Fifth IFSA World Congress (1993), 1009-1012

Functions of Chaos Neuron Models with a Feedback Slaving Principle

Masayoshi Inoue
Department of Physics, Kagoshima University
Kagoshima 890, Japan

Key words: chaos, neural network, association, optimization problem, learning

Abstract

An association memory, solving an optimization problem, a Boltzmann machine scheme learning and a back propagation learning in our chaos neuron models are reviewed and some new results are presented. In each model its microscopic rule (a parameter of a chaos system in a neuron) is subject to its macroscopic state. This feedback and chaos dynamics are essential mechanisms of our model and their roles are briefly discussed.

1 Introduction

In recent years, there has been increasing interest in the application of chaos to technology. Harnessing chaos to information processing may be one of the most interesting applications, and several authors have discussed on this subject. We have presented chaos neuron models and their abilities of parallel synchronous computation and learning have been studied in previous papers. The power of the ability of parallel synchronous computation have been illustrated for an association memory [1, 2] and solving a difficult optimization problem [1, 2, 3]. An important issue in neurocomputing is learning. Our chaos neuron models have the ability of not only the parallel synchronous computation but ability of learning: (1) Boltzmann machine scheme high speed learning without simulated annealing [4], (2) stochastic back propagation learning [5].

2 Chaos neuron models

2.1 A coupled-oscillator model

The neuron of the first model [1] is realized by two chaos oscillators which are coupled each other. We adopt discrete time ($n = 0, 1, 2, 3, ...$) to illustrate dynamics of the model. The equations of motion of the coupled oscillators in the i-th neuron are described as follows:

$$\begin{pmatrix} x_i(n+1) \\ y_i(n+1) \end{pmatrix} = \frac{1}{1+2D_i(n)} \begin{pmatrix} 1+D_i(n) & D_i(n) \\ D_i(n) & 1+D_i(n) \end{pmatrix} \begin{pmatrix} f[x_i(n)] \\ g[y_i(n)] \end{pmatrix}, \quad (1)$$

where $D_i(n)$ is the coupling coefficient between the two oscillators in the i-th neuron at time n, and $x_i(n)$ $(0 \le x_i(n) < 1)$ and $y_i(n)$ $(0 \le y_i(n) < 1)$ are the variables of the first and the second oscillators in the i-th neuron at time n, respectively. This system reduces to Yamada-Fujisaka model [6] if $f(x) = g(x)$ and $D_i(n)$ is a constant. For the sake of simplicity the equation of motion of the uncoupled oscillators are chosen as $f(x) = ax(1-x)$ and $g(y) = by(1-y)$, where a $(0 < a \le 4)$ and b $(0 < b \le 4)$ are the control parameters.

When the coupling coefficient $D_i(n)$ becomes large the two oscillators synchronized each other if $a = b$ and nearly synchronized if $a \simeq b$. And an asynchronous motion appears if $D_i(n)$ takes a certain small value. These phenomena can be measured by the difference $\Delta_i(n)$ which is defined by

$$\Delta_i(n) = |x_i(n) - y_i(n)|. \quad (2)$$

The binary state of the i-th neuron $u_i(n)$ is defined using $\Delta_i(n)$ as

$$u_i(n) = \begin{cases} 1 & \text{(excitation)} \quad \text{if } \Delta_i(n) < \epsilon, \\ 0 & \text{(inhibition)} \quad \text{otherwise,} \end{cases} \quad (3)$$

where ϵ is the criterion parameter of the synchronization.

We synthesize a network of the binary neurons where i-th and j-th neurons are connected to each other with the weight w_{ij}. The state of neurons have an influence on $D_i(n)$ through the medium of the connection. The relation between w_{ij} and $D_i(n)$ is defined as follows:

$$DD_i(n) = \sum_j w_{ij} u_j(n) + s_i - \theta_i, \quad (4)$$

$$D_i(n) = \begin{cases} DD_i(n) & \text{if } DD_i(n) > 0, \\ 0 & \text{otherwise,} \end{cases} \quad (5)$$

where s_i is the external input and θ_i is the threshold value. We put a limitation of the coefficient as $D_i(n) \ge 0$ which prevents the coupled-oscillator from unsuitable motion.

We choose the value θ_i in order to the coupled-oscillator takes the critical state of synchronization when $w_{ij} = 0, s_i = 0$. The control parameters of the coupled oscillators are usually taken as $b \simeq a = 4$, and this case θ_i is calculated as $\theta_i = -0.5$ [1].

An analog version of the binary model can be easily obtained if we change the rule of Eq.(3). An example is given by [3]

$$u_i(n) = \frac{1}{1+\exp[-z_i(n)/z_0]}, \quad (6)$$

with

$$z_i(n) \equiv \frac{\epsilon}{\Delta_i(n)} - 1, \tag{7}$$

where z_0 is the analog parameter. This analog neuron becomes the binary one if $z_0 \to 0$.

2.2 A single oscillator model

In the second model, a single chaotic oscillator in a critical state acts as a neuron [2]. The equation of the oscillator is described by

$$x_i(n') = a_i(n')x_i(n')[1 - x_i(n')], \tag{8}$$

where $a_i(n')$ $(0 < a_i(n') \leq 4)$ and $x_i(n')$ $(0 \leq x_i(n') < 1)$ are the control parameter and the variable of i−th oscillator at the internal time n', respectively.

The logistic map $f_a(x) = ax(1-x)$ has a wide window of a period three, namely a stable period three motion appears if the control parameter a exceeds the critical value $a_c = 1 + 2\sqrt{2}$ and chaos is produced below the critical point. The neuron is designed with the aid of the critical state. The control parameter $a_i(n')$ changes its value at every m steps, namely

$$a_i(mn) = \sum_j w_{ij}u_j(n) + s_i - \theta_i, \tag{9}$$

$$\begin{aligned} a_i(mn) &= a_i(mn+1) = ... \\ &= a_i(m(n+1) - 1), \end{aligned} \tag{10}$$

where n $(= 0, 1, 2, 3, ...)$ is the external time and θ_i is taken as $\theta_i = -a_c = -(1 + 2\sqrt{2})$. Occasionally, the value of r.h.s of Eq.(9) takes a small (large) one, which brings unsuitable motion, therefore we put the limitations that $\mathrm{Min}\{a_i(mn)\} = (a_c - c_1)$ and $\mathrm{Max}\{a_i(mn)\} = (a_c + c_2)$, where c_1 and c_2 are the positive parameters. The value of the neuron is determined in the same way with Eq.(3) or Eq.(6), but the difference $\Delta_i(n)$ is defined by

$$\Delta_i(n) = | x_i(mn) - x_i(mn+3) | . \tag{11}$$

This neuron becomes a formal neuron, which has no transient time, if $m \to \infty$ with $\epsilon = 0$. A network of these neurons has a good function for a self association with a set of parameters $\{m = 8, \epsilon = 0.0001, c_1 = c_2 = 0.005\}$, however it spends much steps to solve a TSP (Traveling Salesman Problem) [2].

2.3 A 3-valued model

The third model is a network of three-valued neurons. A system of coupled 3-oscillator is used for the neuron. The equation of motion of the system is given by

$$\begin{pmatrix} x_i(n'+1) \\ y_i(n'+1) \\ z_i(n'+1) \end{pmatrix} = \frac{1}{1 + 3D_i(n')} \begin{pmatrix} 1 + D_i(n') & D_i(n') & D_i(n') \\ D_i(n') & 1 + D_i(n') & D_i(n') \\ D_i(n') & D_i(n') & 1 + D_i(n') \end{pmatrix} \begin{pmatrix} f[x_i(n')] \\ g[y_i(n')] \\ h[z_i(n')] \end{pmatrix}. \tag{12}$$

This system has three different phases of motion according to the value of the coupling coefficient $D_i(n') = D$ if the three oscillators are the same each other $f(x) = g(x) = h(x)$ [7]. Namely if $f(x) = 4x(1-x)$ a synchronous motion, a partially synchronous motion and an asynchronous motion are observed when $D > 1/3$, $D \simeq 0.2$ and $D \simeq 0$, respectively. In a similar manner to the first model, a well defined three-valued neuron can be obtained with the aid of these motions. The value of the state $u_i(n)$ is defined by

$$u_i(n) = \begin{cases} 1 & \text{for synchronous,} \\ 0 & \text{for partially synchronous,} \\ -1 & \text{for asynchronous.} \end{cases} \tag{13}$$

The judgments of the phases of the motions have been made using the criterion parameter ϵ in the same way as Eq.(3) where the three maps slightly different from each other.

The coupling coefficient of the i−th neuron $D_i(n')$ is changed according to the rules which are the same with Eqs.(9) and (10), where θ_i is settled as $\theta_i = -0.2$. The three-valued neuron models is superior to that of the binary neuron in some respects.

2.4 A multi-valued model

The real neuron is a multi-valued one whose value is expressed by the density of impulses. The fourth model is synthesized by multi-valued neurons whose impulses are generated by chaotic sequences. Internal dynamics of the neuron is driven by the map [8] which takes in the following form:

$$f_r(x) = \frac{1}{2}[\sin\{r\pi(x - \frac{1}{2})\} + 1], \tag{14}$$

where r $(0 < r \leq 3)$ and x $(0 \leq x < 1)$ are the control parameter and the variable, respectively.

A time sequence, which is generated by the map, is transformed into a symbolic sequence owing to the rule; (1) ↑ if $0 \leq x < 1/2$, (2) ↓ if $1/2 \leq x < 1$. Property of the symbolic sequence depends on the control parameter r that (1) a ferromagnetic sequence {↑, ↑, ↑, ↑, ↑, ...} is obtained if $0 < r \leq r_F(= 2)$, (2) a random sequence is obtained if $r \simeq 2.76$, (3) an anti-ferromagnetic sequence {↑, ↓, ↑, ↓, ↑, ...} is obtained if $3 \geq r > r_{AF}(= 2.930576...)$ [8]. We assume that an impulse is generated whenever the direction of the spin changes with time step. The number of the impulses N_P is counted during in $N(\geq N_P)$ internal time steps where the initial spin has been given. The number N_P is obtained for examples; (1) $N_P = 0$ for the ferromagnetic sequence, (2) $N_P \simeq N/2$ for the random sequence, (3) $N_P = N$ for the anti-ferromagnetic sequence. The density of the impulses is considered to be proportional to the number N_P, and the value of the i−th neuron at the external time n is defined by

$$u_i(n) = \frac{1}{1 + \exp[-k\{N_{iP}(n) - N_0\}]}, \tag{15}$$

where k and N_0 are the parameters, and $N_{iP}(n)$ is the number of impulses of the i−th neuron at the external time n. Here we have assumed that each external time step is comprised of N internal time steps.

In the same manner as the second model, the internal control parameter of the i-th neuron r_i is renewed by

$$r_i(Nn) = \sum_j w_{ij} u_j(n) + s_i - \theta_i. \qquad (16)$$

There are many cases of the parameter set $\{N, N_0, k, \theta_i\}$ which bring us good results. When we take $N = 10$ and $k = 5.0$, for example, self association can be easily carried out with the parameters; (1) $\theta_i = -2.01$ (near the ferromagnetic transition point), $N_0 = 0.5 \sim 6.0$, (2) $\theta_i = -2.76$ (the random state), $N_0 = 0.5 \sim 7.5$, (3) $\theta_i = -2.93$ (near the anti-ferromagnetic transition point), $N_0 = 0.5 \sim 9.5$. If values of the parameters are taken as $\{N = 8, N_0 = 7, k = 3.6, \theta_i = -2.88\}$, the model shows good behavior for solving a TSP.

3 Mechanisms and properties

One of the essential mechanisms of our models is that the control parameter of the neuron (microscopic element) is changed its value by the macroscopic state $\{u_j(n)\}$ through the weights $\{w_{ij}\}$. On the other hand, the macroscopic state is obviously determined by the set of neuron's states. Namely, the *microscopic rule* (the parameter of the neuron) is subject to the *macroscopic state* by means of a kind of feedback which is expressed by Eq.4, Eq.9 or Eq.16. This mechanism has been named as *feedback slaving principle* [3] which is different from Haken's slaving principle [9] where there is no such feedback. Each neuron appropriately changes its state depends on the macroscopic state owing to the principle, and the same time a new macroscopic state is produced by the set of renewed neurons. One of the solutions is founded by the model when its macroscopic state becomes steady. A similar mechanism can be observed in the *reference circulation*, where a meaning of a sentence is feedback to its words, if we consider that words and a sentence correspond to the neurons and the model, respectively. Furthermore, a *society* (macro) and its *people* (micro) metaphorically obey the principle.

Chaos dynamics inside the neuron also plays an extremely important role. Our models run on deterministic rules, but they have ability of stochastic search. The stochastic property is caused by *chaos noise* which is not a white noise but highly intermittent. The noise brings about *self annealing*, and a simulated annealing is not necessary for solving an optimization problem [1, 2, 3] or Boltzmann machine scheme learning [4].

Motion of a synchronous Hopfield model falls into a fix point or a two-state cycle. On the contrary, our model is synchronous one, however it runs properly owing to *transient motion* of the oscillator(s) in the neuron. This synchronous parallel processing in the level of elements is very benefit for speed up the computation if it take into account actually. A coupled nonlinear LCR circuit (analog) [10] and a chaotic chip (digital) [11] may be useful to realized a hardware of the model, which can be called as a *chaos neuro-computer.*

4 Functions of the models

The first model has efficient functions such as self association [1], solving an optimization problem (TSP) [1, 3], a Boltzmann machine scheme learning [4] and a back propagation learning [5]. Whereas each conventional neural model has only one or two of them. In the following we compare our model with the conventional models:

1. **Hopfield model** [12]
   The neuron is the *binary* one and its state is changed *asynchronously* by a *deterministic* rule. The "energy" of the model always decreases in scales of 1 or 0 Hamming distance per one time step, therefore it easily falls into a trap of local minimum. As a result of this property, the model has a function of self association, however it can scarcely solve an optimization problem.

2. **Hopfield-Tank model** [13]
   Analog neurons are governed by *deterministic* differential equations in the model. It can solve TSP in general, however the result depends on its initial condition [14].

3. **Boltzmann machine** [15]
   The neuron is the *binary* one and its state is *asynchronously* changed by a *stochastic* rule. The model has functions of solving an optimization problem and Boltzmann machine learning. However, simulated annealing processes are necessary, and these spend long time steps.

4. **The chaos neuron model**
   The neuron has a dual structure, namely the output $u_i(n)$ is *digital* but variable of the coupled-oscillator in the neuron $x_i(n), y_i(n)$ and $D_i(n)$ take *analog* values. It runs on *deterministic* rules, but it has the ability of *stochastic* search without *annealing*. And the *synchronous* update of all neurons enables to change a state of the model with long Hamming distance in one step. It has all functions which are shown in the above models, and also ability of back propagation learning is confirmed.

A TSP has been solved using the first model [1] and also the analog version of it [3]. Here we introduce a new analog neuron and apply it to solve the TSP of 10 cities whose coordinates are the same as those of Hopfield-Tank [13]. The value of the neuron is determined by Eq.(6), where a new $z_i(n)$ is used which is defined by

$$z_i(n) \equiv \frac{\epsilon}{\Delta_i(n)} - \frac{A_z}{1 - \Delta_i(n)}, \qquad (17)$$

where A_z is the parameter.

In ref.[3], analog values have been translated into binary values, which are necessary to find a solution, with the aid of the boundary value $u_b = 0.5$. The meaning of the boundary value u_b is that $u_i(n) = 0$ if the analog value takes $u_i(n) < u_b$ and $u_i(n) = 1$ if the analog value takes $u_i(n) \geq u_b$. The boundary value u_b and the threshold value θ_i are regarded as the parameters in the present simulation and the other methods are the same with ref.[3]. We have carried out 100 runs of 1000 cut off time steps where the parameters are chosen as $\{\epsilon = 0.0000001, a = 4, b = (a - 50\epsilon), 1/z_0 = 5.5, A_z = 1.0, u_b = 0.05, \theta_i = -2.4\}$ and the parameters of the "energy" terms A and B in ref.[1] are taken as $\{A = 2.5, B = A/0.9\}$. The results obtained are that all (100%) runs find valid tours which include 60 (60%)

best solutions whose distance is 2.69. The average distance and the variance of the results (all possible tours) are 2.73 (4.77) and 0.01 (0.25), respectively. The above results are very good, however, the parameter set and the analog function may be not the best choice, more efficient results could be obtained if we select the best choice.

Very recently, a mechanism similar to our feedback has been incorporated into Aihara's model [16] by Nozawa [17] who has obtained efficiently the best solution of the TSP using the model. However he reduced the problem with the help of an external condition, and it may not generally apply to other problem [3].

The first model has a superior ability of Boltzmann machine scheme learning [4].A back propagation learning of the logical operation XOR in the model has been studied in [5]. We simulate the same back propagation learning in the analog version of the model. The method of the learning is the same as the conventional one but $f_s(DD_{Hi} - 0.5)$ and $f_s(DD_{Ok} - 0.5)$ are used instead of H_j (out put value of j-th hidden neuron) and O_k (out put value of k-th out put neuron), respectively. Where $f_s(x)$ is the sigmoid function $f_s(x) = 1/[1 + \exp(-2x/u_0)]$. In the learning, after 20 time steps, the model is assumed to be close to equilibrium, and then the learning signals are calculated. One of the simulations using the analog neurons $\{\epsilon = 0.005, a = 4, b = (a - \epsilon), 1/z_0 = 5.0\}$ is shown in Fig.1, where the sigmoid parameter u_0, the weight learning parameter α, the threshold value learning parameter β are chosen as $\{u_0 = 1.0, \alpha = 0.8, \beta = 0.4\}$. The results are better than those of the binary model in general.

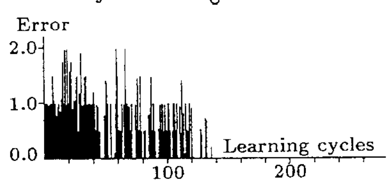

Fig.1 Learning process in our network.

We find that there are two typical neuron's parameter sets which are (A) $\{\epsilon \approx 0.0000001, (a - b) \approx 50\epsilon\}$, and (B) $\{\epsilon \approx 0.001, (a - b) \approx 5\epsilon\}$. In the case (A), the model behaves similar to Hopfield model. The set is good for self association and solving an optimization problem such as TSP, but learning is slow. On the other hand, the model of the case (B) has all functions and it behavior is similar to Boltzmann machine.

The second and the fourth models have similar functions with the first model, but their performance are not so good as that of the first model.

The third model has a distinctive feature whose neuron can take one of $\{1, 0, -1\}$. We illustrate self association in Nakano's Associatron [18] which is realized by the third model. The three patterns (1,4,8) are retrieved from their initial patterns which are shown in Fig.2 where the control parameters of the logistic maps $\{a, b, c\}$ and other parameters are chosen as $\{a = 4, b = 3.9995, c = 3.9992, \epsilon = 0.00001, m = 15\}$. A weakly deformed pattern is necessary at the initial time for self association in binary models, however only small part of pattern is enough for the Associatron where unknown information is represented by the neurons with $\{u_i = 0\}$. Mutual association can be easily carried out in the Associatron with the aid of the robust retrieval.

Acknowledgment. The author wishes to thank Messrs. A.Nagayoshi, S.Fukushima, H.Kawamura, K.Nakamoto and J.Omachi for their assistance with the computer simulations.

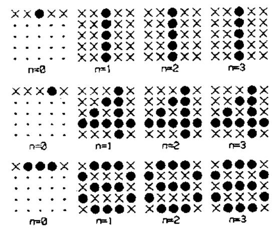

Fig.2 Retrieval from their initial patterns.

References

[1] M. Inoue and A. Nagayoshi, Phys. Lett. **A 158** (1991) 373.

[2] M. Inoue and S. Fukushima, Prog. Theor. Phys. **87** (1992) 771.

[3] M. Inoue and A. Nagayoshi, Prog. Theor. Phys. **88** (1992) 769.

[4] M. Inoue and A. Nagayoshi, *Proc. International Conference on Fuzzy Logic and Neural Networks*, Iizuka Japan, (1992) 559.

[5] M. Inoue and A. Nagayoshi, *Proc. International Symposia on Information Sciences; Neural Information Processing*, Iizuka Japan, (1992) 78.

[6] T. Yamada and H. Fujisaka, Prog. Theor. Phys. **70** (1983) 1240.

[7] M. Inoue, H. Anai, H. Hata and H. Fujisaka, Prog. Theor. Phys. **87** (1992) 103.

[8] M. Inoue, K. Tanabe and H. Fujisaka, Prog. Theor. Phys. **81** (1989) 1135.

[9] H. Haken, *Synergetics* (Springer-Verlag, Berlin, 1978).

[10] K. Fukushima and T. Yamada, J. Phys. Soc. Jpn. **57** (1988) 4055.

[11] T. Yamakawa, T. Miki and E. Uchino, *Proc. International Conference on Fuzzy Logic and Neural Networks*, Iizuka Japan, (1992) 563.

[12] J. J. Hopfield, Proc. Natl. Acad. Sci. USA **79** (1982) 2554.

[13] J. J. Hopfield and D. W. Tank, Biol. Cybern. **52** (1985) 141.

[14] G. V. Wilson and G. S. Pawley, Biol. Cybern. **58** (1988) 63.

[15] G. E. Hinton, D. H. Ackley and T. J. Sejnowski, Tech. Rep. CMU-CS-84-119, Carnegie Mellon Univ., (1984).

[16] K. Aihara, T. Takabe and M. Toyoda, Phys. Lett. **A144** (1990) 333.

[17] H.Nozawa, CHAOS, **2** (1992) 377.

[18] K. Nakano, *IEEE Trans.* **SMC-2** (1972) 380.