Crossbar Switch Problem Solver by Hysteresis Neural Networks

Wei Zhang† and Zheng Tang††,

Faculty of Engineering, Toyama University, Toyama-Shi, JAPAN

Summary
In this paper, we propose a continuous hysteresis neurons (CHN) Hopfield neural network architecture for efficiently solving crossbar switch problems. A Hopfield neural network architecture with continuous hysteresis and its collective computational properties are studied. It is proved theoretically and confirmed by simulating the randomly generated Hopfield neural network with CHN. The network architecture is applied to a crossbar switch problem and results of computer simulations are presented and used to illustrate the computation power of the network architecture. The simulation results show that the Hopfield neural network architecture with CHN is much better than the binary hysteresis Hopfield neural network architecture for crossbar switch problem in terms of both the computation time and the solution quality.

Key words:
Network architecture, crossbar switch problem, continuous hysteresis, Hopfield neural network

1. Introduction

The auto associative memory model proposed by Hopfield [1,2] has attracted considerable interest both as a content address memory (CAM) and, more interestingly, as a method of solving difficult optimization problems [3-5]. The Hopfield neural network contain highly interconnected nonlinear processing elements ("neurons") with two-state threshold neurons [1] or graded response neurons [2]. Takefuji and Lee proposed a two-state (binary) hysteretic neuron model to suppress the oscillatory behaviors of neural dynamics [6]. However, Tateshi and Tamura showed Takefuji and Lee's model did not always guarantee the descent of energy function [7], Wang also explained why the model may lead to inaccurate results and oscillatory behaviors in the convergence process [8]. Since their report, several modifications on the hysteretic function, for example Galán and Muñoz's binary [9] and Bharitkar and Mendel's multivalued [10] hysteretic functions. Xia proposed a new algorithm that the Hopfield neural network with the binary hysteresis and claimed that the solution quality found by their algorithm was superior to that of the best existing parallel algorithm [11].

Since Hopfield and Tank's works, there has been a lot of interest in the Hopfield neural networks because its advantage over other approaches for solving optimization problems. The advantages include massive parallelism, convenient hardware implementation of the neural network architecture, and a common approach for solving various optimization problems [12]. The Hopfield neural network architecture has also been applied to real-time control of a crossbar switch used for switching high-speed packets at maximum throughput [13-19], and shown to be capable of achieving very good performance especially for small size crossbar switch problems.

In this paper, we introduce a new Hopfield neural network algorithm for efficient solving crossbar switch problem. Different to the binary hysteresis Hopfield neural network , our architecture uses continuous hysteresis neurons (CHN). We prove theoretically that the emergent collective properties of the original Hopfield neural network also are present in the Hopfield network with continuous hysteresis neurons (CHN). Simulations of randomly generated neural networks are performed on both networks and show that the Hopfield neural networks with CHN have the collective computational properties like the original Hopfield neural networks. What a more, the Hopfield neural networks withCHN converges faster than the Hopfield neural networks with the binary hysteresis neurons do. In order to see how well the Hopfield neural networks with CHN do for solving practical combinatorial optimization problems, a large number of computer simulations are carried out for the crossbar switch problem. The simulation results show that the Hopfield neural network architecture with CHN is much better than the previous works including the Hopfield neural network with hysteresis binary neurons for the crossbar switch problem in terms of both the computation time and the solution quality.

2. Binary Hysteresis Neurons Hopfield Network

Like the original Hopfield neural networks, the total input to neuron *i* of the Hopfield neural networks with hysteresis binary neuron is:

$$x_i = \sum_{j \neq i} w_{ij} y_j + h_i \qquad (1)$$

where x_i is the total input of neuron *i*, y_j is the output of neuron *j*. The element w_{ij} is the symmetric interconnection strength from neuron *j* to neuron *i*, and h_i is the offset bias of neuron *i*.

Manuscript received January 5, 2010
Manuscript revised January 20, 2010

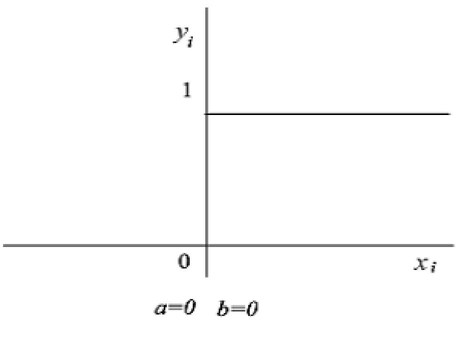
Each neuron samples its input at random times. But, unlike the Hopfield neural network's two-state threshold neurons (Fig.1 (a)), the hysteresis binary neurons change the value of their output or leave them fixed according to a hysteretic threshold rule (Fig.1 (b))

$$y_i = \begin{cases} 1 & if\ x_i > a \\ 0 & if\ x_i < b \\ no\ change & if\ b \le x_i \le a \end{cases} \quad (2)$$

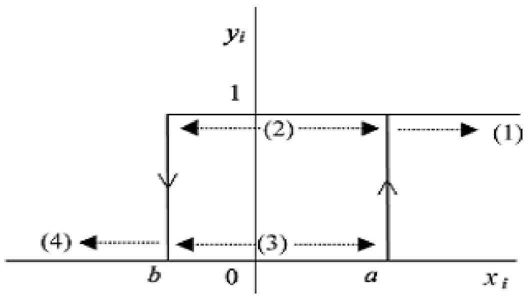
where a is the upper trip point (*UTP*), and b is the lower trip point (*LTP*). As shown in Fig.1, if $x_i>a$, then $y_i=1$ and if $x_i<b$, then $y_i=0$. When $b \le x_i \le a$, y_i keeps unchanged, i.e., $y_i=1$ if the last y_i was 1 and $y_i=0$ if the last y_i was 0.

Consider the erengy:

$$E = -\frac{1}{2}\sum_i \sum_{j \ne i} w_{ij} y_i y_j - \sum_i h_i y_i \quad (3)$$



(a)



(b)

Fig.1 Hysteresis binary functions

3. Hopfield Network with Continuous Hysteresis Neurons(CHN)

For the original Hopfield neural networks, let the output variable y_i for neuron i have the range $y_i^0 \le y_i \le y_i^1$ and be an continuous and monotone-increasing function of the instantaneous input x_i to neuron i. The typical input-output relation $g_i(x_i)$ $y_i = 1/(1+e)$ shown in Fig.2(a) is sigmoid with asymptotes y_i^0 and y_i^1. as,

$$y_i = g(x_i) = 1/(1+e^{-r(x_i-\theta)}) \quad (4)$$

Where r is the gain factor and θ is the threshold parameter.

In biological system, x_i will lag behind the instantaneous outputs y_i of the other cells because of the input capacitance C of the cell membranes, the transmenmbrane resistance R, and the finite impedance w_{ij}^{-1} between the output y_i and the cell body of cell i. Thus there is a resistance-capacitance(*RC*) charging equation that determines the rate of change of x_i.

$$\begin{aligned} C_i\left(\frac{dx_i}{dt}\right) &= \sum_{j=1}^{N} w_{ij} y_j - \frac{x_i}{R_i} + I_i \quad i = 1,2,\cdots,N \\ x_i &= g_i^{-1}(y_i) \end{aligned} \quad (5)$$

where C_i is the total input capacitance of the amplifier i and its associated input lead. $w_{ij}y_j$ represents the electrical current input to cell i due to the present potential of cell j, and w_{ij} is thus the synapse efficacy. I_i is any other (fixed) input current to neuron i. In terms of electrical circuits, $g_i(x_i)$ represents the input-output characteristic of a nonlinear amplifier with negligible response time. It is convenient also to define the inverse output-input relation, $g_i^{-1}(y_i)$.

In order to improve the solution quality of crossbar switch problem, we proposed a new neural network method for efficiently solving the crossbar switch problem. In this method, a continuous hysteresis neuron(CHN) is applied to the Hopfield neural network.

The hysteresis continuous neurons change the value of their output or leave them fixed according to a hysteretic threshold rule (Fig.2 (b)). Mathematically, hysteretic neuron function is described as:

$$y(x_i(t)\big|\dot{x}_i(t-\delta t)) = g\left[x_i(t) - \theta(\dot{x}_i(t-\delta t))\right] \quad (6)$$

where

$$\gamma(\dot{x}_i(t-\delta t)) = \begin{cases} \gamma_\alpha, & \dot{x}_i(t-\delta t) \ge 0 \\ \gamma_\beta, & \dot{x}_i(t-\delta t) \le 0 \end{cases} \quad (7)$$

And

$$\theta(\dot{x}_i(t-\delta t)) = \begin{cases} -\alpha, & \dot{x}_i(t-\delta t) \ge 0 \\ \beta, & \dot{x}_i(t-\delta t) \le 0 \end{cases} \quad (8)$$

$\beta > \alpha$, and $(\gamma_\alpha, \gamma_\beta) > 0$,and $\dot{x}_i(t-\delta t) \underline{\underline{\Delta}} \frac{dx_i(t-\delta t)}{dt}$. Thus, there is a resistance-capacitance(RC) charging equation that determines the rate of change of x_i.

$$C_i\left(\frac{dx_i}{dt}\right) = \sum_{j=1}^{N} w_{ij} y_j - \frac{x_i}{R_i} + I_i + \frac{\theta_i(\dot{x}_i(t-\delta t))}{R_i} \quad i = 1,2,\cdots,N \quad (9)$$

$$x_i = g_i^{-1}(y_i)$$

Consider the energy:

$$E = -\frac{1}{2}\sum_{i=1}^{N}\sum_{j=1}^{N} w_{ij} x_i y_j + \sum_{i=}^{1} \frac{1}{R_i}\int_0^{-1} g_i^{-1} \times \left[y_i(x_i \middle| \dot{x}_i) \right] dy_i + \sum_{i=1}^{N} I_i y_i \quad (10)$$

Its time derivative for a symmetric W is:

$$\frac{dE}{dt} = -\sum_{i=1}^{N} \frac{dy_i}{dt}\left(\sum_{j=1}^{N} w_{ij} y_j - \frac{x_i}{R_i} + I_i + \frac{\theta_i(\dot{x}_i(t-\delta t))}{R_i}\right) \quad (11)$$

The parenthesis is the right-hand side of Eq.9, so

$$\frac{dE}{dt} = -\sum_{i=1}^{N} C_i\left(\frac{dy_i}{dt}\right)\left(\frac{dx_i}{dt}\right) = -\sum_{i=1}^{N} C_i g_i^{-1'}(y_i)\left(\frac{dy_i}{dt}\right)^2 \quad (12)$$

Since $g_i^{-1}(y_i)$ is a monotone increasing function and C_i is positive, each term in this sum is nonnegative. Therefore:

$$\frac{dE}{dt} \le 0, \quad \frac{dE}{dt} = 0 \rightarrow \frac{dy_i}{dt} = 0 \quad \text{for all } i \quad (13)$$

Together with the boundedness of E, Eq.(9) shows that the time evolution of the system is a motion in state space that seeks out minima in E and comes to a stop at such points. E is a Liapunov function for the system.

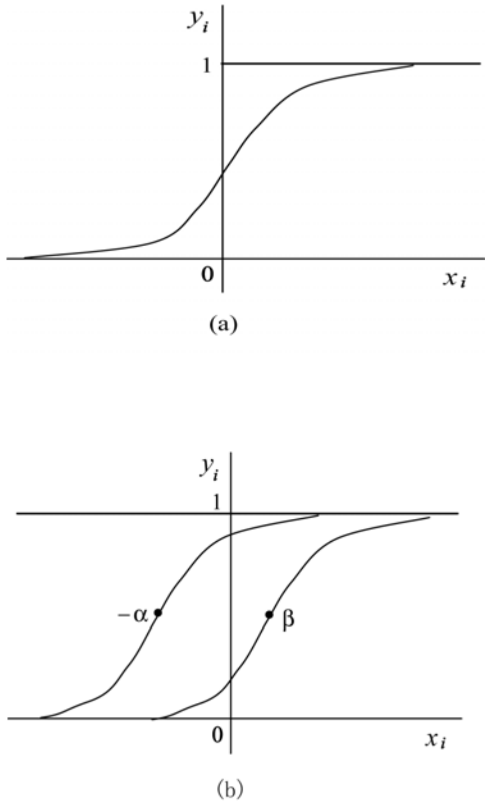


Fig.2. Hysteresis functions

4. Comparison of two Hysteresis Neurons

The continuous hysteresis neurons (CHN) Hopfield neural network and the binary hysteresis neurons Hopfield neural network are two kinds of the hysteresis network. The two algorithms differ from other ways:
it 1) is multivalued; 2) has memory; and 3) is adaptive.
In the Eq.5, if the value of $e^{-\gamma(x_i-\theta)}$ get to be infinite,the output variable $y_i = g(x_i) = 1/(1+e^{-r(x_i-\theta)})$ will be close to "0", and if the value of $e^{-\gamma(x_i-\theta)}$ get to the infinitesimal, it will be close to "1".
In this way, we can consider the binary hysteresis neuron is a special case of continuous hysteresis neuron (Fig.3). The algorithms that how to update the network can be discussed with the application to the crossbar switch problem.

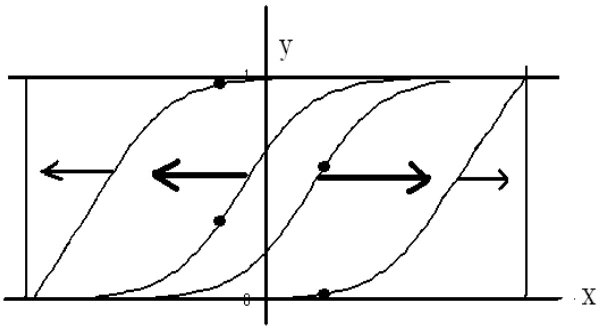


Fig.3. hysteretic activation function

5. Application to Crossbar Switch Problem

The problem of maximizing the throughput of packets through a crossbar switch is best described by referring to Fig. 2, which shows how requests to switch packets through an $N \times N$ crossbar switch can be represented by an $N \times N$ binary request matrix R [12][13]. Rows and columns of the matrix R are associated with inputs and outputs, respectively, of the crossbar switch. A matrix element $r_{ij} = 1$ indicates that there is a request for switching at least one packet from input line i to output line j of the switch; otherwise $r_{ij} = 0$. If we consider the crossbar switch for point-to-point connections, then at most one crosspoint may be closed on any row or column of the switch during packed transmission. The state of the switch can be represented by an $N \times N$ binary configuration matrix C, where $c_{ij} = 1$ indicates that input line i is connected to output line j by the "closed" crosspoint (ij). $c_{ij} = 0$ indicates that crosspoint (ij) is "open". For proper operation of the switch, there should be at most one closed crosspoint in each row and each column. The throughput of the switch is optimal when the matrix C, which is a subset of the matrix R (i.e., $c_{ij} \le r_{ij}$ for every (i, j)), contains at most a "1" in each row/column, and has maximum overlap with R. Examples of optimal matrices are shown in Fig.4 for a 4×4 crossbar.

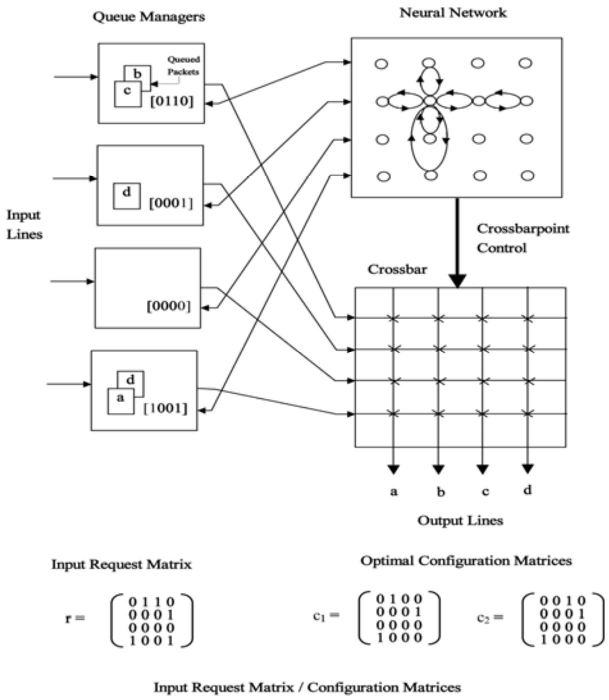


Fig.4. Schematic architecture of 4×4 crossbar control with an example of input request matrix and its optimal configuration matrices

Each switch inlet has a queue manager. When an inlet queue manager receives a packet, it examines the packet's destination address and determines its switch outlet. It then updates the row request vector for that inlet by setting to "1", the bit corresponding to the switch outlet, and places the packet on the inlet queue. The crossbar switch is controlled by a neural network that has one neuron in correspondence to each switch crosspoint. Row request vectors from all the inlets are supplied to the neural network, which uses them to compute an optimal configuration matrix for the switch. The resulting row configuration vectors are then returned to the corresponding queue manager, while the crossbar switch crosspoints selected by the computed configuration matrix are closed. Each queue manager presents to its inlet a single packet destined to the outlet selected by the row vector returned by the neural network, which thus gets routed through the closed crosspoint to its proper outlet. The queue manager also updates its row request vector by clearing the selected column bit, provided that no packets remain queued for that output. This process is cyclical: new packets are received while queued packets are being transmitted. If all packets were of a constant length, then it would be possible to receive new packets, transmit selected packets, and compute the next configuration in parallel. The computation of an optimal configuration matrix should be completed in a few microseconds, which is less than it takes to transmit a packet in a high-speed fiber optic based communication system [17][18].

The crossbar switch problem can be solved by constructing an appropriate energy function and minimizing the energy function to zero (*E*=0) using an $N \times N$ two-dimensional Hopfield neural network, i.e., a matrix $V = [y_{ij}]$.

The objective energy function of the crossbar switch problem is given by [18]:

$$E = \frac{A}{2}\sum_{i=1}^{N}\left(\sum_{k=1}^{N} y_{ik} - 1\right)^2 + \frac{B}{2}\sum_{j=1}^{N}\left(\sum_{k=1}^{N} y_{kj} - 1\right)^2 \quad (14)$$

where A and B are coefficients, y_{ik} is the output value of neuron *ik* and y_{kj} is the output value of neuron *kj*. The first term will be zero if each row contains no more than one "1", with all the other values being zero. Similarly, the second term is zero if each column contains no more than one "1". We can get the total input (x_{ij}) of neuron by using the partial derivation term of the energy function. Thus, the total input (x_{ij}) of neuron is given:

$$x_{ij} = -A\left(\sum_{k=1}^{N} y_{ik} - 1\right) - B\left(\sum_{k=1}^{N} y_{kj} - 1\right) \quad (15)$$

For a given $N \times N$ matrix R , i.e., an $N \times N$ matrix V , updating the matrix element y_{ij} by Eqs. (5)(6)(7) and (15) can produce a new matrix V , which is either an optimal or a local minimal matrix, i.e., a subset of the matrix V or R by the convergence characteristics of the Hopfield networks with continuous hysteresis neurons.

5.Simulation Results

Since our architecture is valid for any Hopfield neural networks as illustrated in the previous section, we used the architecture for some randomly generated problems and a large number of real crossbar switch problems up to 1000×1000 switches.

Experiments were first performed to show the convergence of the Hopfield neural networks with continuous hysteresis neurons. In the simulations, a 100-neuron Hopfield neural network with continuous hysteresis neurons (i = 1, 2, …, 100) was chosen. Initial parameters of the network, connection weights and thresholds were randomly generated uniformly between –1.0 and 1.0. Simulations on a randomly generated 100-neuron Hopfield network with different value of α and β ($\alpha = \beta = 0$ and $\alpha = 0.6, \beta = 0.6$ for i =1, …, 100) were also carried out. Fig.5 shows the convergence charactoristics of both networks. From this figure we can see that both the Hopfield neural networks with continuous hysteresis neurons ($\alpha = 0.6, \beta = 0.6$) converged to stable states that did not further change with time. It is worth to note that the Hopfield neural network with continuous hysteresis neurons ($\alpha = 0.6, \beta = 0.6$) seek out a smaller minimum at seek out $E = -158.87$ than the Hopfield neural network without hysteresis neurons at $E = -127.85$ ($\alpha = \beta = 0$).

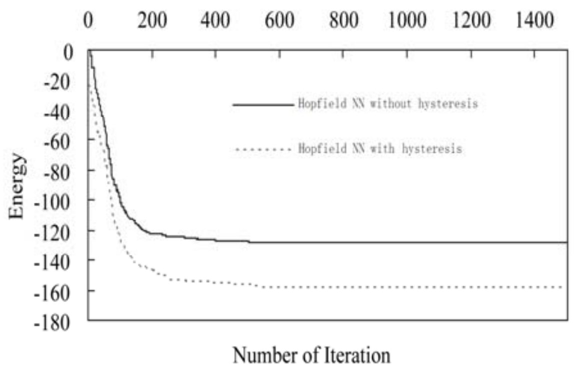


Fig. 5 The convergence characteristic of a 100-neuron Hopfield network with and without the continuous hysteresis neurons .

The Hopfield neural network with continuous hysteresis neurons were also applied to crossbar switch problem and simulations were also carried out. The proposed algorithm was implemented in C++ on PC Station (PentiumIIII 3GHz). A sigmoid function was used as intput/output function and the temperature parameter $\gamma_\alpha = \gamma_\beta$ was set to 0.03.

We now present simulation results of the architecture when applied to real crossbar switch problem. The parameters *A* and *B* were set to *A*=1, *B*=0.5. The weights and external input currents were all the same as the original Hopfield neural network. Continuous hysteresis neurons was set to $\alpha = 0.6, \beta = 0.6$. The network architectures for 14 kinds of instances from 4×4 to

1000×1000 crossbar switches were simulated on a digital computer. In simulations, 100 simulation runs with different randomly generated initial states were performed on each of these instances. Eq.(5)(6)(7) were used as the input/output function. Using Eq.(15), all neurons were computed exactly once in one iteration step. The maximum iteration step was set to 1000. When the iteration steps exceeded the maximum iteration step, the iteration was terminated.

In Table.1, the column labeled "optimal" is the global convergence times among 100 simulations and the column labeled "steps" is the average number of iteration steps required for the convergence in the 100 simulations. The simulation results show that the architectures with continuous hysteresis neurons (CHN) could almost find optimum solution to all crossbar switch problems within short computation times.

Table 1. Computational results.

Crossbar Switches	Hysteresis Binary Hopfield		Proposed Architecture	
	Optimal(%)	Steps	Optimal(%)	Steps
4×4	100	2	100	2
6×6	100	3	100	2
8×8	100	4	100	2
10×10	100	6	100	2
20×20	100	5	100	2
30×30	100	5	100	2
50×50	100	5	100	2
80×80	100	5	100	2
100×100	100	5	100	2
200×200	100	5	100	2
300×300	100	5	100	2
500×500	100	5	100	2
800×800	100	5	100	2
1000×1000	100	5	100	2

We compared our results with that found by the binary hysteresis Hopfield neural network [11]. Table 1 shows the results by the two different networks: the binary hysteresis Hopfield neural network and our network architecture, where the convergence rates and the average number of iteration steps required for the convergence are summarized. From Table.1, we can see that the Hopfield neural network architecture with continuous hysteresis neurons (CHN) was very effective, and was better than the binary hysteresis Hopfield neural network in terms of the computation time and the solution quality for crossbar switch problem. Further, the number of iteration steps of our architecture was almost independent on the problem size, while that the binary hysteresis Hopfield neural network was three steps more than our network architecture when we reach optimum solution to the large size.

6.Conclusions

We have proposed a continuous hysteresis Hopfield neural network architecture for the crossbar switch problem, and showed its effectiveness by simulation experiments. The proposed architecture was based on a modified Hopfield neural network in which the continuous hysteresis neurons(CHN) were added to improve solution quality. We proved theoretically that the emergent collective properties of the original Hopfield neural network also were present in the Hopfield neural network architectures with continuous hysteresis neurons (CHN). In order to verify the performance of the proposed architecture, we have tested the architectures with a large number of randomly generated examples and crossbar switch problems up to 1000×1000 switches. The simulation results showed that the Hopfield neural network architecture with continuous hysteresis neurons(CHN) was much better than the previous works including the Hopfield neural network with hysteresis binary neurons for crossbar switch problem in terms of both the computation time and the solution quality.

References

[1] J.J. Hopfield, " Neural network and physical systems with emergent collective computational abilities," Proc. Natl. Acad. Sci. USA. Vol.79 pp.2554-2558, 1982.
[2] J.J. Hopfield, " Neurons with graded response have collective computational properties like those of two-state neurons," Proc. Natl. Acad. Sci. USA. Vol.81 pp.3088-3092, 1984.
[3] J.J. Hopfield and D.W.Tank, " 'Neural' computation of decisions in optimization problems," Biol. Cybern., Vol.52 pp.141-152, 1985.
[4] J.J. Hopfield and D.W.Tank, " Computing with neural circuits: a model," Science, no.233, pp.625-633,1986.
[5] D.W. Tank and J.J. Hopfield, "Simple neural optimization network: An A/D converter, signal decision circuit, and linear programming circuit," IEEE Trans, Circuits & Systems, vol.CAS-33, no.5, pp.533-541,1986.
[6] Y. Takefuji and K.C. Lee, "An hysterisis binary neuron: A model suppressing the oscillatory behavior of neural dynamics," Biol. Cybern., Vol.64 pp.353-356, 1991.
[7] M. Tateishi and S. Tamura, "Comments on `Artificial neural networks for four-coloring map problems and K-colorability problems'," IEEE Trans. Circ. Syst.-I: Fundamental Theory and Applications, vol. 41, no. 3, pp. 248-249, March, 1994.
[8] L. Wang, "Discrete-time convergence theory and updating rules for neural networks with energy functions," IEEE Trans. Neural Networks, vol.8, no.5, pp. 445-447, 1997.
[9] G. Galán-Marín, J. Muñoz-Pérez, "A new input-output function for binary Hopfield neural networks," Proceedings of the International Work-Conference on Artificial and

Natural Neural Networks: Foundations and Tools for Neural Modeling, pp.311-320, June 02-04, 1999.
[10] S. Bharitkar, J. M. Mendel, "The hysteretic Hopfield neural network," IEEE Trans. Neural Networks, vol.11, no.4, pp. 879 888, 2000.
[11]Guangpu Xia, Zheng Tang, Ronglong Wang and Yong Li, "Hopfield Neural Network with Hysteresis for maximum Cut Problem," Neural Information Processing - Letters and Reviews, Vol.4, No.2, pp. 19-26 ,2004.
[12] X.C.Zeng and T. Martinez, " A new relaxation procedure in the Hopfield neural networks for solveing Optimization Problems, " Neuron Processing Letters, 10, pp.211-222 ,1999.
[13] S. Matsuda, "Capabilities of hopfield neural networks for searching the optimal solution to n-rook problem and crossbar switch problem (in Japanese)," Trans. of Inst. Of Electronics, Information and Communication Engineers, J80-D-II, No. 9, pp. 2549-2555 ,1997.
[14] S. Matsuda," Guaranteeing efficiency and safety of a hopfield neural network for crossbar (in Japanese)," Trans. Of Inst. Of Electronics, Information and Communication Engineers, J80-D-II, No. 12, pp. 3200-3209 ,1997.
[15] M.M. Ali and H.T. Nguyen,"A neural network implementation of an input access scheme in a high-speed packet switch," IEEE Global Telecommunications Conference and Expo, pp. 1192-1196 ,1989.
[16] R.J.T. Morris and B. Samadi," Neural network control of communications systems," IEEE Trans. on Neural Networks, 5, No. 4, pp. 639-650 ,1994.
[17] A. Marrakchi and T. Troudet," A neural net arbitrator for large crossbar packet-switches," IEEE Trans. On Circuits and Systems, 36, No. 7, pp. 1039-1041 ,1989.
[18] T.P. Troudet and S.M. Walters,"Neural network architecture for crossbar switch control," IEEE Trans. On Circuits and Systems, 38, No. 1, pp. 42-56 ,1991.
[19] Y. Takefuji," Neural Network Parallel Computing," Kluwer Academic Publishers ,1992.

Wei Zhang received a B.S. degree from Zhejiang Normal University, Zhejiang, China in 2004. She received an M.S. degree from Toyama University, Toyama, Japan, in 2007. Form 2007 she is working toward the.D.E. degree at Toyama University, Japan. Her main research interests are neural networks and optimization problems.

Zheng Tang received a B.S. degree from Zhejiang University, Zhejiang, China in 1982 and an M.S. degree and a D.E. degree from Tsinghua University, Beijing, China in 1984 and 1988, respectively. From 1988 to 1989, he was an Instructor in the Institute of Microelectronics at Tsinghua University. From 1990 to 1999, he was an Associate Professor in the Department of Electrical and Electronic Engineering, Miyazaki University, Miyazaki, Japan. In 2000, he joined Toyama University, Toyama, Japan, where he is currently a Professor in the Department of Intellectual Information Systems. His current research interests include intellectual information technology, neural networks, and optimizations.