convergence.

4. Experimental Results and Analysis

We use the conventional controlling method and the new method to realize associative memory. The values of the parameters for the both methods are:

$r = 0.2, \quad \beta = 1, \quad N = 100, \quad \alpha_{\min} = 2.8, \quad \alpha_{\max} = 3.7,$

$\gamma = 0.945 \;\; \tau = 3, \; \kappa_n = 0.6, \; \varepsilon = 0.001$

The initial value of α_i is between $\alpha_{\min}$ and $\alpha_{\max}$, and the value of α_i is updated after with every 2 time steps. According to the Fig. 3, the four binary patterns 100 bit for the both models stored.

To test the effectiveness of the proposed method, for each pattern we built 10 different patterns by adding 25% random noise. Therefore 10 training set contains 4×10 test patterns. One of the training set is shown in Fig. 5.

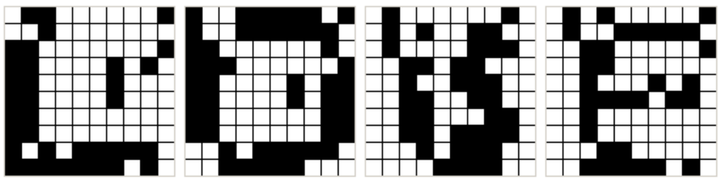

Fig. 5. A training set sample.

Both networks can retrieve the noisy patterns, correctly. The recall speed average of controlled CNN shown in Table 1, also the recall speed average of controlled CNN with new controlling parameter shown in Table 2.

Initial pattern	***L***	***O***	***V***	***E***
Average recall speed	20.1	24.3	25.6	**19.8**

Table 1:Control results for a controlled CNN

Initial pattern	***L***	***O***	***V***	***E***
Average recall speed	10.5	13.2	14	**10.1**

Table 2:Control results for a controlled CNN with new controlling parameters

It appears that the recall speed in new method increase to 50% in comparision to controlled CNN model.

Additionally, we will prove convergence in new method. For the system, we proposed the following energy function:

$$E = -\sum_{p=n}^{1}\sum_{s=p}^{1}\sum_{i}\sum_{j} w_{ij}\cdot(\chi_i^{(s)} + \chi_j^{(s)})^2 \tag{18}$$

In the following, we prove every change in $\underline{\chi}$ during the process will cause E to decreas this much:

$$-4\cdot\sum_{p=n}^{1}\sum_{s=p}^{1}\Delta\chi_r^{(s)}\sum_j w_{rj}\cdot\chi_j^{(s)}$$

Let's assume there would be a change in χ_r unit, so Eq. (5) should be:

$$E_1=-\sum_{p=n}^{1}\sum_{s=p}^{1}\left(\sum_{i\neq r}\sum_{j\neq r}w_{ij}\cdot(\chi_i^{(s)}+\chi_j^{(s)})^2+\sum_j w_{rj}\cdot(\chi_{r_1}^{(s)}+\chi_j^{(s)})^2+\sum_i w_{ir}\cdot(\chi_i^{(s)}+\chi_{r_1}^{(s)})^2\right)$$

$$E_2=-\sum_{p=n}^{1}\sum_{s=p}^{1}\left(\sum_{i\neq r}\sum_{j\neq r}w_{ij}\cdot(\chi_i^{(s)}+\chi_j^{(s)})^2+\sum_j w_{rj}\cdot(\chi_{r_2}^{(s)}+\chi_j^{(s)})^2+\sum_i w_{ir}\cdot(\chi_i^{(s)}+\chi_{r_2}^{(s)})^2\right)$$

The difference of Energy function is as follows:

$$E_2-E_1=$$

$$-\sum_{p=n}^{1}\sum_{s=p}^{1}\left(\sum_j w_{rj}\cdot\left[(\chi_{r_2}^{(s)}+\chi_j^{(s)})^2-(\chi_{r_1}^{(s)}+\chi_j^{(s)})^2\right]+\sum_i w_{ir}\cdot\left[(\chi_i^{(s)}+\chi_{r_2}^{(s)})^2-(\chi_i^{(s)}+\chi_{r_1}^{(s)})^2\right]\right)$$

$$=-\sum_{p=n}^{1}\sum_{s=p}^{1}\left(\sum_j w_{rj}\cdot\left[\chi^2{}_{r_2}^{(s)}+\chi^2{}_j^{(s)}+2\chi_{r_2}^{(s)}\chi_j^{(s)}-\chi^2{}_{r_1}^{(s)}-\chi^2{}_j^{(s)}-2\chi_{r_1}^{(s)}\chi_j^{(s)}\right]\right)+$$

$$-\sum_{p=n}^{1}\sum_{s=p}^{1}\left(\sum_j w_{ir}\cdot\left[\chi^2{}_{r_2}^{(s)}+\chi^2{}_i^{(s)}+2\chi_{r_2}^{(s)}\chi_i^{(s)}-\chi^2{}_{r_1}^{(s)}-\chi^2{}_i^{(s)}-2\chi_{r_1}^{(s)}\chi_i^{(s)}\right]\right)$$

$$=-\sum_{p=n}^{1}\sum_{s=p}^{1}\left(2\cdot\sum_j w_{rj}\cdot\chi_j^{(s)}\cdot\Delta\chi_r^{(s)}+2\cdot\sum_i w_{ir}\cdot\chi_i^{(s)}\cdot\Delta\chi_r^{(s)}\right)$$

Because of the symmetry of the net ($w_{ij}=w_{ji}$), so ΔE will be obtained as follows:

$$\Delta E=-4\cdot\sum_{p=n}^{1}\sum_{s=p}^{1}\Delta\chi_r^{(s)}\sum_j w_{rj}\cdot\chi_j^{(s)} \tag{19}$$

At first, we show that Eq. (18) is consistent when just one pattern is stored in memory: $\sum_j\chi_i\cdot w_{rj}$ is the weighted sum of the enterd inputs to the r th node.

- The first status ($\chi_{r_1}=+1$ and $\chi_{r_2}=-1$): for χ_r is changed from $+1$ in to -1 , therefore $\sum_j\chi_i\cdot w_{rj}<0$ and $\Delta\chi_r=-1-1=-2<0$ thus $\Delta E<0$.
- The second status ($\chi_{r_1}=-1$ and $\chi_{r_2}=+1$): for χ_r is changed from -1 in to $+1$, therefore $\sum_j\chi_i\cdot w_{rj}>0$ and $\Delta\chi_r=+1+1=+2>0$ thus $\Delta E<0$.

As a result, it is seen that with every change in χ_r, energy function will decrease. Now assuming that the issue is consistent for the time when $m-1$ patterns are stored in the system, we will verify that it is also consistent for m patterns.

$$-4\cdot\sum_{p=n}^{1}\sum_{s=p}^{1}\Delta\chi_r^{(s)}\sum_j w_{rj}\cdot\chi_j^{(s)} =$$

$$-4\cdot\left(\underbrace{\sum_{p=n-1}^{1}\sum_{s=p}^{1}\Delta\chi_r^{(s)}\sum_j w_{rj}\cdot\chi_j^{(s)}+\sum_{s=n-1}^{1}\Delta\chi_r^{(s)}\sum_j\chi_j^{(s)}\cdot w_{rj}}_{Supposed}+\underbrace{\Delta\chi_r^{(n)}\sum_j\chi_j^{(n)}\cdot w_{rj}}_{\text{Proved}}\right)$$

According to Induction, it is consistent for m patterns. So that ΔE is always negative or zero.

5. Conclusion

We have proposed a new function for the CNN and proved that it can increase the speed of convergence and decrease chaos. In order to apply CNN to pattern recognition, a new chaos control scheme was proposed. The improved controlled CNN have a superior ability to distinguish initial patterns comparing with mentioned controlled CNN in this paper. However, we have not discussed the control parameters in detail in this paper. How to choose the control parameters and the effects of control parameters on pattern recognition will be investigated in our next work.

References

[1] Adachi, M.; Aihara, K., &Kotani, M., “An analysis of associative memory dynamics with a chaotic neural network,” Proceedings of the International Symposium on Nonlinear Theory and its Applications, Pages **1169-1172**, 1993.

[2] Adachi, M.; Aihara, K., “Associative dynamics in a chaotic neural network,” Neural Networks, Pages **83 –98**, 1997.

[3] Afraimovich, V., & Hsu, S.B. (2003). Lectures on Chaotic Dynamical Systems: Studies in Advanced Mathematics. American Mathematical Society, International Press.

[4] C.A. Skarda, W.J. Freeman, “How brains make chaos in order to make sense of the world,” Behavior Brain Science, Volume 10, Pages **161-195**, 1987.

[5] Chen, L.; Aihara, K., “Chaotic simulated annealing by a neural network model with transient chaos,” Neural Networks, Volume 8, Issue 6, Pages **915–930**, 1995.

[6] He, J.; Chen, L.; Aihara, K., “Associative memory with a controlled chaotic neural network,” Neurocomputing, Volume 71, Pages **2494-2805**, 2008.

[7] Kuroiwa, J; Nara, s.; Aihara, k., “Response properties of a single chaotic neuron to stochastic inputs,” Int.J.Bifurcation Chaos 5, Pages **1447 –1460**, 2001.

[8] Kushibe, M.; Liu, Y.; Ohtsubo, J., “Associative memory with spatiotemporal chaos control,” Phys.Rev.E 53, Pages **4502 –4508**, 1996.

[9] Li, S.; Mou, X.; Cai, Y.; Ji Z., & J. Zhang, “On the security of a chaotic encryption scheme: problems with computerized chaos in finite computing precision,” Computer Physics, Pages **52-58**, 2003.

[10] Li, Y.; Zhu, P.; Chen, H.; Aihara, K.; He, J., “Controlling a chaotic in chaotic neural network for information processing,” Neurocomputing, Pages **111-120**, 2013.

[11] Mizutani, S.; Sano, T., “Controlling chaos in chaotic neural networks,” Electron.Commun.Jpn.Part 3 81, Pages **73 –8**2, 1998.

[12] Nakamura, K.; Nakagawa, M., “On the associative model with parameter controlled chaos neurons,” J.Phys.Soc.Jpn 62, Pages **2941 –2955**, 1993.

[13] Schuster, H.G., & Just, W.. Deterministic Chaos: An Introduction. Fourth edition, WILEY-VCH (2005).

[14] Sinha, S.; Munakata, T., & Ditto, W.L., “Chaos Computing: Implementation of Fundamental Logical Gates by Chaotic Elements,” IEEE TRANS, **11**, 2002.

[15] Sinha, S.; Murali, K., & Ditto, W.L., “Computing with chaos,” Physical Review. E, Statistical, Nonlinear, And Soft Matter Physics, 2004.

[16] W.J. Freeman, “Simulation of chaotic EEG patterns with a dynamic model of the olfactory system,” Biol.Cybern.56, Pages **139 –150**, 1987.

[17] Yong Yao and Walter J. Freeman, “Model of biological pattern recognition with spatially chaotic dynamics,” Neural Networks,Volume 3, Issue 2, Pages **153-170**, 1992.