2007 International Symposium on Nonlinear Theory and its Applications
NOLTA'07, Vancouver, Canada, September 16-19, 2007

An Analysis of Non-periodic Oscillation During a Learning of a Complex-valued BAM

Ryuuichi Sasaki† and Masaharu Adachi‡

†Department of Electronic Engineering, Graduate School of Engineering, Tokyo Denki University
2-2 Kanda–Nishiki–cho, Chiyoda–ku, Tokyo 101–8457, Japan
Email: †06gmd09@ed.cck.dendai.ac.jp,‡adachi@d.dendai.ac.jp

Abstract—Bidirectional associative memories have been proposed by Kosko[1]. Kosko's model consists of two layers of neurons with synaptic connections between the layers. In the original model, the synaptic weight connections are determined by a correlation matrix of stored patterns. However, when the training pattern vectors are not orthogonal, cross-talk noise may occur in the system. To solve such problem, Oh.*et.al.* have been proposed Pseudo-relaxation learning algorithm for BAM (PRLAB) which uses a projection method[2]. PRLAB does not require orthogonality or any special encoding of the training pairs and extremely increase the storage capacity. We extend the real-valued PRLAB algorithm to complex value. In the case that the complex-valued BAM trained by the extended PRLAB, we show that the storage capacity increases as the same as in the real-valued PRLAB. Moreover, it is also shown that when the complex-valued BAM fails to store the patterns to have high storage capacity, the time series of directional cosine during the training show non-periodic oscillation. We analyze the non-periodic oscillation in the present paper. As a result the non-periodic oscillation has a low-dimensional structure.

1. Introduction

Associative memory models which use neural networks have been studied since Hopfield's associative memory model was invented[3]. It is widely known that synaptic connection weights of the Hopfield model are determined by a correlation matrix of the stored patterns. Another popular associative memory is a bidirectional associative memory (BAM) which was proposed by Kosko[1]. The BAM consists of two layers of neurons with symmetric synaptic connections between the layers. The BAM behaves as a hetero-associative memory which enables to store and recall pairs of patterns. When a noisy pattern is presented to the BAM, the recalling process of the BAM can correctly reconstruct the pattern through a sequence of successive updates until it reaches to a steady state. The feedback mechanism of the BAM helps to filter the noise as successive updates of the network.

However, because the BAM uses correlation learning, as many associative models do, the memory capacity is poor. To solve such a problem, Oh.*et.al.* have been proposed Pseudo-relaxation learning algorithm for BAM(PRLAB) which uses a projection method. PRLAB does not require orthogonality or any other special encoding of the training patterns and it shows a lager storage capacity than that of the correlation learning. A learning of synaptic weights can be represented by a relaxation method for solving a system of linear inequalities. If training pairs are not contradiction, it is guaranteed that the network recalls all the training pairs. Hattori.*et.al.*[4] proposed QLBAM which combines Hebbian learning with PRLAB. It was shown that memory capacity is greatly improved.

In recent years, complex-valued neural network models have been proposed and applied to associative memory models and it is shown that such networks show different features from the real-valued networks. A complex-valued auto-associative network stores patterns that are presented in complex-valued region. In contrast to a real-valued neuron which usually stores patterns with two states. A complex-valued neuron easily stores various states as the phase of the complex value. Therefore, it can be applied to store natural scene pictures. However, it has been reported that the more states a neuron takes, the less memory capacity the associative network consists of the neuron shows.

Therefore, an improvement of the storage capacity is an important problem of complex-valued associative memory model. In this paper, we extend the real-valued PRLAB algorithm to complex value. Moreover, it is also shown that when the complex-valued BAM fails to store patterns to have high storage capacity, the time series of directional cosine during the training show non-periodic oscillation.

2. Complex-valued Neural Networks

In this section, we describe complex-valued neural networks. There are two kinds for complex-valued neuron models. One is a continuous-state model and the other is a discrete-state model. A discrete-state neuron is allowed to take one of the equally partitioned phases on the unit circle of the complex plane. The operation of a complex-valued neuron is represented as Eq.(1) - (3). Let W_{ij} denotes the complex-valued connection weight associated to the coupling from the j-th neuron to the i-th one. Let θ_i be the threshold for the i-th neuron. Eq.(1) shows the updating of the internal state of a single neuron.

$$S_i(t+1) = \sum_{j=1}^{N} W_{ij} X_j - \theta_i \quad (1)$$

The discrete-state models' output is represented by

$$x_i(t+1) = exp\left(i\frac{2\pi l}{k}\right), \quad (2)$$

$$if \quad \frac{2\pi(2l-1)}{k} < arg(s_i(t+1)) < \frac{2\pi(2l+1)}{k}$$

where l=0, 1, $\cdots$, K-1, and K denotes the resolution of the the complex-valued neuron. Therefore, the neuron has K states for its output. Eq.(2) implies that the absolute value of every state of the neuron is unity. The discrete-state model takes the nearest phase among Eq.(2) on the unit circle of the complex plane as its output.

The continuous-state models' output is represented by

$$x_i(t+1) = exp\left(i \; arg(s_i(t+1))\right). \quad (3)$$

A continuous-state neuron model takes $e^{i\theta}$ when the internal state of the neuron is $re^{i\theta}$ ($r > 0$ and $0 < \theta < 2\pi$).

Therefore, the output of both continuous-state and discrete-state model is on the unit circle of complex plane and its absolute value is unity. However, it has been reported that the more resolution factor K a neuron takes, the less memory capacity the network shows. For example, when the resolution factor K=4, the storage capacity of a complex-valued Hopfield model is 0.07, when, K=8 the storage capacity is only 0.025 [5]. These examples show that the storage capacity of the complex-valued Hopfield model is smaller than that of the real-valued one with a conventional learning rule like the Hebbian learning.

3. PRLAB

In this section, we describe pseudo-relaxation learning for BAM(PRLAB)[2]. At first, we consider an N-M BAM which means N neurons in the first layer and M neurons in the second layer. Let W_{ij} be the connection weight between the i-th neuron in the first layer and the j-th neuron in the second layer. Let θ_{x_i} and θ_{y_j} be the threshold for the i-th neuron in the first layer and the one for the j-th neuron in the second layer, respectively. Let $\boldsymbol{V} = \{(\boldsymbol{X}^{(p)}, \boldsymbol{Y}^{(p)})\}_{p=1,\cdots,P}$ be a set of pairs of training patterns. Suppose that each training vector consists of elements that are $\boldsymbol{X}^{(p)} \in \{-1,1\}^N$ and $\boldsymbol{Y}^{(p)} \in \{-1,1\}^M$. The vector $\boldsymbol{V}$ is guaranteed to be recalled if the following system of linear inequalities are satisfied for all $\boldsymbol{p = 1, \cdots, P}$

$$\left(\sum_{p=1}^{n} W_{ij} X_i^{(p)} - \theta_{Y_j}\right) Y_j^{(p)} > 0, \quad (4)$$

for $\boldsymbol{j}$=1,$\cdots$,M ,

$$\left(\sum_{p=1}^{n} W_{ij} Y_j^{(p)} - \theta_{x_i}\right) X_i^{(p)} > 0, \quad (5)$$

for $\boldsymbol{i}$=1,$\cdots$,N .

In PRLAB each training pair of patterns $\{\boldsymbol{X}^{(p)}, \boldsymbol{Y}^{(p)}\}$ is presented sequentially, then the weights and thresholds are updated if Eq.(4) and (5) are not satisfied. The updating is executed by the following updating rules. For the neurons in the first layer

if $S_{X_j}^{(p)} X_j^{(p)} \leq 0$

$$\Delta W_{ij} = -\frac{\lambda}{1+M}\left(S_{X_i}^{(p)} - \xi X_i^{(p)}\right) Y_j^{(p)}, \quad (6)$$

$$\Delta\theta_{X_i} = \frac{\lambda}{1+M}\left(S_{X_i}^{(p)} - \xi X_i^{(p)}\right) \quad (7)$$

where

$$S_{X_i}^{(p)} = \sum_{j=1}^{M} W_{ij} Y_j^{(p)} - \theta_{X_i}. \quad (8)$$

For the neurons in the second layer,

if $S_{Y_j}^{(p)} Y_j^{(p)} \leq 0$

$$\Delta W_{ij} = -\frac{\lambda}{1+N}\left(S_{Y_j}^{(p)} - \xi Y_j^{(p)}\right) X_i^{(p)}, \quad (9)$$

$$\Delta\theta_{Y_j} = \frac{\lambda}{1+N}\left(S_{Y_j}^{(p)} - \xi Y_j^{(p)}\right) \quad (10)$$

where

$$S_{Y_j}^{(p)} = \sum_{i=1}^{N} W_{ij} X_i^{(p)} - \theta_{Y_i}. \quad (11)$$

The above updating is repeated until the synaptic weights converge. It is shown that the updating converges if the relaxation coefficient $\boldsymbol{\lambda}$ is in between 0 and 2 [6].

4. Complex-valued PRLAB

In Sec.3, we described the condition for the recalling of all the training pairs with real-valued PRLAB. This condition represents that the signs of the internal state of the neuron and the desired output are the same. We consider to extend real-valued PRLAB to complex-value. We assume that the recalling condition for complex-valued BAM is as follows:

In every neuron, the real part of the product of the output and conjugate of the desired output becomes unity and the complex part of it is naught.

By considering the recalling condition for a complex-valued BAM, the condition for updating the weights is as follows. Considering the first layer, product of the output and the desired output take the value on the unit circle in the complex plane, this formula means that we can not extend the recalling condition of Eq.(4) and (5) to the complex-valued BAM as they are. Instead, we consider the product of the output and the conjugate of the desired output. Therefore, we introduce a condition that the real part of the product of the output and the conjugate of the desired output is close enough to unity and the imaginary part of it should be close enough to naught, as represented by the following formulas. For the neurons in the first layer

$$\mathbf{1} - \boldsymbol{Re}\left\{\left(\sum_{p=1}^{n} W_{ij} X_i^{(p)} - \theta_{y_j}\right)\overline{Y_j^{(p)}}\right\} < \epsilon, \qquad (12)$$

for j=1,⋯, M.

$$\boldsymbol{Im}\left\{\left(\sum_{p=1}^{n} W_{ij} X_i^{(p)} - \theta_{y_j}\right)\overline{Y_j^{(p)}}\right\} < \epsilon, \qquad (13)$$

for j=1, ⋯, M.

For the neurons in the second layer

$$\mathbf{1} - \boldsymbol{Re}\left\{\left(\sum_{p=1}^{n} W_{ij} Y_j^{(p)} - \theta_{x_i}\right)\overline{X_i^{(p)}}\right\} < \epsilon, \qquad (14)$$

for j=1, ⋯,N.

$$\boldsymbol{Im}\left\{\left(\sum_{p=1}^{n} W_{ij} Y_j^{(p)} - \theta_{x_j}\right)\overline{X_i^{(p)}}\right\} < \epsilon \qquad (15)$$

for j=1,⋯,N.

In Eq.(12)-(15) ϵ should be a small value, for example ϵ=0.05.

Then, the weights and the thresholds are updated if the conditions of Eq.(12)-(15) are not satisfied, the updating in the first layer of the weights and the threshold is executed by the following equations.

$$\boldsymbol{Re}(\Delta W_{ij}) = \frac{\lambda}{1+M}\overline{X_i^{(p)}}\left((1+\xi) - \overline{X}_i^{(p)} S_{X_i}^{(p)}\right) Y_j^{(p)}, \qquad (16)$$

$$\boldsymbol{Re}(\Delta\theta_{X_i}) = \frac{\lambda}{1+M}\overline{X_i^{(p)}}\left((1+\xi) - \overline{X}_i^{(p)} S_{X_i}^{(p)}\right) \qquad (17)$$

where

$$S_{X_i}^{(p)} = \sum_{j=1}^{M} W_{ij} Y_j^{(p)} - \theta_{X_i} \qquad (18)$$

$$\boldsymbol{Im}(\Delta W_{ij}) = -\frac{\lambda}{1+M}\left(S_{X_i}^{(p)} - \xi X_i^{(p)}\right) Y_j^{(p)}, \qquad (19)$$

$$\boldsymbol{Im}(\Delta\theta_{X_i}) = -\frac{\lambda}{1+M}\left(S_{X_i}^{(p)} - \xi X_i^{(p)}\right) \qquad (20)$$

where the real part and imaginary part of a complex number z are denoted by $\boldsymbol{Re}(z)$ and $\boldsymbol{Im}(z)$, respectively. Cojugate of z is denoted by $\overline{z}$. The updating in the second layer of the weights and the threshold is executed by the following equations.

$$\boldsymbol{Re}(\Delta W_{ij}) = \frac{\lambda}{1+N}\overline{Y_j^{(p)}}\left((1+\xi) - \overline{Y}_j^{(p)} S_{Y_j}^{(p)}\right) X_i^{(p)}, \qquad (21)$$

$$\boldsymbol{Re}(\Delta\theta_{Y_j}) = \frac{\lambda}{1+N}\overline{Y_j^{(p)}}\left((1+\xi) - \overline{Y}_j^{(p)} S_{Y_j}^{(p)}\right), \qquad (22)$$

$$\boldsymbol{Im}(\Delta W_{ij}) = -\frac{\lambda}{1+N}\left(S_{Y_j}^{(p)} - \xi Y_j^{(p)}\right) X_i^{(p)}, \qquad (23)$$

$$\boldsymbol{Im}(\Delta\theta_{Y_j}) = -\frac{\lambda}{1+N}\left(S_{Y_j}^{(p)} - \xi Y_j^{(p)}\right) \qquad (24)$$

where

$$S_{Y_j}^{(p)} = \sum_{i=1}^{N} W_{ij} X_i^{(p)} - \theta_{Y_i} \qquad (25)$$

5. Numerical Experiment and Result

In this section, we apply complex-valued PRLAB to a BAM. We observe characteristics of the training when the complex-valued BAM fails to store patterns. In the following numerical experiments, we use conditions as follows: the resolution factor K= 4, the number of the neurons in both layers is 5 (5-5 BAM). ξ = 0.1, the number of training patterns is 4. The training patterns are randomly generated. With these conditions, when the directional cosine between the output of a layer and the one of the stored pattern is unity we consider that the pattern is recalled.

Figure 1 shows a bifurcation diagram of the directional cosine of the network during the training by the complex-valued PRLAB with changing the relaxation factor λ as the bifurcation parameter in between 0 to 2. The bifurcation diagram of Fig.1 is obtained during the training step of 30000 to 31000.

Figure 2 shows the same bifurcation diagram as Fig.1 except that the bifurcation parameter λ are changed from 1.94 to 2.01. From Fig.2, we find that period doubling bifurcations are occurred by changing the relaxation factor λ.

We reconstruct an attractor from the time series of directional cosine of the network during the fails of the training of storing patterns by plotting the data in a delay coordinate of three dimensions. The following figure is obtained by the conditions as follows: the resolution factor K=3, the number of the neurons in both layers is 5 (5-5 BAM). ξ = 0.1254, λ = 1.8, the number of the training patterns is 4.

Figure 3, shows the reconstructed attractor of the directional cosine with time delay parameter τ=100. Figure 3 is obtained by plotting the data during of training step 30000 to 600000.

6. Conclusion and Discussion

We consider complex-valued BAM whose input, output, and weights, and threshold are complex numbers. We extend the real-valued PRLAB algorithm to complex value. Moreover, it is shown that when the complex-valued BAM fails to store the patterns to have high storage capacity.

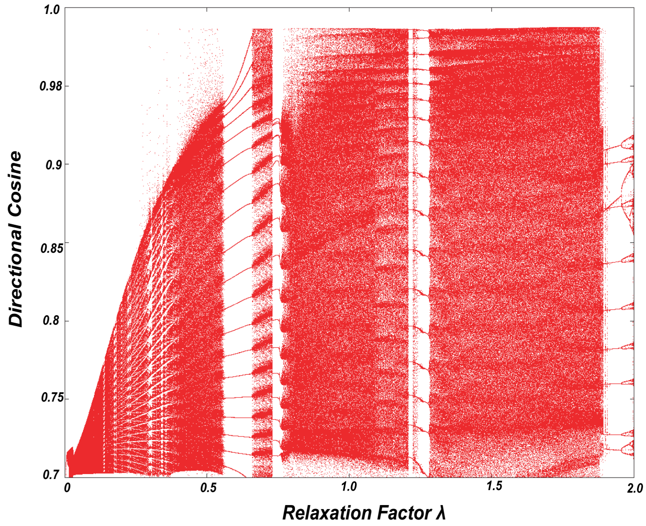

Figure 1: Bifurcation diagram of the directional cosine by changing the relaxation factor λ ($\mathbf{0 < \lambda < 2}$).

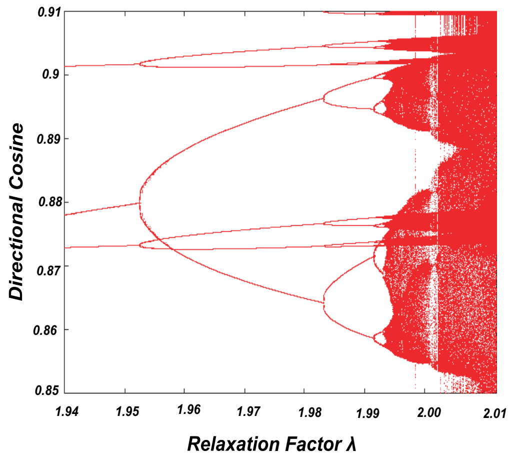

Figure 2: Enlarged bifurcation diagram of Fig1. ($\mathbf{1.94 < \lambda < 2.01}$).

When we vary relaxation factor λ for the training with the complex-valued PRLAB, directional cosine show period doubling bifurcation. Next, we reconstruct an attractor of the time series of the directional cosine of the network. The reconstructed attractor shows that the training dynamics have a certain low-dimensional structure.

It has been reported that when acomplex-valued associative memory model fails to recall memorized patterns. It shows non-periodic oscillation[8]. The result shown in the present paper differs from the one in Ref.[8], because the present result is for a failure during the training. It is a future problem to see the relation between these results.

Acknowledgement

The authors are grateful to anonymous reviewer for his/her valuable comments.

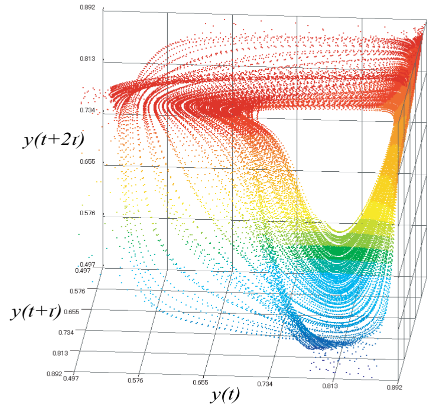

Figure 3: Reconstructed attractor of the time series of directional cosine during training step of 30000 to 60000 with time delay τ=100.

References

[1] B. Kosko, "Bidirectional associative memories," *IEEE Trans.Syst.,* Man & Cybern., Vol.18, No.1, pp.49-60, 1988

[2] H. Oh and S.C. Kothari, "Adaptation of the relaxation method for leaning in didirectional associative memory," *IEEE Trans.Neural Networks.,*Vol.5, No.4, July 1994.

[3] J. J. Hopfield, "Neural networks and physical system with emergent collective computational abilities," *Proc. Natl. Acad. Sci. USA,*Vol.79, pp.2554-2558 1982.

[4] M. Hattori,M. Hagiwara, and M. Nakagawa "Quick learning for bidirectional associative memory," *IEICE Trans. Inf. & Syst.,*Vol. E-D, No.4, April 1994.

[5] S.Jankowski,A.Lozowski,and J.M.Zurada "Complex-valued multistate neural associative memory," *IEEE Trans, Neural Networks.*, vol. 7, no. 6, pp.1491-1496, 1996.

[6] S. Agmon "The relaxation method for linear inequalities ," *Canadian J.Math.*, vol.6, No3, pp.382-392, 1954.

[7] I. Nemoto and T. Kono "Complex-neural networks," *Systems and Computers in Japan.*, vol.23, No3, pp.382-392, 1954.

[8] I. Nemoto and M. Kubono, "Complex associative memory," *Neural Netw.* ,vol. 9, no. 2, pp.5-84, 1992.