A new learning method for S-GCM

Hamed Rahimov[1], Mohammad Reza Jahed motlagh[2], Nasser Mozayani[3]

Iran University of Science and Technology
March 2006

Abstract

Recently, there have been many studies on artificial neural network models with nonequilibrium dynamics. For example, Ishii et al.'s model, which is an improvement version of Kaneko's globally coupled map (GCM) model, is called globally coupled map using the symmetric map (S-GCM). A new learning method for S-GCM is proposed in this paper. In the proposed method, we use modified saprse matrix for learning method. Both the theory analyses and computer simulation results show that the performance of S-GCM can be improved greatly by using the MIMS learning method. Our learning method named as More Iterate More Store (MIMS) learning. The method is like sparse method, with difference in sparse method. This method to recur the stored patterns and in result it will be dependent on the sequence of storing the patterns, on the other hand, primary patterns have more effect in creating the weight matrix in comparison to the patterns will be stored finally, it means they are recurred more and consequently they are stored and stick better in the memory. It seems this method of learning is more similar to the man's way of learning, as the patterns which we repetition during time we will keep them in our long-term memory better.

Keywords: Chaotic Associative Memory; Symmetric Globally Coupled Map (S-GCM), Sparse Learning Method, MIMS Learning

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]. Many people are certain that chaotic dynamic behavior plays an important role in real neurons and 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. Aihara et al. [8] have 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 analyses due to the spatiotemporal complex dynamics of the neurons. Adachi et al. [9] proposed a system based on model proposed by Aihara et al. A model based on coupled chaotic elements, called globally coupled map model, is proposed by Kaneko [10], and an improvement version of this model, called globally coupled map using the symmetric map, is proposed by Ishii et al. [11]. Ishii et al.'s studies show that in S-GCM both memory capacity and the basin volume for each memory are larger than those in the Hopfield model applying the same learning rule. Moreover, Inoue et al. [12,13] have

[1]Email Address: **Hrahimov@iust.ac.ir**
[2]Email Address: **Jahedmr@iust.ac.ir**
[3]Email Address: **Mozayani@iust.ac.ir**

presented a chaotic neuro-computer in which a neuron is composed of a pair of coupled oscillators. The neuro-computer runs on a deterministic rule, but it is capable of stochastic searching and solving difficult optimization problems. Ishii's modified global coupled chaotic system and Inoue's chaotic neuro-computer are two main chaotic neural networks used for pattern recognition and associative memory. Zhang et al. [14,15] proposed a one-dimensional, two-way coupled map network and a modified definition of an auto-associative matrix. Ishii et al. [16] proposed an associative memory system based on parametrically coupled chaotic elements. The proposed system was obtained by adding a new parameter control to Ishii's previously proposed system. A chaotic activity in an early association stage makes an efficient association over the memories that are stored by means of autocorrelational learning. When the system successfully recalls the target memory, the system's motion is dominated by a spatially coherent oscillation, while unstable motions remain when the system fails to make the association. This system had a large memory capacity. In addition, Zheng et al. [17] proposed a new parameter control method for S-GCM. With the proposed method, the changes of the parameters are decided not only by the value of system partial energy, but also by the difference value of the partial energy. Results showed that the performance of S-GCM could be improved greatly by using the new parameter control method.

In the other hand, Menhaj et al. [18] proposed a learning method for Hopfield network, named sparse learning. They showed that the method learning is compatible to energy function of Hopfield model, and it cans storage target patterns in attractors with minimum energy. Additionally, they proved this learning method has more storage rate and more speed convergence contrast with modified Heb learning method. Based on an analysis of above-mentioned chaotic neural network models and their applications in information processing, to increase the recall speed and capacity of S-GCM, we proposed a new learning method named MIMS learning. Analyses show that the performance of S-GCM can be improved by using the MIMS learning method. Computer simulation results show that the recall speed is increased greatly by using the new method, too.

The rest of this paper is as following. In Section 2, the S-GCM is reviewed briefly. In Section 3, the MIMS learning method is clarified. In Section 4, computer simulation results of solving associative memory problem, by using the conventional learning method and new method, respectively, are presented. Some conclusions are given in Section 5.

2. Globally Coupled Map using the Symmetric map (S-GCM)

This section provides a brief introduction of S-GCM. Ishii et al. give the system.

$$\begin{aligned} x_i(t+1) &= (1-\varepsilon)\cdot f_i[x_i(t)] + \frac{\varepsilon}{N}\cdot\sum_{j=1}^{N} f_j[x_j(t)] \\ f_i(x_i(t)) &= \alpha_i(t)\cdot x_i(t)^3 - \alpha_i(t)\cdot x_i(t) + x_i(t) \end{aligned} \tag{1}$$

Where $x_i(t)$ denotes the ith unit value at time t, N is the number of units, and t is the discrete-time, ε is a constant parameter. The cubic function f, which has a symmetric function shape, can produce chaos with a specific value of its bifurcation parameter α.

The characteristics of S-GCM are determined mainly by the values of its parameters, ε and α. The parameter α indicates the strength of each unit chaotic, and the parameter ε indicates the strength of the coupling. Therefore, as α becomes large the S-GCM becomes chaotic, and as ε becomes large the S-GCM becomes coherent. By using S-GCM for associative memory, one of the most crucial works is to convert the stored patterns into the parameters of S-GCM. There are two methods to complete this work: adjusting parameter α_i and adjusting parameter ε_i. The principles of these two methods are similar. Both of them employ the covariance-learning rule, which is broadly used in associative memory neural networks. So, here we only introduce the method for adjusting parameter α_i (note: the value of parameter ε in Eq. (1) is a constant with this method).

The general strategy of constructing association memory by S–GCM is as follows: An N-dimensional binary coding function C, which converts a state vector, $x \in [-1,1]^N$ to a binary vector $C(x) \in \{-1,1\}^N$ is defined as:

$$C(x)_i = \begin{cases} 1 & x_i > x^* \\ -1 & Otherwise \end{cases} \qquad (2)$$

Where x^* denotes the stationary point of the S–GCM, which is equal to 0. Using this binary coding function C, an S–GCM state can be translated into a N-bit binary representation. The N-dimensional function V converts a binary vector $I \in \{-1,1\}^N$ into a state vector $V(I) \in [-1,1]^N$. Function V is defined as follows:

$$V(I)_i = \begin{cases} x^+ + rand & I_i = 1 \\ x^- + rand & I_i = -1 \end{cases} \qquad (3)$$

Where x^+ and x^- ($x^- < 0 < x^+$) denote the two-cycle periodic solutions of the asymmetric cubic map with $\alpha = \alpha_{\min} = 3.4$, namely $f(x^+) = x^-$ and $f(x^-) = x^+$. The notation $rand$ represents a small random value.

When the S–GCM is regarded as an associative memory system, which processes an N-dimensional binary vector $I \in \{-1,1\}^N$ to an N-dimensional binary output vector $O \in \{-1,1\}^N$, the general strategy of S–GCM for associative memory is:

$$I \xrightarrow{V} x(0) \xrightarrow{S-GCM} x(T) \xrightarrow{C} O$$

S–GCM has two working modes, namely unit map and random evolution, or the preserving mode and the destroying mode. The two modes are all global. If the parameters of each neuron could be controlled, the two modes can switch partially. Let $\left\{\chi^1, \chi^2, ..., \chi^m \middle| \chi^k \in \{1,-1\}^N \right\}$ be a set of N-dimensional binary patterns to be stored. χ_i^k

denotes the i th unit value in the k th binary pattern, and M is the number of stored patterns. The evolution of parameter α_i in Eq. (4) is given by:

$$\alpha_i(t+2) = \alpha_i(t) + [\alpha_i(t) - \alpha_{\min}] \cdot \tanh(\beta \cdot E_i(t))$$
$$E_i(t) = -x_i(t) \cdot \sum_{j=1}^{N} w_{ij} \cdot x_j(t) \quad (4)$$

$$w_{ij} = \frac{1}{N} \sum_{k=1}^{m} \chi_i^k \chi_j^k \quad (5)$$

Where $\alpha_{\max}$, $\alpha_{\min}$ and β are constant parameters. x_i and $E_i(t)$ are the i th system state and the i th partial energy, respectively. Matrix $[w_{ij}]$ is called covariance matrix. The system energy is defined as ($E = \sum_i E_i$). The S-GCM searches for a local minimum of the energy function by making each partial energy, E_i, small and negative as follows. If E_i is high and positive, which means the i th unit value x_i does not suit the covariance matrix, then α_i increases according Eq. (4) and the unit becomes disturbed. During the course of this disturbance, the unit-wise processing mode is changed from the preserving mode to the destroying mode. Before this mode change occurs, the unit is preserving its input. Once the mode is changed, the unit is disturbed enough to make a chaotic motion, which enables the unit to search for proper state. When the unit suits the covariance matrix, E_i becomes small and negative, and α_i becomes small. In this case the unit-wise processing mode is changed from the destroying mode to the preserving mode. When each E_i becomes small and negative, it finally becomes $\alpha_i = \alpha_{\min}$ for all t, and the system output is equal to the stored pattern required.

3. The MIMS learning method

Studies show that one of the disadvantages of S-GCM is that the recall speed is slow [14,15,17]. Additionally, another of the disadvantages of S-GCM is that the memory capacity is low [16]. To increase the recall speed and memory capacity of S-GCM, we use the following learning method instead of modified Heb learning.

Let $\{\chi^1, \chi^2, ..., \chi^m \,|\, \chi^k \in \{1,-1\}^N\}$ be a set of N-dimensional binary patterns to be stored. χ_i^k denotes the i th unit value in the k th binary pattern, and M is the number of stored patterns.

$$t_{ij} = \sum_{k=m}^{1} \sum_{s=k}^{1} \frac{1}{N} \prod_{k=1}^{m} (\chi_i^{(s)} + \chi_j^{(s)})$$
$$T = [t_{ij}], \qquad W = [w_{ij}], \qquad W + = T^T \cdot T \quad (6)$$

Matrix $[w_{ij}]$ is called MIMS matrix. The method is like sparse method, with this difference that in sparse method, T matrix in firstly built according through stored patterns and then

$T^T \cdot T$ product is considered as weight matrix. But in the innovative method, by adding every new pattern, the previous T matrix is also considered and $T^T \cdot T$ producet is calculated and is added to the previous weight matrix. This process will continue till training all patterns. This way of creating the weight matrix will result this method to recur the stored patterns and in result it will be dependent on the sequence of storing patterns. In other words, primary patterns have more effect in creating the weight matrix in comparison to the patterns will be stored finally, it means they are recurred more and consequently they are stored and stick better in the memory. It seems this method of learning is more similar to the man's way of learning, as the patterns which we repetition during time we will keep them in our long-term memory better. The method learning is compatible to energy function of Hopfield model. It cans storage target patterns in attractors with minimum energy. Additionally, will prove this learning method contrast with Heb learning method Have more capacity storage and more speed convergence. For the system, we proposed the following energy function:

$$E = -\sum_{k=m}^{1}\sum_{s=k}^{1}\sum_{i}\sum_{j} w_{ij} \cdot (\chi_i^{(s)} + \chi_j^{(s)})^2 \tag{7}$$

In the following, we prove every change in $\underline{\chi}$ during the process will couse E to decreas this much: $-4 \cdot \sum_{k=m}^{1}\sum_{s=k}^{1} \Delta\chi_r^{(s)} \sum_{j} w_{rj} \cdot \chi_j^{(s)}$.

Let's assume there would be a change in χ_r unit, so Eq. (7) should be:

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

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

The difference of Energy function is as follows:

$$\Delta\chi_r^{(s)} = \chi_{r_2}^{(s)} - \chi_{r_1}^{(s)}$$

$$E_2 - E_1 =$$

$$-\sum_{k=m}^{1}\sum_{s=k}^{1}\left(\sum_{j} w_{rj} \cdot \left[(\chi_{r_2}^{(s)} + \chi_j^{(s)})^2 - (\chi_{r_1}^{(s)} + \chi_j^{(s)})^2\right] + \sum_{i} w_{ir} \cdot \left[(\chi_i^{(s)} + \chi_{r_2}^{(s)})^2 - (\chi_i^{(s)} + \chi_{r_2}^{(s)})^2\right] \right)$$

$$= -\sum_{k=m}^{1}\sum_{s=k}^{1}\left(2 \cdot \sum_{j} w_{rj} \cdot \chi_j^{(s)} \cdot \Delta\chi_r^{(s)} + 2 \cdot \sum_{i} w_{ir} \cdot \chi_i^{(s)} \cdot \Delta\chi_r^{(s)} \right)$$

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

$$\Delta E = -4 \cdot \sum_{k=m}^{1} \sum_{s=k}^{1} \Delta \chi_r^{(s)} \sum_j w_{rj} \cdot \chi_j^{(s)} \tag{8}$$

At first, we show that Eq. (8) 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 secondt 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.

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

$$-4 \cdot \left(\underbrace{\sum_{k=m-1}^{1} \sum_{s=k}^{1} \Delta \chi_r^{(s)} \sum_j w_{rj} \cdot \chi_j^{(s)} + \sum_{s=m-1}^{1} \Delta \chi_r^{(s)} \sum_j \chi_j^{(s)} \cdot w_{rj}}_{Supposed} + \underbrace{\Delta \chi_r^{(m)} \sum_j \chi_j^{(m)} \cdot w_{rj}}_{\Pr oved} \right)$$

According to Induction, it is consistent for m patterns. So that ΔE is always negative or zero. To continue, we will analyze the rule of learning in S-GCM and will notice its performance in this network.

The system energy is defined as $E = \sum_i E_i$. The S-GCM with this method learning searches for a local minimum of the energy function by making each partial energy, E_i, small and negative as follows. If E_i is high and positive, which means the i th unit value x_i does not suit the MIMS matrix, then α_i increases according Eq. (4) and the unit becomes disturbed. During the course of this disturbance, the unit-wise processing mode is changed from the preserving mode to the destroying mode. Before this mode change occurs, the unit is preserving its input. Once the mode is changed, the unit is disturbed enough to make a chaotic motion, which enables the unit to search for proper state. When the unit suits the MIMS matrix, E_i becomes small and negative, and α_i becomes small. In this case the unit-wise processing mode is changed from the destroying mode to the preserving mode. When each E_i becomes small and negative, it finally becomes $\alpha_i = \alpha_{\min}$ for all t, and the system output is equal to the stored pattern required.

4. Experimental results and analysis

We use the conventional learning method and the new method to realize associative memory. Using Eqs. (4), (5) for the conventional method (System-1), and Eqs. (4), (6) for the new one (System-2), and the value of the parameters for the both methods are:

$$\beta = 2, \qquad N = 100, \qquad \alpha_{\min} = 3.4, \qquad \alpha_{\max} = 4, \qquad \varepsilon = 0.1$$

The training set contains five binary patterns (Fig. 1), and the size of each pattern is 10 by 10.

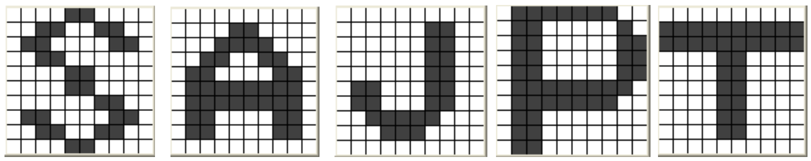


Fig. 1. Patterns to be stored.

The initial value of α_i is $\alpha_{\max} = 4$, and the value of α_i is updated once with Eq. (4) every 2 time steps. To test the effectiveness of the proposed method, for each character we built 3 different patterns by adding 0%, 5%, and 10%, random pepper and salt noise to each pattern. Fig. 2 shows time-series of the overlap in S-GCM with these two methods when the initial overlap is set at various values. All of test patterns can be retrieved correctly and systems can recall the target pattern from a fairly distant initial state. Similar associations have been obtained for other stored patterns.

(a) (b)

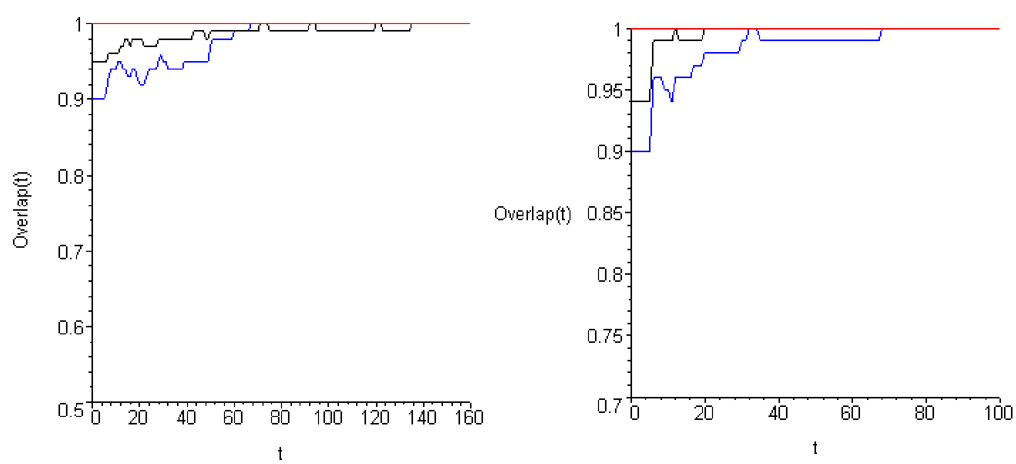


Fig. 2. Time-series of the overlap in S-GCM with two learning methods, P = 5 = 0.05N. (a) Heb learning method. (b) MIMS learning method. The abscissa denotes continuous-time t.

Let us show an example association process. When memorizes five 100-bit binary patterns shown in Fig. 1, and the input binary vector I is a 15% reversed pattern of "A", it associates "A" after scores of transitions. Figures 3(a) and 3(b) show this association process. In Fig.3 (b), the abscissa denotes the association time t and the ordinate the time-series of overlap values. In this figure, highly chaotic motions are observed at the early association stage. As time elapses, these motions become quiet, and the association is completed when the system falls into a 4-cluster frozen attractor. At this time, its binary representation O is successfully equivalent to "A" as Fig. 3(a) shows. Fig. 3(c) shows the time-series of every unit's partial energy E_i. Fig. 3(d) shows the time-series of every unit's α_i value. As this figure shows, at the early stage of association, some of the α_i values are large, which make the unit search for a proper state. However, as time elapses, they become small, and finally all of them come to be equal to $\alpha_{\min}$, which make the system equivalent to the S-GCM and frozen with four clusters.

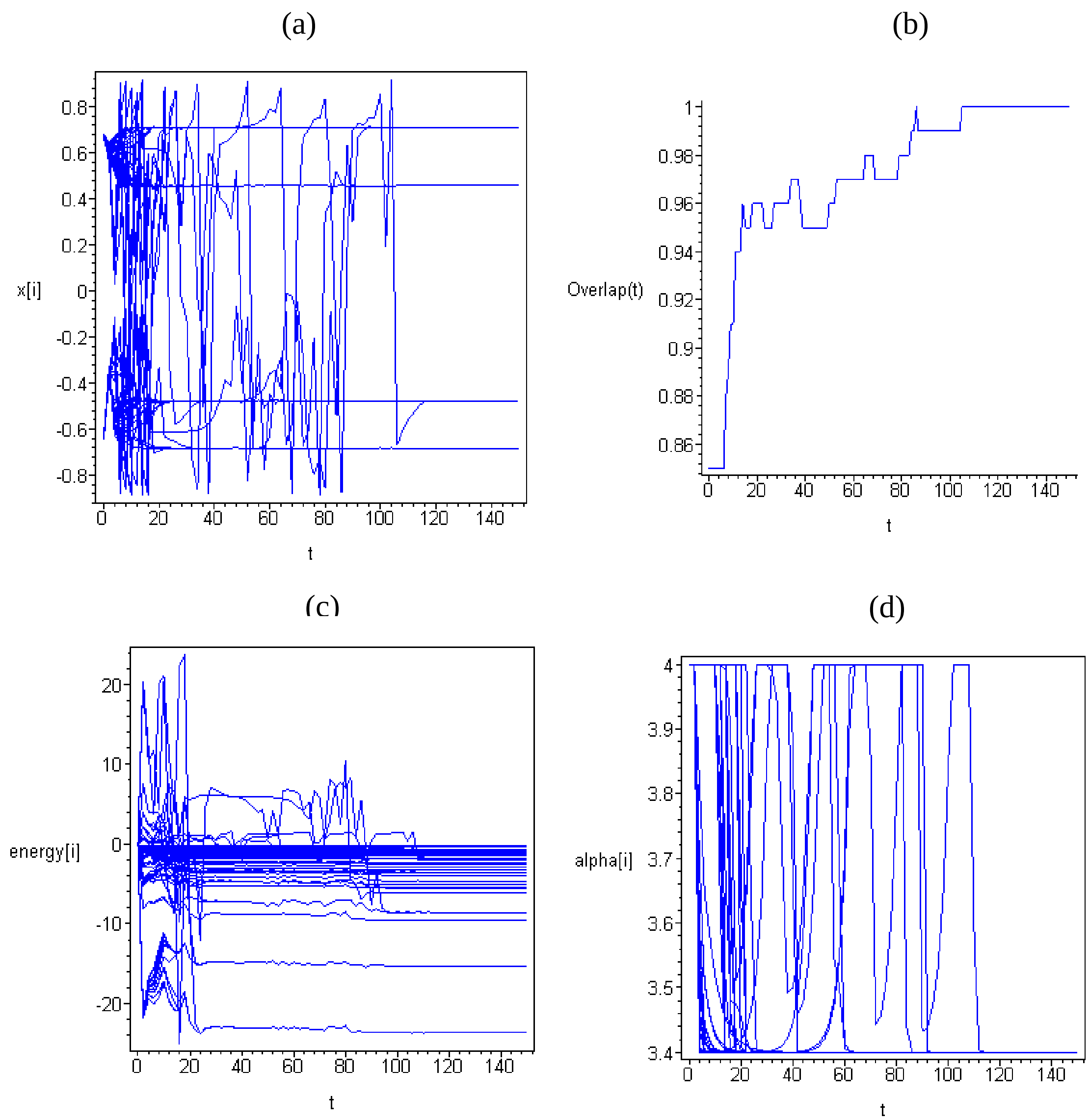


Fig. 3. Time-series in S-GCM with MIMS learning methods for A, (a) output values of all neurons (b) overlap association process (c) every unit's partial energy E_i, (d) every unit's α_i values.

In this system, there will be similar results for the three patterns, which are stored at the beginning, but for the two last patterns, there would not be an appropriate result. If we alter the sequence of storing the patterns, the result would be proper for the primary patterns and inappropriate for the final patterns. We offer a solution to solve this problem. Additionally, to contrast memory storage of these two learning methods, we store 100 Random patterns of 120 patterns (Fig. 4), and the size of each pattern is 10 by 10 for both methods. Then, in retrival stage, we use stored patterns as input patterns.

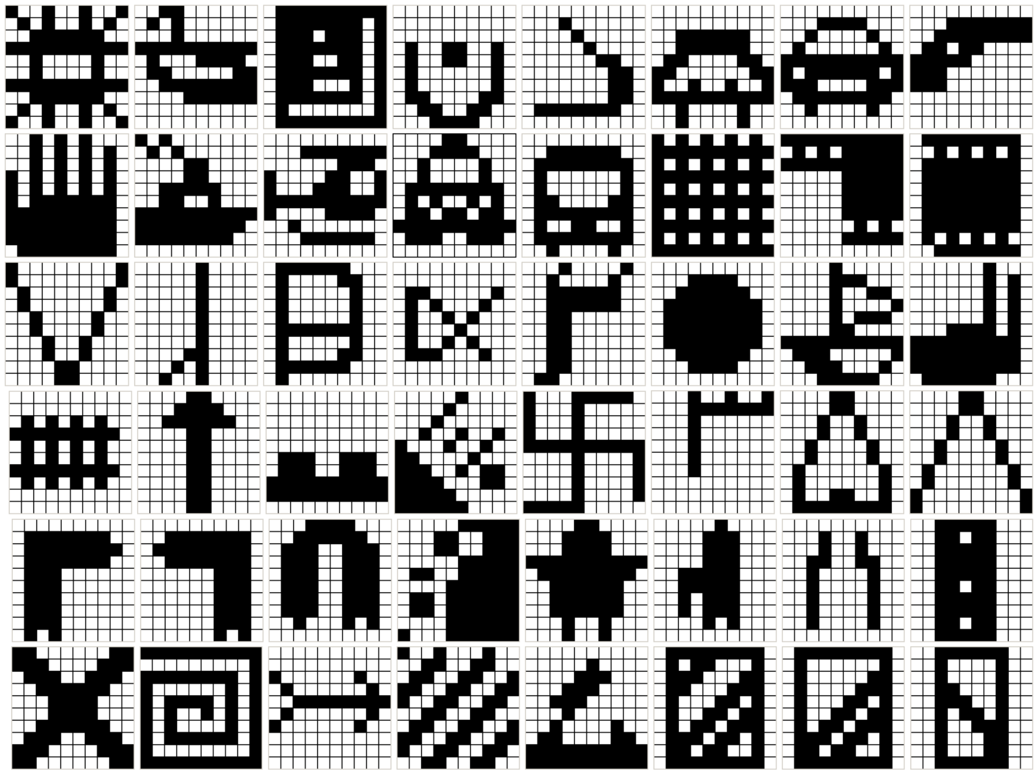


Fig. 4. Patterns to be selected and then stored.

Fig. 5 shows time-series of the overlap of “A” pattern in retrival stage when the initial overlap is set at 1. As you see, in MIMS learning method, input pattern convergenses to target pattern, but in Heb learning method the system cannot recall the target pattern. Similar associations have been obtained for other stored patterns that stored first. Therefore, using MIMS learning method for S-GCM is caused capacity storage of model increased. But for the patterns stored finally, if we give each of the target patterns as input patterns to the system, we will face 5% error, which is again an acceptable result.

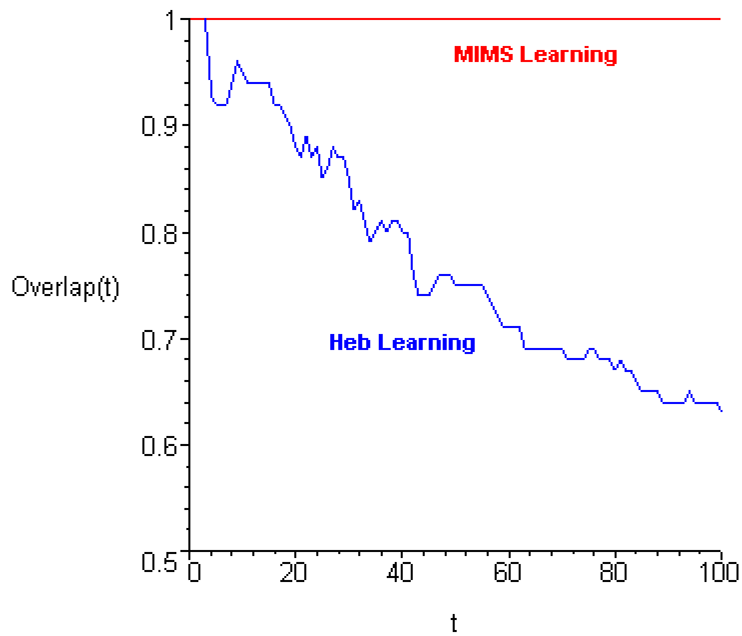
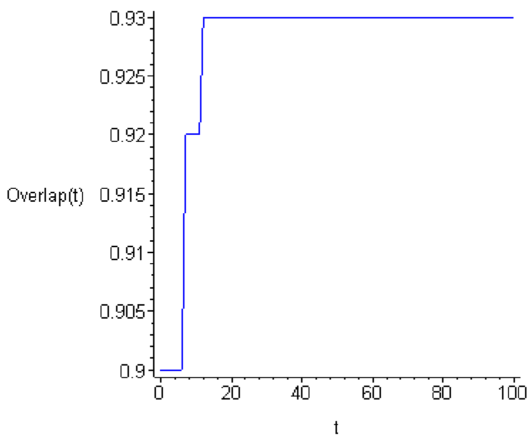


Fig. 5. Time-series of the overlap in S-GCM with two learning methods, P = 100 = N. The abscissa denotes continuous-time t.

The most important point of strength in our system is that for the first-stored patterns, i.e. even at the time the storage rate is high, if we give them as inputs with random noise, our system will be able to correct the noise upto 5%. In the Fig. 6 have shown time-series of the overlap in S-GCM with two learning methods. As shown in this figure, the output of system-1 (Fig. 6(b)) is completely gone far from the target patterns while system-2 (Fig. 6(b)) has done 3% error correction. In general, when the storage rate is high, system-1 is not only unable to correct the error, but also by giving the target pattern itself, it will go completely for as an input. While system-2 will perform the error correction for the primary-stored patterns even when the storage rate is so high.

(a) (b)



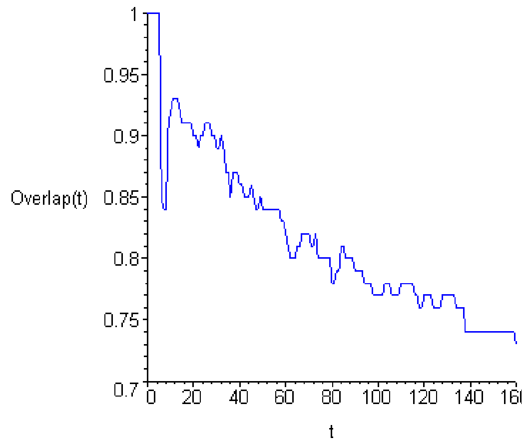


Fig. 6. Time-series of the overlap in S-GCM with two learning methods, P = 100 = N. The abscissa denotes continuous-time t.

The weakest point in the method of MIMS learning is in the final patterns or on the other hand, in the case of the patterns which has less influence in creating weight matrix. In order for the system to have suitable success rate for the final patterns, we will perform the method of learning as follows: for this purpose, we will store the patterns from the beginning to the end once and another time we store the same patterns from the end to the beginning. The result of our implementation indicates that applying this method will increases the final patterns of success rate, especially when the storage rate is low is more conspicuous. The reason for the final patterns of the success rate to be less than the primary patterns is that applying this method will not result in removing the sensitivity in sequence of the pattern storage. The strength of the final patterns in creating the weight matrix wil increases more only to some extent, but yet, the primary patterns have more influence in creating the weight matrix.

5. Conclusion

In this paper, MIMS learning method for S-GCM is proposed. The core of this method is that using MIMS matrix instead of covariance matrix for learning S-GCM. Experimental results show that the performance of S-GCM is increased greatly with the new method.

References

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

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

[3] Sinha, S.; Munakata, T., & Ditto, W.L., “Chaos Computing: Implementation of Fundamental Logical Gates by Chaotic Elements,” IEEE TRANS, Volume 49, No.11, 2002.

[4] 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.

[5] Sinha, S.; Murali, K., & Ditto, W.L., “Computing with chaos,” Physical Review. E, Statistical, Nonlinear, And Soft Matter Physics, Volume 68, Issue 1, Part 2, Page 016205, 2004.
[6] 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.
[7] 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.
[8] 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.
[9] 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.
[10] Kaneko, K., “Clustering, coding, switching, hierarchical ordering, and control in a network of chaotic elements,” Physica D, Volume 41, Pages 137-172, 1990.
[11] Ishii, S.; Fukumizu, K.; Watanabe, S., “A network of chaotic elements for information processing,” Neural Networks, Volume 9, Issue 1, Pages 25–40, 1996.
[12] Inoue, M.; Nagayoshi, A., “A chaos neuro-computer,” Phys. Lett. A, Volume 158, Pages 373–376, 1991.
[13] Inoue, M.; Fukushima, S., “A neural network of chaotic oscillators,” Progress. Letters, Volume 87, Issue 3, 1992.
[14] Zhang, Y.; Yang, L.; He, Z., “Chaotic neural network for associative memory,” J. Appl. Sci., Volume 17, Issue 3, Pages 259–266, 1999.
[15] Zhang, Y.; Yang, L.; He, Z., “Study of chaotic neural network and its applications in associative memory,” Neural Processing Letters, Volume 9, Issue 2, Pages 163-175, 1999.
[16] Ishii, S.; Sato, M., “Associative memory based on parametrically coupled chaotic elements,” Physica D, Volume 121, Pages 344-366, 1998.
[17] Zheng, L.; Tang, X., “A new parameter control method for S-GCM,” Pattern Recognition Letters, Volume 26, Issue 7, Pages 939-942, 2005.
[18] M.B. Menhaj, Fundamentals neural networks: Computational intelligence, 2nd edition, 2002, Amirkabir university, AKU press, Volume 1, Pages 222-229.