PERGAMON

Pattern Recognition 34 (2001) 1093–1104

PATTERN RECOGNITION
THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Annealed chaotic neural network with nonlinear self-feedback and its application to clustering problem

Jzau-Sheng Lin*

Department of Electronic Engineering, National Chin-Yi Institute of Technology, Taichung, Taiwan, ROC

Received 6 May 1999; received in revised form 19 November 1999; accepted 1 February 2000

Abstract

Chaos is a revolutionary concept, which brings a novel strategy of science for researchers. In this paper, a chaotic neural network is proposed and the simulated annealing strategy also embedded to construct an annealed chaotic neural network (ACNN) and apply to the clustering problem. In addition to retain the characteristics of the conventional neural units, the ACNN displays a rich range of behavior reminiscent of that observed in neurons. Unlike the conventional neural network, the ACNN has rich range and flexible dynamics, so that it can be expected to have higher ability of searching for globally optimal or near-optimum results. However, the chaotic neural network does not stay in the global solution due to the chaotic dynamical mechanism being not clear. A chaotic mechanism with annealing strategy is introduced into the Hopfield network to construct a ACNN for expecting a better opportunity of converging to the optimal solution in this paper. In experimental results, unlike the fuzzy clustering methods getting local minima solutions, the ACNN method can always obtain the near-global optimal results. From the classification of real multispectral images, the ACNN can obtain suitable results.

Keywords: Chaotic neural network; Hopfield network; Self-feedback; Clustering problem; Annealing

1. Introduction

Clustering algorithms attempt to organize the training patterns into clusters such that patterns within a cluster are more similar to each other than those belonging to different clusters. In many fields such as segmentation, pattern recognition, and vector quantization, clustering process is an indispensable step in these problems. Since the range of clustering application is so large, there is no fundamental clustering problem formulation due to variant relationship between the input objects. There are several algorithms based upon the least-mean-squares criterion for clustering problem such as traditional strategies, fuzzy clustering methods, and probability clustering algorithms. Generally speaking, conventional methods such as *K*-means (*C*-means) [1] and ISODATA [2] are traditional clustering methods in which a sample belonging only one cluster. FCM [3–6], PFCM [7,8] and CFCM [9] are called fuzzy clustering methods in which every sample belonging all clusters with different degrees of membership while the probability clustering algorithms such as simulated annealing [10–13] in which every sample belonging all clusters with different degrees of probability.

The Hopfield network model, proposed by Hopfield and Tank [14,15], has been extensively applied to many fields [16–19] for optimization problem in the last decade. In the application of optimization problem, this network exploits the massive parallelism of neurons. However, due to it is a stable system with gradient descent mechanism, the results obtained are not satisfied [20]. The Hopfield model has no scheme to escape the local minima and may be trapped on one of them.

The chaotic dynamic in chaotic neural networks has been discussed in the past studies [20–25] in the light of

* Tel.: + 886-4-3924505 ext 7311; fax: + 886-4-3926610.
E-mail address: jslin@chinyi.ncit.edu.tw (J.-S. Lin).

0031-3203/01/$20.00
PII: S0031-3203(00)00040-6

its potential biological functional role. A chaotic neural network proposed by Aihara et al. [23] effectively searches the global minimum using the chaotic searching mechanism without stacking in undesirable local minima. However, the network does not stay in the global solution due to the chaotic dynamical mechanism being not clear.

The chaotic mechanism with annealing strategy is introduced into the Hopfield network to construct an annealed chaotic neural network (ACNN) for expecting a better opportunity of converging to the optimal solution in this paper. In the optimization problem, most of systems implement the straightly forward algorithm that easily traps into a local minimum. In order to resolve this problem, Kirkpatrick et al. [26] used the Monte Carlo technique to propose the simulated annealing algorithm in 1983. In 1953 simulated annealing strategy was first proposed by Metropolis et al. [27] to simulate molecular processes. The simulated annealing technique had nonzero probability to go from one state to another, moves temporarily toward a worse state so as to escape from local traps. The probability function depends on the temperature and the energy difference between the two states. With the probabilistic hill-climbing search approach, the simulated annealing technique has a better probability to go to a higher energy state at a higher temperature. In the simulated annealing algorithm, the annealing mechanism is also named stochastic simulated annealing due to the optimal cooling schedules being processed with stochastic scheme. Geman et al. [28] have proven that the energy function can be converged to the global minimum only if the cooling schedule is decreased slowly and inverse proportion to the logarithm of iteration number. In 1994 Rosen and Nakano [29] proposed a very fast annealing algorithm which could improve the performance of the original stochastic simulated annealing.

In this paper, the Hopfield-model net embedded the chaotic mechanism with annealing strategy called ACNN is proposed so that the parallel implementation to find optimal solution for clustering problem is feasible. The cluster problem can be cast as an optimal problem that may also be regarded as a minimization of a criterion defined as a function of the least-squares Euclidean distance between training pattern and cluster center. The ACNN is trained to classify the input patterns into feasible cluster when the defined energy function converges to globally optimal or near-global optimal solution. The training patterns can be mapped to a two-dimensional Hopfield neural network. In the ACNN, the columns represent the number of clusters and the rows represent the training patterns. In the context of clustering problem, a training pattern can be as the input of all neurons at a row in the two-dimensional net that is fully connected structure. After a number of iterations and bifurcation of chaotic dynamics, the neuron states are refined to reach near optimal result when the defined energy function is converged. However, a training pattern does not necessarily belong to only one cluster. Instead, a certain probability grade belonging to proper class is associated with every pattern. In addition to the annealing strategy, the chaotic behavior in the chaotic network is controlled to escape from local minima results. Consequently, the energy function can be converged into a global or near global minimum to produce satisfactory clusters. Compared with conventional and fuzzy cluster techniques, the major strength of the presented ACNN is computationally more promising due to the chaotic characteristics. In a simulated study, the ACNN is described to have the capability for clustering problem and shown the globally optimal results.

The rest of this paper is organized as follows. Section 2 reviews the chaotic neural networks; Section 3 proposes the annealing strategies; Section 4 demonstrated the chaotic simulated annealing neural network for clustering problem; Section 5 presents several experimental results; and finally, Section 6 gives the discussion and conclusions.

2. Chaotic neural network

The conventional neural units can not generate the dynamics of great complexity occupied by the biological neurons. A chaotic neural unit has the nonlinear recursive equation that features to display the rich range of behavior reminiscent. In other words, the chaotic neural network has rich dynamics with various coexisting attractors, not only of fixed points but also periodic and even chaotic attractors, it can escape from local minima and converging to the global-minimum or near-global-minimum result. The study of chaotic neural network is important not only as a model for nonlinear systems with many degrees of freedom, but also from the viewpoint of biological information processing and possible engineering application [30]. Several chaotic neural networks have been proposed in the past [30–36]. Although the chaotic neural network is a promising technique for optimization problem, the converging process has not been satisfactorily solved in relation to chaotic dynamics. It is difficult to decide how to control the chaotic behavior in a chaotic neuron for converging to a stable equilibrium point corresponding to an acceptably near-optimal solution. To harness the chaotic dynamics, Chen and Aihara [35] proposed a chaotic simulated annealing (CSA) for combinatorial optimization problem. Different from the conventional stochastic simulated annealing, the chaotic simulated annealing is a deterministic optimizer that converges from chaotic state at high temperature through successive bifurcations during temperature reducing process to an equilibrium point at low temperature. In this paper a 2-D chaotic neural network

embedded with the Feigenbaum's bifurcation formula [37] and self-feedback connection weight, which has an ability of parallel synchronous computation in the bifurcation states. The model of the chaotic neural network can be demonstrated as follows:

$$u_{x,i}(k) = \frac{1}{1 + e^{-v_{x,i}(k)/\lambda}}, \tag{1}$$

$$v_{x,i}(k+1) = \delta \sin[\pi v_{x,i}(k)] + \sum_{y,j} w_{x,i;y,j} u_{y,j}(k) + I_{x,i} - T_{x,i}(k) u_{x,i}(k), \tag{2}$$

where $D_{x,i}(k) = \sum_{y,j} w_{x,i;y,j} u_{y,j}(k) + I_{x,i}$, $u_{y,j}$ is the output of neuron (x,i), $v_{x,i}$ the internal state of neuron (x,i), $w_{x,i;y,j}$ the connection weight from neuron (y,j) to (x,i), $I_{x,i}$ the input bias of neuron (x,i), δ the damping factor of nerve membrane $(0 \leqslant \delta \leqslant 1)$, $T_{x,i}(k)$ the self-feedback connection weight, λ the steepness parameter of the output function ($\lambda > 0$), and π the ratio of the circumference of a circle to its diameter.

This structure can be directly mapped into the Hopfield neural network. The chaotic dynamics in an output state with distinct self-feedback connection weights for this chaotic neural model are shown in Fig. 1. The values of parameters are set as $\delta = 0.3$, $D = 0$ and $\lambda = 1/250$ during 4000 iterations for all pictures in Fig. 1. For a self-feedback connection weight in an interconnection strength $T = 0.08$ chaotic activity is generated, while $T < 0.08$ the neuron state $u(t)$ gradually transient from chaotic characteristics through periodic bifurcation to an equilibrium point ($T = 0.0399$). From Fig. 1 we can find that different values for the self-feedback connection weight may result in variant output states. This phenomenon indicates that the chaotic behavior is not easily harnessed in the chaotic neural network.

3. Annealed chaotic neural networks

Although the chaotic neural network is a promising technique for optimization problem, the converging process has not been satisfactorily solved in relation to chaotic dynamics. It is difficult to decide how to control the chaotic behavior in a chaotic neuron for converging to a stable equilibrium point corresponding to an acceptably near-optimal solution. To harness the chaotic dynamics, Chen and Aihara [35] proposed a transient chaotic neural network (TCNN) embedded a simulated annealing strategy for combinatorial optimization problem. Different from the conventional stochastic simulated annealing, the chaotic simulated annealing is a deterministic optimizer that converges from chaotic state at high temperature through successive bifurcations during temperature reducing process to an equilibrium point at low temperature.

In order to control the chaotic dynamical mechanism, the simulated annealing strategy is also embedded into the proposed chaotic neural network in the last section for expecting a better opportunity of converging to the optimal solution. In the simulated annealing process, a feasible cooling schedule is required. The reaching thermal equilibrium at low temperature might take a very long time. Geman et al. [28] demonstrated that if the temperature is lowered at the rate:

$$T(k) = \frac{T(0)}{\log(k+1)}, \tag{3}$$

where $T(0)$ is a constant and k is the number of iterations, the algorithm will converge to the set of states of least energy. Jalali et al. [12] presented that the value of the constant $T(0)$ for which Geman et al. were able to guarantee convergence is in general very high, so that the convergence time becomes impractically slow. Jalali et al. used a schedule very similar to that of Geman et al., given in Eq. (8), but with a steeper descent at higher iterations as follows:

$$T(k) = \frac{T(0)}{\log(k+1)^3}. \tag{4}$$

Jalali et al. showed that the value of $T(0)$ in Eq. (4) has to be kept as small as possible, so that the number of iterations can be held within a reasonable limit. Unfortunately, the cooling schedules specified by Eqs. (3) and (4) with high value of $T(0)$ are too slow to be of practical use [38]. Kirkpatrick et al. [26] proposed a cooling schedule specified a finite sequence of values of the temperature and a finite number of transitions attempted at each value of the temperature. The decrement function of cooling schedule is defined by

$$T(k) = (\alpha)^k T(0), \quad k = 1, 2, \ldots, \tag{5}$$

where α $(0.8 \leqslant \alpha < 1)$, is a constant smaller but close to unit. The author [39] proposed another decrement function of cooling schedule as follows:

$$T(k) = \frac{1}{\beta + 1} [\beta + \tanh(\alpha)^k] T(k-1), \quad k = 1, 2, \ldots, \tag{6}$$

where α is a constant same as the one in Eq. (5). And β is also a constant. Eq. (6) can result in a faster decrement speed than those resulted from Eq. (5). Fig. 2 shows the reduction process using different decrement functions described from Eqs. (3)–(6) with $\alpha = 0.98$, initial constant $T(0) = 4000$, and 60 iterations. From Fig. 2, we can find Eq. (6) results in the fastest decrement speed.

In this paper, the simulated annealing strategy was embedded into Eq. (2) to construct the ACNN. In the ACNN, the self-feedback connection weight was decreased with the decrement function proposed by the

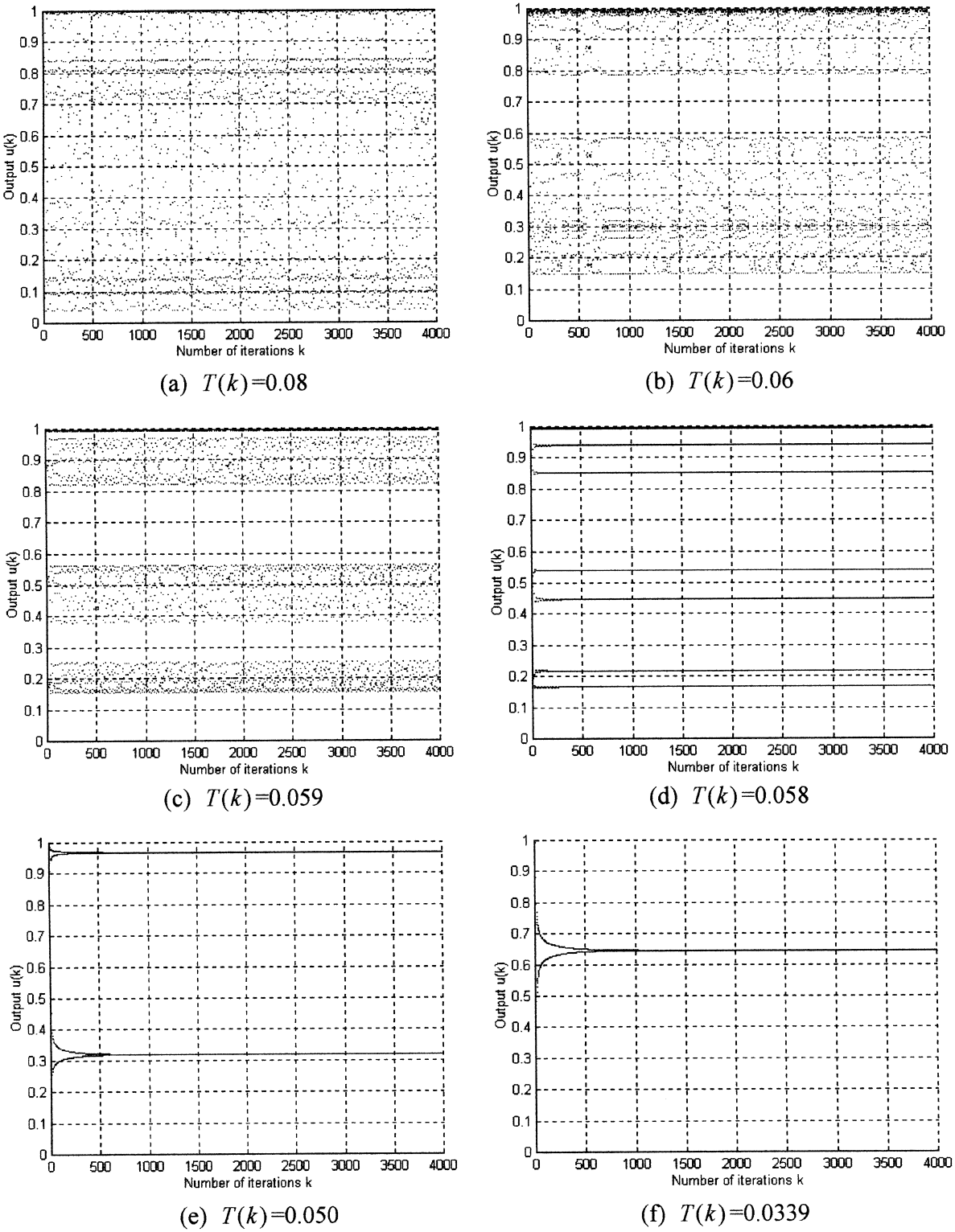


Fig. 1. The bifurcation states of a neuron with the variant self-feedback connection weights $T(k)$ during 4000 iterations: (a) $T(k) = 0.08$; (b) $T(k) = 0.06$; (c) $T(k) = 0.059$; (d) $T(k) = 0.058$; (e) $T(k) = 0.050$; and (f) $T(k) = 0.0339$.

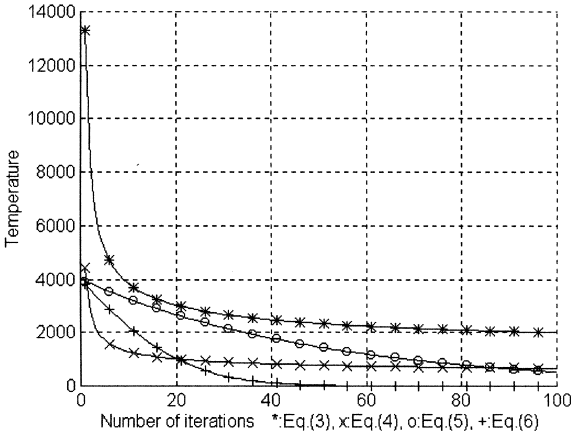


Fig. 2. The reduction process using different decrement functions described from Eqs. (3)–(6) with $T(0) = 4000$ within 100 iterations.

author [39] shown in Eq. (6). The ACNN is defined as follows:

$$u_{x,i}(k) = \frac{1}{1 + e^{-v_{x,i}(k)/\lambda}}, \tag{7}$$

$$v_{x,i}(k+1) = \delta \sin[\pi v_{x,i}(k)] + D_{x,i} - T_{x,i}(k)[u_{x,i}(k) - I_0] \tag{8}$$

and

$$T_{x,i}(k) = \frac{1}{\beta + 1}\,[\beta + \tanh(\alpha)^k] T_{x,i}(k-1), \quad k = 1, 2, \ldots, \tag{9}$$

where $D_{x,i}(k) = \sum_{y,j} w_{x,i;y,j} u_{y,j}(k) + I_{x,i} u_{x,i}$ is the output of neuron (x,i), $v_{x,i}$ the internal state of neuron (x,i), $w_{x,i}$ the connection weight from neuron (y,j) to (x,i), $I_{x,i}$ the input bias of neuron (x,i), δ the damping factor of nerve membrane $(0 \leqslant \delta \leqslant 1)$, $T_{x,i}(k)$ the self-feedback connection weight, λ the steepness parameter of the output function of neuron (x,i) $(\lambda > 0)$, α the positive constant and $(0.8 \leqslant \alpha < 1)$, β the positive constant, and π the ratio of the circumference of a circle to its diameter.

And a single neuron model in the ACNN can also be expressed as

$$u(k) = \frac{1}{1 + e^{-v(k)/\lambda}}, \tag{10}$$

$$v(k+1) = \delta \sin[\pi v(k)] + D - T(k)\,[u(k) - I_0], \tag{11}$$

$$T(k) = \frac{1}{\beta + 1}\,[\beta + \tanh(\alpha)^k] T(k-1), \quad k = 1, 2, \ldots. \tag{12}$$

One can study the dynamics regimes of the model in detail. Since the purpose in this paper is mainly to show the potential application of the model to optimization problems of various scales. To study the chaotic dynamics of a single neuron model in the ACNN, the value of parameters are fixed as $\delta = 0.3$; $\lambda = 1/250$; $I_0 = 0.65$; $T(0) = 0.08$ just vary α, β, and D. The dynamics regimes are illustrated by bifurcation diagrams for output state with respect to self-feedback connection weight in a single neuron. Figs. 3(a) and (b) show the time evolutions of the output of a single neuron and the self-feedback connection weight when $\alpha = 0.998$, $\beta = 500$, $D = 0$ and $\alpha = 0.9998$, $\beta = 500$, $D = 0$, respectively. Fixed points, periodic orbits and complex oscillation can be detected in these figures. The chaotic dynamics also disappears quickly due to the self-feedback connection weight decreasing rapidly with a small value of α that is shown in Fig. 3(a). On the other hand, in Fig. 3(b), the chaotic dynamics of in turn lasts longer owing to a larger value of α. Therefore, the constant α can govern the bifurcation speed of the chaotic neuron.

4. Application to the clustering problem

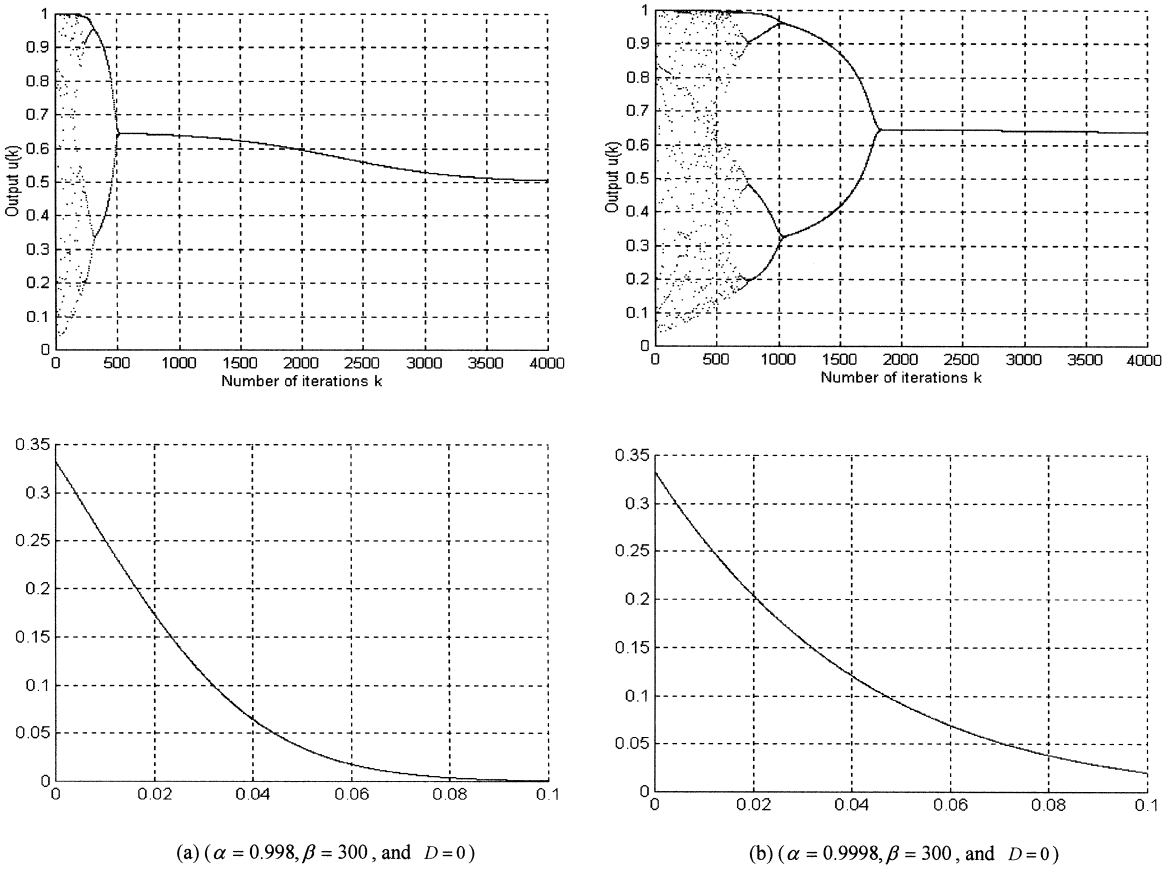
The ACNN used the two-dimensional Hopfield neural network architecture with chaotic simulated annealing to classify the training patterns to generate feasible clusters in clustering problem. In order to increase the capability of the proposed approach, the energy function is formulated on the basis of within-class scatter matrix, a concept widely used in pattern classification. Where, the within-class scatter matrix is defined by the average Euclidean distance between training pattern and cluster center within the same cluster. Let $u_{x,i}$ be the probability state of the (x,i)th neuron and $w_{x,i;y,i}$ present the interconnected weight between neuron (x,i) and neuron (y,i) in a two-dimensional neuron array. A neuron (x,i) in the network receives weighted inputs $w_{x,i;y,i}$ from each neuron (y,i) and a bias $I_{x,i}$ from output. The total input to neuron (x,i) is computed as

$$Net_{x,i} = |z_x - \sum_{y=1}^{n} w_{x,i;y,i} u_{y,i}|^2 + I_{x,i}. \tag{13}$$

The modified Lyapunov energy function of the two-dimensional Hopfield neural network is given by

$$E = \frac{1}{2} \sum_{x=1}^{n} \sum_{i=1}^{c} u_{x,i} \left| z_x - \sum_{y=1}^{n} w_{x,i;y,i} u_{y,i} \right|^2 + \frac{1}{2} \sum_{x=1}^{n} \sum_{i=1}^{c} I_{x,i} u_{x,i}, \tag{14}$$

where $|\cdot|$ is the average Euclidean distance between training patterns to cluster center, $\sum_{y=1}^{n} w_{x,i;y,i}$ is the total



(a) ($\alpha = 0.998, \beta = 300$, and $D = 0$) (b) ($\alpha = 0.9998, \beta = 300$, and $D = 0$)

Fig. 3. Time evolutions of single neuron model with variant values α in the ACNN: (a) $\alpha = 0.998$, $\beta = 500$, and $D = 0$; (b) $\alpha = 0.9998$, $\beta = 500$, and $D = 0$.

weighted input received from neuron (y, i) in row i, $u_{x,i}$ is the analog output state at neuron (x, i), respectively. Each column of this modified Hopfield network represents a class and each row represents the training pattern. The network reaches a stable state when the modified Lyapunov energy function is minimized. For example, a neuron (x, i) in a maximum probability state indicates that training pattern z_x belongs to class i.

The objective function, used to generate a suitable clustering that has a minimum average distance between training pattern and the cluster centroid within class, is given by

$$E = \frac{A}{2} \sum_{x=1}^{n} \sum_{i=1}^{c} u_{x,i} \left| z_x - \sum_{y=1}^{n} \frac{1}{\sum_{h=1}^{n} V_{h,i}} z_y u_{y,i} \right|^2 + \frac{B}{2} \sum_{x=1}^{n} \sum_{i=1}^{c} \sum_{j=1}^{c} u_{x,i} u_{x,j} + \frac{C}{2} \left[\left(\sum_{x=1}^{n} \sum_{i=1}^{c} u_{x,i} \right) - n \right]^2. \quad (15)$$

In Eq. (15), each state $u_{x,i}$ is looked upon as the probability of finding training sample z_x currently occupied by class i undergo random thermal perturbations. As each training sample can be occupied by some classes with different probabilities. To simplify the updating process of neuron states, the sigmoid function displayed in Eq. (7) was replaced by the normalized manner shown in Eq. (19). That is, every raw can have at most 1. In other words, the summation of states in the same row equals 1. It also ensures that only n vectors will be classified into these c clusters. That is the network must match the following constraints:

$$\sum_{i=1}^{c} u_{x,i} = 1$$

and

$$\sum_{x=1}^{n} \sum_{i=1}^{c} u_{x,i} = n.$$

Therefore, using the normalized method the objective function of the ACNN can be further simplified as

$$E = \frac{1}{2}\sum_{x=1}^{n}\sum_{i=1}^{c} u_{x,i}\left|z_x - \sum_{y=1}^{n}\frac{1}{\sum_{h=1}^{n} V_{h,i}} z_y u_{y,i}\right|^2. \quad (16)$$

Comparing Eq. (16) with the modified cost function Eq. (14), the synaptic interconnection weights and the bias input can be obtained as

$$w_{x,i;y,i} = \frac{1}{\sum_{h=1}^{n} u_{h,i}} z_y, \quad (17)$$

and

$$I_{x,i} = 0. \quad (18)$$

In order to control the dynamic mechanism, the simulated annealing strategy is embedded into this network. For the purpose of improving the computation performance in the 2–D neuron array using the ACNN for clustering problem, the probability grade of neural state of the training sample z_x occupied by class i can then be modified from Eq. (5) and normalized as follows:

$$u_{x,i}(k) = \frac{e^{-v_{x,i}(k)/\lambda}}{\sum_{j=1}^{c} e^{-v_{x,j}(k)/\lambda}}. \quad (19)$$

And the ACNN can further be defined as follows for the clustering problem:

1. Input a set of training samples $\mathbf{Z} = \{z_1, z_2, \ldots, z_n\}$, the number of cluster c, and randomly set the internal states for all neurons.
2. Start with an initial temperature $T(0)$ and randomly initialize the probabilities for all neurons.
3. Calculate the new internal states for all neurons using Eq. (8).
4. Change the output for all neurons using Eq. (19).
5. Decrease T with the annealing factor $T(k)$ shown in Eq. (9) iteratively.
6. Repeat steps 3–5 until the energy function is convergent.

5. Experimental results

The Butterfly example given by Ruspini [40] and Jou [41] is considered to see the performance of the fuzzy methods FCM, PFCM, CFHNN, the proposed algorithm ACNN, and the TCNN proposed by Chen and Aihara [35]. Tables 1 and 2 list 15 input patterns in R^2. Patterns 7–9 construct a bridge between the wings of the butterfly. The membership grades of different fuzzy

Table 1
The training patterns of Butterfly and membership grades with $c = 2$ after convergence in different fuzzy c-means algorithms

Patterns		$m = 1.25$						$m = 2.0$					
		FCM		PFCM		CFHNN		FCM		PFCM		CFHNN	
x	z_x	μ_1	μ_2	μ_1	μ_2	μ_1	μ_2	μ_1	μ_2	μ_1	μ_2	μ_1	μ_2
1.	(0, 0)	0.000	1.000	0.000	1.000	0.000	1.000	0.022	0.978	0.034	0.966	0.029	0.971
2.	(0, 2)	0.000	1.000	0.000	1.000	0.000	1.000	0.001	0.999	0.006	0.996	0.003	0.997
3.	(0, 4)	0.000	1.000	0.000	1.000	0.000	1.000	0.022	0.978	0.034	0.966	0.029	0.971
4.	(1, 1)	0.000	1.000	0.000	1.000	0.000	1.000	0.003	0.997	0.016	0.984	0.010	0.990
5.	(1, 2)	0.000	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.007	0.993	0.003	0.997
6.	(1, 3)	0.000	1.000	0.000	1.000	0.000	1.000	0.003	0.997	0.016	0.984	0.010	0.990
7.	(2, 2)	0.000	1.000	0.000	1.000	0.000	1.000	0.020	0.980	0.060	0.940	0.043	0.957
8.	(3, 2)	0.792	0.208	0.738	0.262	0.555	0.445	0.502	0.498	0.498	0.502	0.498	0.502
9.	(4, 2)	1.000	0.000	1.000	0.000	1.000	0.000	0.981	0.019	0.939	0.061	0.956	0.044
10.	(5, 1)	1.000	0.000	1.000	0.000	1.000	0.000	0.997	0.003	0.984	0.016	0.990	0.010
11.	(5, 2)	1.000	0.000	1.000	0.000	1.000	0.000	1.000	0.000	0.993	0.007	0.997	0.003
12.	(5, 3)	1.000	0.000	1.000	0.000	1.000	0.000	0.997	0.003	0.984	0.016	0.990	0.010
13.	(6, 0)	1.000	0.000	1.000	0.000	1.000	0.000	0.978	0.022	0.966	0.034	0.971	0.029
14.	(6, 2)	1.000	0.000	1.000	0.000	1.000	0.000	0.999	0.001	0.994	0.006	0.997	0.003
15.	(6, 4)	1.000	0.000	1.000	0.000	1.000	0.000	0.978	0.022	0.966	0.034	0.971	0.029
Cluster Centers		(0.757, 0.000) (5.063, 2.000)		(0.920, 2.000) (5.228, 2.000)		(0.825, 2.000) (5.137, 2.000)		(0.799, 2.000) (5.206, 2.000)		(0.798, 2.000) (5.206, 2.000)		(0.792, 2.000) (5.204, 2.000)	

Table 2
The training patterns of Butterfly and membership/probability grades with $c = 2$ after convergence in ACNN, TCNN and the optimal clustering result

Patterns		Optimal clustering		ACNN		TCNN	
x	z_x	$\tilde{\mu}_1$	$\tilde{\mu}_2$	μ_1	μ_2	μ_1	μ_2
1.	(0, 0)	0.000	1.000	0.000	1.000	0.000	1.000
2.	(0, 2)	0.000	1.000	0.000	1.000	0.000	1.000
3.	(0, 4)	0.000	1.000	0.000	1.000	0.000	1.000
4.	(1, 1)	0.000	1.000	0.000	1.000	0.000	1.000
5.	(1, 2)	0.000	1.000	0.000	1.000	0.000	1.000
6.	(1, 3)	0.000	1.000	0.000	1.000	0.000	1.000
7.	(2, 2)	0.000	1.000	0.002	0.998	0.001	0.999
8.	(3, 2)	0.500	0.500	0.499	0.501	0.466	0.534
9.	(4, 2)	1.000	0.000	0.998	0.002	0.999	0.001
10.	(5, 1)	1.000	0.000	1.000	0.000	1.000	0.000
11.	(5, 2)	1.000	0.000	1.000	0.000	1.000	0.000
12.	(5, 3)	1.000	0.000	1.000	0.000	1.000	0.000
13.	(6, 0)	1.000	0.000	1.000	0.000	1.000	0.000
14.	(6, 2)	1.000	0.000	1.000	0.000	1.000	0.000
15.	(6, 4)	1.000	0.000	1.000	0.000	1.000	0.000
Cluster Centers		(0.867, 2.000)		(0.867, 2.000)		(0.876, 2.000)	
		(5.133, 2.000)		(5.133, 2.000)		(5.144, 2.000)	

Table 3
The average distortions between the membership/probability grades in the optimal clustering results and those in the variant methods

	$m = 1.25$			$m = 2.0$			ACNN	TCNN
	FCM	PFCM	CFHNN	FCM	PFCM	CFHNN		
Average distortion A.d	5.7×10^{-3}	3.8×10^{-3}	2.0×10^{-4}	1.8×10^{-4}	8.76×10^{-4}	5.06×10^{-4}	6×10^{-7}	7.7×10^{-5}

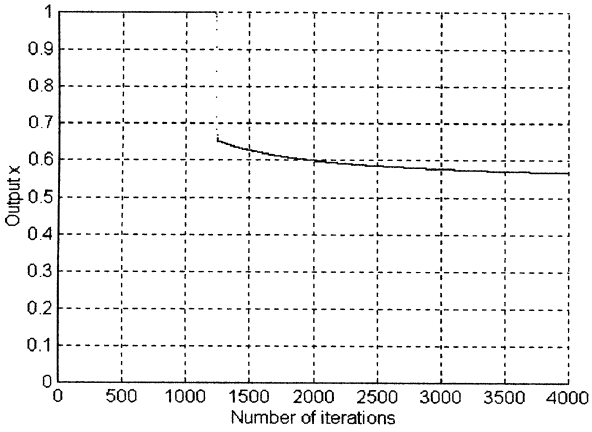
algorithms with fuzzy parameter $m = 1.25$ and 2.0 for all fuzzy methods in this paper are listed in Table 1, respectively. The optimal clustering grades, and probability grades for the ACNN and TCNN are shown in Table 2. In Table 1, the grades are nearly symmetric with respect to pattern z_8 in both data coordinate directions for all fuzzy algorithms. The lager m, the fuzzier becomes the membership grades of the final partition. For all fuzzy algorithms, all of the energy functions trap into local minima within several iterations. The TCNN can just converge to near-global minima in all experiments. But the proposed approach can almost converge to the global minimum and get two unique cluster centers (0.867, 2.000) and (5.133, 2.000) after several iterations for all experiments. In addition, it is fairly expected that the membership grade of pattern z_8 with respect to both clusters should be close to 0.5 with a smaller fuzzy parameter for all fuzzy algorithms and the probability grade of pattern z_8 with respect to both clusters also close to 0.5 for the TCNN. But the probability grade of pattern z_8 with respect to both clusters almost equals to 0.5 in the proposed algorithm ACNN.

To observe the performance for all methods, the average distortions between the membership/probability grades in the optimal clustering result ant those in different approaches are calculated and listed in Table 3. The average distortion is defined as follows:

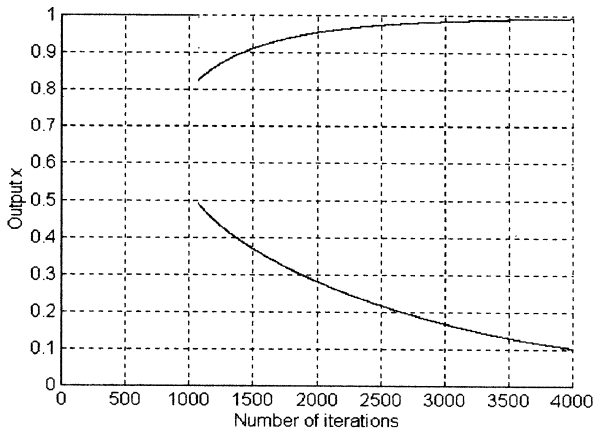
$$a.d = \frac{1}{n \times c} \sum_{x=1}^{n} \sum_{i}^{c} (u_{x,i} - \tilde{u}_{x,i})^2, \tag{20}$$

where $a.d$ is the average distortion, n the number of training samples, c the number of clusters, $u_{x,i}$ the membership/probability grade at the neuron (x, i) for proper method in this paper, and $\tilde{u}_{x,i}$ the membership/probability grade at the neuron (x, i) in the optimal clustering result.

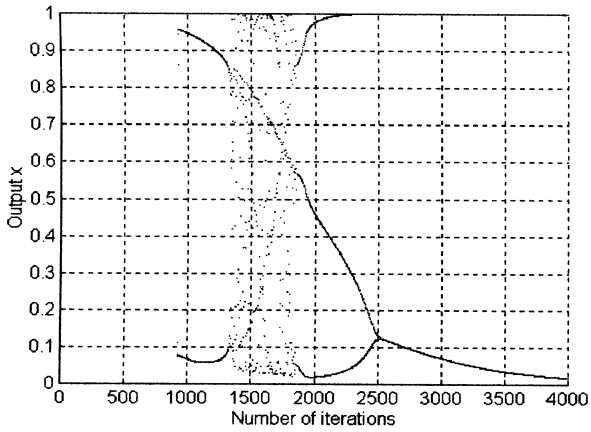
From Table 3, we can see that the proposed ACNN algorithm can obtain the minimum average distortion.



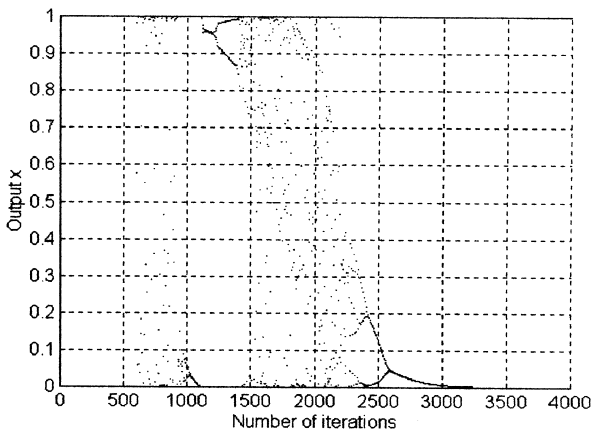
(a) $T(k)$ =0.01



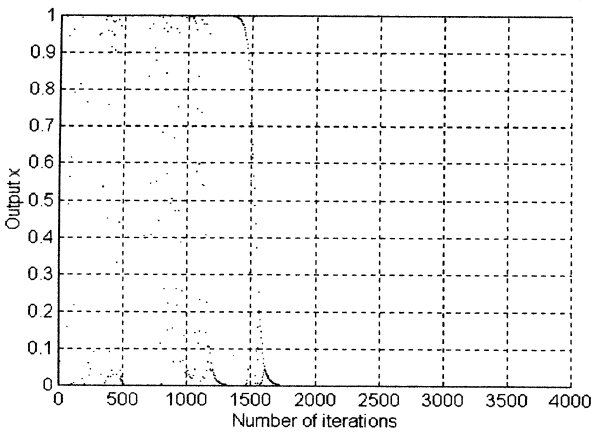
(b) $T(k)$ =0.04



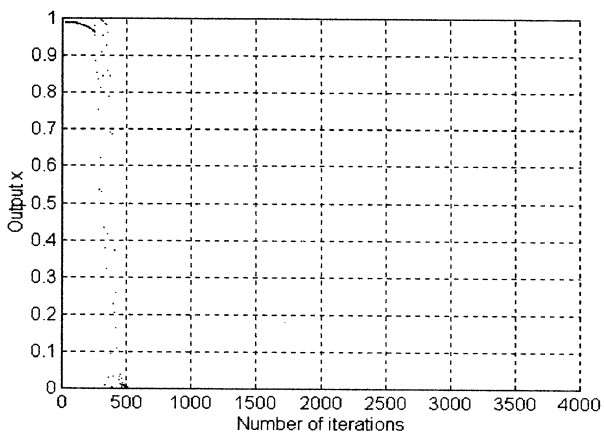
(c) $T(k)$ = 0.08



(d) $T(k)$ = 0.2



(e) $T(k)$ = 0.8

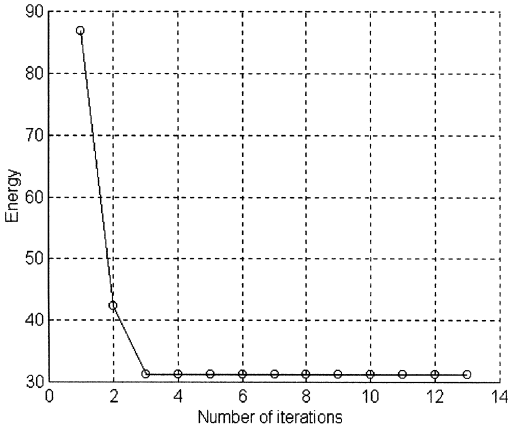


(f) $T(k)$ = 1.4

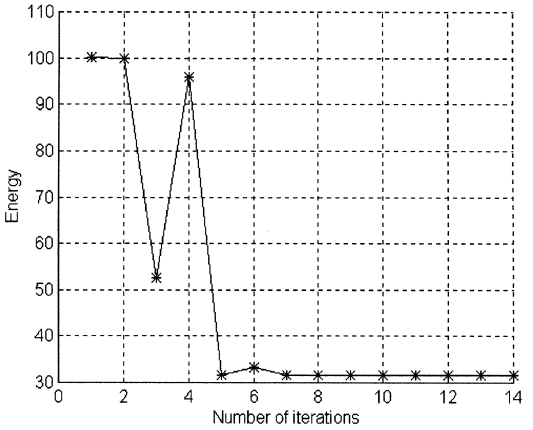
Fig. 4. Chaotic dynamics of single neuron with different fixed $T(k)$ within $0.08 \leqslant \delta \leqslant 0.6$.

In another viewpoint, the chaotic dynamics could be influenced by different parameters. From simulation results, the neuron states display the bifurcation phenomenon when $0.01 \leqslant T(k) \leqslant 1.6$ and $0 \leqslant \delta \leqslant 1$. The chaotic characteristics are shown in Fig. 4 within 4000 iterations in which δ was changed with $\delta(k+1) = 0.995\delta(k)$ and $\delta(0) = 0.6$. In Fig. 3, the output state converges to an equilibrium point with the same bifurcation manner controlled by the proposed cooling schedule for the self-feedback connection weight. Instead of controlling $T(k)$, the bifurcation processes are exhibited various manners with fixed $T(k)$ using a cooling schedule to decrease δ. Therefore, harness the chaotic dynamics with δ is not suitable for the chaotic neural network.

In Fig. 5, we can see that the energy functions in the fuzzy algorithms will either decrease or retain its energy as a result of each individual update as these approaches undergoing transitions with a gradient decreasing manner. Therefore, they could easily trap into a local minimum. But, the energy functions in the ACNN can escape from a local minimum to the global or near-global minimum with the bifurcation characteristics in the chaotic dynamics controlled by the simulated annealing strategy.



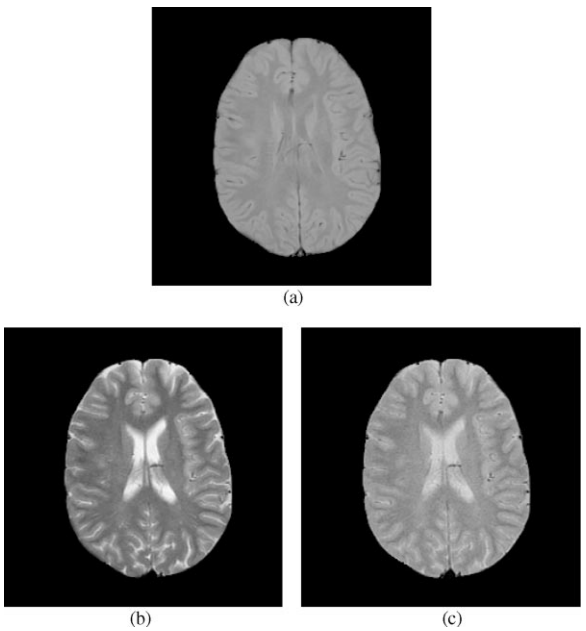
(a) convergence curve of the energy for different fuzzy c-means algorithms



(b) convergence curve of the energy for the annealed chaotic neural networks.

Fig. 5. The convergence curve of the energies for different algorithms: (a) convergence curve of the energy for different fuzzy c-means algorithms; (b) convergence curve of the energy for the annealed chaotic neural networks.

To show the performance of ACNN, multispectral brain image shown in Fig. 6, acquired from a patient with normal physiology using T_2-weighted sequences, is used for simulation. The multispectal image consists three-channel images which are formed as 256×256 pixels and 8-bit gray levels. The acquisition parameters with different repetition time (TR) and echo time (TE) are $TR_1/TE_1 = 1500$ ms/57 ms, $TR_2/TE_2 = 1500$ ms/100 ms, and $TR_3/TE_3 = 1500$ ms/75 ms. The cerebral spinal fluid (CSF) appears brighter than gray and white matters in T_2-weighted image. And the T_2-weighted image also shows the gray matter (GM) slightly brighter compared with white matter (WM). The number of cluster was preset 4. The regions indicated background, GM, WM, and CSF are classified and shown in Fig. 7 for $TR_2/TE_2 = 1500$ ms/100 ms image. The simulation was executed in a Pentium-II-200 personal computer with C++ language. The classified results were obtained after about 12 iterations and cost 35 s. Each iteration is defined that all pixels in the three-channel images are



(a) (b) (c)

Fig. 6. Multispectral brain images for simulation: (a) $TR_1/TE_1 = 1500$ ms/57 ms; (b) TR_2/TE_2 2500 ms/100 ms; (c) $TR_3/TE_3 = 2500$ ms/75 ms.

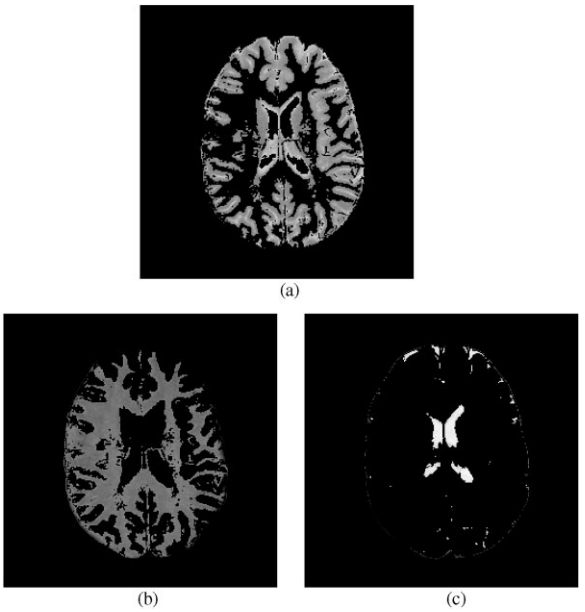


Fig. 7. Classified results in TR_2/TE_2 2500 ms/100 ms image: (a) gray matter; (b) white matter; and (c) cerebral spinal fluid.

completely trained. During the first 5 iterations, the neural states displayed the chaotic dynamics. Then periodic bifurcation appeared and finally an equilibrium point was converged in the last 3 iterations.

6. Discussion and conclusions

In this paper, an approach using the Hopfield neural network imposed by a simulated annealing strategy and chaotic mechanism (named ACNN) is proposed for clustering analysis. To harness the chaotic dynamics, the simulated annealing with the proposed cooling schedule is embedded into ACNN. Accordingly, a near-global minimum can be obtained while the convergence of network was still guaranteed in the proposed algorithm. Instead of the stochastic simulated annealing with decreasing statistic fluctuations based on the Monte Carlo scheme, the annealing process in the ACNN is associated with a series of the bifurcation. It is also unlike the bifurcation mechanism in the conventional chaotic neural network; the ACNN can converge from chaotic state at high temperature through successive bifurcation's during high temperature decreasing process to an equilibrium state at low temperature. In the classification application of multispectral brain images, the promising results can be obtained using the ACNN.

Acknowledgements

This work was supported by the National Science Council, R.O.C., under the Grant # NSC89-2213-E-167-009.

References

[1] J. MacQueen, Some methods for classification and analysis of multivariate observations, Proceedings of the fifth Berkeley Symposium on Mathematical Statistics and Probability, Vol. 281, 1967.
[2] G.H. Ball, D.J. Hall, A clustering technique for summarizing multivariate data, Behav. Sci. 12 (1967) 153.
[3] M.A. Ismail, S.Z. Selim, Fuzzy c-mean: optimality of solutions and effective termination of the algorithm, Pattern Recognition 19 (1986) 481.
[4] H.J. Zimmermann, Fuzzy Set Theory and its Application, Cluwer, Boston, 1991.
[5] J.C. Dunn, A fuzzy relative of the ISODATA process and its use in detecting compact well-separated clusters, J. Cybernet. l.3 (1974) 32.
[6] J.C. Bezdek, Fuzzy Mathematics in pattern classification, Ph.D. Dissertation, Applied Mathematics, Cornell University, Ithaca, New York, 1973.
[7] M.S. Yang, On a class of fuzzy classification maximum likelihood procedures, Fuzzy Sets and Systems 57 (1993) 365.
[8] M.S. Yang, C-F. Su, On parameter estimation for normal mixtures based on fuzzy clustering algorithms, Fuzzy Sets and Systems 68 (1994) 13.
[9] Jzau-Sheng Lin, Fuzzy clustering using a compensated fuzzy Hopfield network, Neural Process. Lett. 10 (1999) 35.
[10] C.P.L. Carnevalli, S. Patarnello, Image processing by simulated annealing, IBM J. Res. Dev. 29 (1985) 569.
[11] H.L. Tan, S.B. Gelfand, E.J. Delp, A cost minimization approach to edge detection using simulated annealing, IEEE Trans. PAMI 14 (1991) 3.
[12] S. Jalali, J.F. Boyce, Determination of optimal general edge detectors by global minimization of a cost function, Image Vision Comput. 13 (1995) 683.
[13] R.W. Klein, R.C. Dubes, Experiments in projection and clustering by simulated annealing, Pattern Recognition 22 (1989) 213.
[14] J.J. Hopfield, Neural networks and physical systems with emergent collective computational abilities, Proc. Natl. Acad. Sci., USA 79 (1982) 2554.
[15] J.J. Hopfield, D.W. Tank, Neural computation of decisions in optimization problems, Biol. Cybernet. 52 (1985) 141.
[16] Jzau-Sheng Lin, Shao-Han Liu, Chi-Yuan Lin, The application of fuzzy Hopfield neural network to design better codebook for image vector quantization, IEICE Trans. Fund. E81-A (1998) 1645.
[17] K.S. Cheng, Jzau-sheng Lin, C.W. Mao, The application of competitive Hopfield neural network to medical image segmentation, IEEE Trans. Med. Imaging 15 (1996) 560.

[18] Jzau-Sheng Lin, K.S. Cheng, C.W. Mao, A fuzzy Hopfield neural network for medical image segmentation, IEEE Trans. Nucl. Sci. 43 (1996) 2389.
[19] Jzau-sheng Lin, K.S. Cheng, C.W. Mao, Multispectral magnetic resonance images segmentation using fuzzy Hopfield neural network, J. Biomed. Comput. 42 (1996) 205.
[20] G.V. Wilson, G.S. Pawley, On the stability of the traveling salesman problem, Biol. Cybernet. 58 (1988) 63.
[21] Y. Yao, W.J. Freeman, Models of biological pattern recognition with spatially chaotic dynamics, Neural Networks 3 (1987) 153.
[22] I. Tsuda, Dynamics link of memory-chaotic memory map in nonequilibrium neural networks, Neural Networks 5 (1992) 313.
[23] K. Aibara, T. Takabe, M. Toyoda, Chaotic neural networks, Phys. Lett. A 144 (1990) 333.
[24] T. Yamada, K. Aihara, M. Kotani, Chaotic neural networks and the traveling salesman problem, Proceedings of the IJCNN'93, Vol. 2, 1993, p. 1553.
[25] H. Nazawa, A neural network model as a globally coupled map and applications based on chaos, Chaos Am. Inst. Phys. 2 (1992) 377.
[26] S. Kirkpatrick, C.D. Gelatt Jr., M.P. Vecchi, Optimization by simulated annealing, Science N.Y. 220 (1983) 671.
[27] N. Metropolis, A.W. Rosenbluth, M.N. Rosenbluth, A.H. Teller, E. Teller, Equations of state calculations by fast computing machines, J. Chem. Phys. 21 (1953) 1087.
[28] S. Geman, D. Geman, Stochastic relaxation, Gibbs distributions and the Baysian restoration of images, IEEE Trans. Pattern Anal. Mach. Intell. 6 (1984) 721.
[29] B.E. Rosen, R. Nakano, Simulated annealing-basics and recent neural network, J. Japan Soc. Artif. Intell. 9 (1994) 365.
[30] K. Kaneeko, Clustering, coding, switching, hierarchical ordering, and control in a network of chaotic elements, Physica D 41 (1990) 137.
[31] K. Aihara, T. Takabe, M. Toyoda, Chaotic neural networks, Phys. Lett. A 144 (1990) 333.
[32] M. Inoue, A. Nagayoshi, A chaos neuro-computer, Phys. Lett. A 158 (1991) 373.
[33] M. Inoue, A. Nagayoshi, Solving an optimization problem with a chaos neural network, Prog. Theor. Phys. 88 (1992) 769.
[34] T. Yamada, K. Aihara, M. Kotani, Chaotic neural networks and the traveling salesman problem, Proceedings of the International Joint Conference on Neural Network Vol. 2, 1993, p. 1549.
[35] L. Chen, K. Aihara, Chaotic simulated annealing by a neural network model with transient chaos, Neural Network 8 (1995) 915.
[36] I. Tokuda, T. Nagashima, K. Aihara, Global bifurcation structure of chaotic neural networks and its application to traveling salesman problems, Neural Network 10 (1997) 1673.
[37] T. Wegner, M. Peterson, Fractal Creations, The Waite Group Inc., 1991.
[38] S. Haykin, Neural Network: A Comprehensive Foundation, Macmillan College Publishing Company, New York, 1994.
[39] Jzau-Sheng Lin, An annealed Hopfield neural network to vector quantization for image compression, Opt. Engng. 38 (1999) 599.
[40] E. Ruspini, Numerical methods for fuzzy clustering, Inform. Sci. 2 (1970) 319.
[41] C.C. Jou, Fuzzy clustering using fuzzy competitive learning networks, IEEE International Conference on Neural Networks, Vol. II, 1992, p. 714.

About the Author—JZAU-SHENG LIN is an Associate Professor of the Department of Electronic Engineering at National Chin-Yi Institute of Technology, Taichung, Taiwan, ROC. He received the B.S. degree in Electronic Engineering from Taiwan University of Science and Technology in 1980, the M.S. and Ph.D. degrees in Electrical Engineering from National Cheng Kung University in 1989 and 1996 respectively. His research interests involve neural network, image compression, pattern recognition, and medical image analysis.