Implementation of a new neurochip using stochastic logic

doi: 10.1109/TNN.2003.816341

Implementation of a New Neurochip Using Stochastic Logic

Shigeo Sato, Ken Nemoto, Shunsuke Akimoto, Mitsunaga Kinjo, and Koji Nakajima

***Abstract*—Even though many neurochips have been developed and investigated, the best suitable way for implementation has not been known clearly. Our approach is to exploit stochastic logic for various operations required for neural functions. The advantage of stochastic logic is that complex operations can be implemented with a few ordinary logic gates. On the other hand, the operation speed is not so fast since stochastic logic requires certain accumulation time for averaging. But huge integration can be achieved and its reliability is high because all of operations are done on digital circuits. Furthermore, we propose a nonmonotonic neuron realized by stochastic logic, since the nonmonotonic property is efficient for the performance enhancement in association and learning. In this paper, we show the circuit design and measurement results of a neurochip comprising 50 neurons are shown. The advantages of nonmonotonic property and stochasticism are shown clearly.**

***Index Terms*—Boltzmann machine (BM), large-scale integration (LSI) implementation, nonmonotonic neuron, simulated annealing, stochastic logic, traveling salesman problem.**

I. Introduction

IT is expected that a neural network is applicable in the fields such as pattern recognition and classification problems. It is not easy to estimate the suitable size of a network for practical use since it depends on a target problem very closely. However, rough estimation indicates that networks, comprising about 10^3 neurons, are necessary for practical applications [1]. Even by using modern large-scale integration (LSI) technology, it is not easy to implement such a large number of neurons. In addition, difficulties related to downscaling devices due to quantum effects have been reported. Then it is important to consider a high-performance neural network with limited resources. It has been reported that the performance both for association and learning are improved with nonmonotonic neurons rather than with monotonic neurons [2]–[9]. These results indicate that it is possible to solve various problems with less neurons. Since the major area of a neurochip is occupied by synapse circuits, the increasing area caused by incorporating nonmonotonic neurons is not serious in general case.

Various hardware neural networks have been proposed until now. They are categorized into two groups roughly, which are digital and analog circuits. Though analog implementations of artificial neural networks (ANNs) present the advantage of reduced silicon area as compared to their digital counterparts, its noise immunity is low and reliability is poor. On the other hand, though digital implementations present high reliability, the chip area is larger than analog implementations generally. This is because a large number of transistors is required. Therefore, marginal solutions utilizing pulse sequence have been proposed in order to overcome these disadvantages [10]–[17]. A neural-network utilizing stochastic logic, which has been proposed by Kondo *et al.* [13], is one of the pulse neural networks. One can convert analog quantity to pulse firing rate by stochastic logic, and various complex operations can be done with basic logic gates. For example, multiplication is done with a single AND gate. Also, nonmonotonic functions are realized by choosing suitable random number used for stochastic operation. Therefore, the number of transistors in a chip can be reduced greatly while the reliability is high. We show both circuit design and measurement results of a new stochastic logic neurochip comprising nonmonotonic neurons.

Manuscript received September 15, 2002.

The authors are with the Laboratory for Electronic Intelligent Systems, Research Institute of Electrical Communication, Tohoku University, Sendai 980-8577, Japan (e-mail: shigeo@riec.tohoku.ac.jp).

Digital Object Identifier 10.1109/TNN.2003.816341

II. Design of a Stochastic Neurochip With Nonmonotonic Neurons

A. Nonmonotonic Neurons

The advantage of a nonmonotonic neuron model was first suggested by Morita [2] in order to improve network performances of associative memory. It has been examined by numerical simulations and studied analytically by several researchers. Association property is improved compared with ordinary monotonic models, and the memory capacity increases. Yoshizawa *et al.* [3] reported that a certain piecewise linear nonmonotonic neuron can store $0.4N$ memory patterns. Fukai *et al.* [4] evaluated the enhancement of memory capacity for several nonmonotonic neurons by SCSNA. Yanai and Amari [5] also investigated memory capacity by statistical neurodynamics. Also the enhancement of learning property of nonmonotonic neurons has been studied by several researchers. Morita [7], [8] reported that nonmonotonic property is effective for storing sequential patterns. Usually, synaptic weights for an associative memory are obtained from correlations of desirable patterns. However, the modified patterns after applying nonmonotonic property to original patterns are superior to original patterns in terms of memory capacity of a nonmonotonic network. Kinjo *et al.* [9] studied the learning ability of a DBM network comprising nonmonotonic neurons by numerical simulations. They reported that the nonmonotonic DBM network requires less neurons for the parity problem. Since it is not easy to implement 10^3 neurons, utilization of nonmonotonic property is an important subject for practical applications.

Stochastic logic realizes pseudo analog operations using stochastically coded pulse sequence [13], [17], [18]. A coding

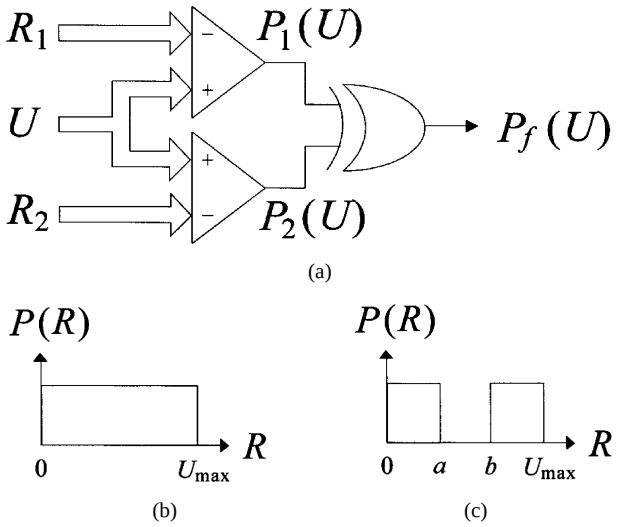


Fig. 1. (a) Nonmonotonic coding circuit, (b) uniform noise distributions, and (c) split noise distributions.

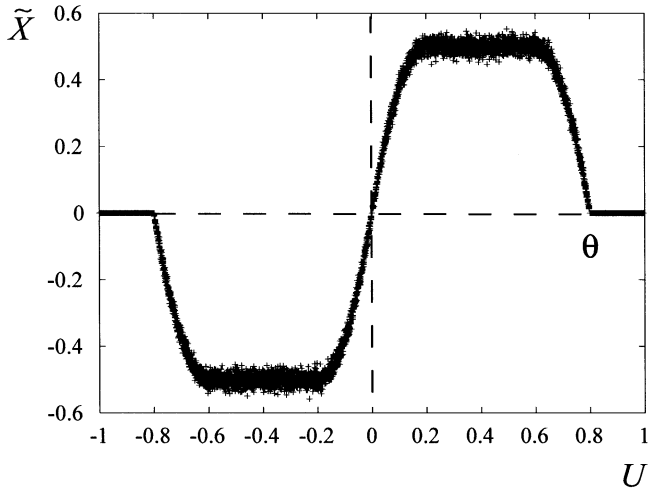


Fig. 2. Nonmonotonic function generated with a nonmonotonic coding circuit and two random numbers having the split distribution $(a = 0.2, b = 0.6, U_{\max} = 0.8, N_{\rm a} = 1000)$.

circuit, which is a simple digital comparator, encodes a digital input value in a stochastic pulse sequence. Let us consider a nonmonotonic neuron using stochastic logic as shown in Fig. 1(a). It comprises two ordinary coding circuits and an XOR gate. U is the membrane potential of a neuron and R_1 and R_2 are noises required for coding. The XOR gate outputs low if both two inputs are high. Therefore, the output firing probability $P_{\rm f}(U)$ decreases where U is sufficiently large. The circuit is compact enough, though two mutually independent uniform random numbers R_1 and R_2 are required. Since $P_1(U) = P_2(U) = U/U_{\max}$, then $P_{\rm f}(U)$ is given in a quadratic form as

$$\begin{aligned} P_{\rm f}(U) &= P_1(U)(1 - P_2(U)) + (1 - P_1(U))P_2(U) \\ &= \frac{2U}{U_{\max}}\left(1 - \frac{U}{U_{\max}}\right) \end{aligned} \tag{1}$$

where $U_{\max}$ is the maximum value of R_1 or R_2 as shown in Fig. 1(b). Then the expectation value E and the variance V of $\tilde{X}$ after accumulation are given as follows:

$$E[\tilde{X}] = (1/N_{\rm a})N_{\rm a}P_{\rm f}(U) = P_{\rm f}(U) \tag{2}$$

$$\begin{aligned} V[\tilde{X}] &= (1/N_{\rm a})^2 N_{\rm a}P_{\rm f}(U)(1 - P_{\rm f}(U)) \\ &= \frac{1}{N_{\rm a}}\left(\frac{1}{4} - 4\left(\frac{U}{U_{\max}} - \frac{1}{2}\right)^4\right) \end{aligned} \tag{3}$$

where $N_{\rm a}$ is accumulation time and $1/N_{\rm a}$ is a normalization constant.

It is useful to change the nonmonotonic characteristic. For example, the second threshold value θ where $P_{\rm f}(U)$ falls to zero, and the gain around $U = 0$ and θ are both important parameters. These parameters should be tuned for a target application. Next, let us consider the case of two random noises having a same split distribution, as shown in Fig. 1(c). The probability distribution of random noises $P(R)$ is given as follows:

$$P(R) = \begin{cases} 1/C, & (0 < R < a) \\ 0, & (a < R < b) \\ 1/C, & (b < R < U_{\max}) \end{cases} \tag{4}$$

where $C \equiv U_{\max} + a - b$. The output firing probability of the upper coding circuit $P_1(U)$ is given as follows:

$$\begin{aligned} P_1(U) &= \int_0^U P(R)\,dR \\ &= \begin{cases} U/C, & (0 < U < a) \\ a/C, & (a < U < b) \\ (U + D)/C, & (b < U < U_{\max}) \end{cases} \end{aligned} \tag{5}$$

where $D \equiv a - b$. Since $P_1(U)$ equals $P_2(U)$, the final output firing probability of the XOR gate $P_{\rm f}(U)$ is obtained as follows:

$$\begin{aligned} &P_{\rm f}(U) \\ &= \begin{cases} 2U(C - U)/C^2, & (0 < U < a) \\ 2a(C - a)/C^2, & (a < U < b) \\ -2(U + D)(U + D - C)/C^2, & (b < U < U_{\max}). \end{cases} \end{aligned} \tag{6}$$

When $a = U_{\max} - b$, $P_{\rm f}(U)$ is symmetrical and its maximum value equals 0.5. Here, we add a sign bit to the system in order to implement negative values. The system counts pulses up or down according to the sign bit. Then the entire characteristic required as a nonmonotonic neuron, which outputs ± 1, has been obtained in a stochastic case as shown in Fig. 2. The characteristic, which takes zero when U is above θ, is called end cutoff (ECO) characteristic. By changing parameters such as the maximum value $U_{\max}$ and the distribution widths a and b, one can obtain proper gain and threshold value θ. Furthermore, if either R_1 or R_2 is set to the maximum, the monotonic characteristic is obtained also. Various functions not restricted to the ECO function can be obtained with the combination of logic gates and suitable noises.

We design a neural network on the basis of the system which was proposed by Kondo *et al.* [13]. The discrete-time dynamics of a neural network is given by

$$u_i(t+1) = \sum_{j=1}^{N} w_{ij}x_j(t) \tag{7}$$

$$x_i(t) = f(u_i(t)) \tag{8}$$

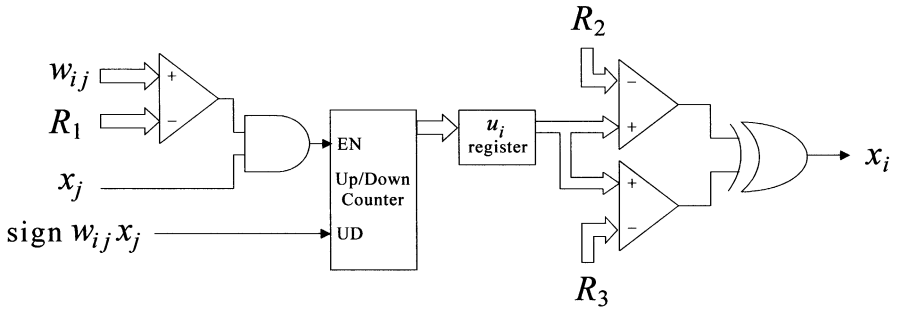


Fig. 3. Block diagram of a neuron circuit comprising a nonmonotonic coding circuit.

where w_{ij} is the synaptic weight from the jth neuron to the ith neuron, x_i is the ith neuron output, u_i is the membrane potential, $f(x)$ is an activation function, and N is the number of neurons. Fig. 3 shows the block diagram of the neuron circuit. The synaptic weight w_{ij} stored in a register is changed to a stochastic pulse sequence and multiplied by the output of the jth neuron x_j. The up/down counter counts the pulses up or down according to the sign of $w_{ij}x_j$ resulting the averaged product $w_{ij}x_j$. In one time step, only one neuron output is broadcasted to all other neurons. Therefore, N time steps are required for the summation $\sum_j w_{ij}x_j$. The accumulated value in the counter, which corresponds to $u_i(t+1)$ in (7), is changed to a pulse sequence by the nonmonotonic coding circuit in a proper time step. Thus, the discrete-time dynamics defined by (7) and (8) has been introduced. Please note that the state update should be done simultaneously to achieve the fastest calculation. However, some networks require asynchronous updates, and it is possible by updating a single neuron randomly in a time step though the computation power decreases in inverse proportion of N.

B. Learning Circuit

We incorporate Boltzmann machine (BM) learning [19] in the stochastic neural network since it has been confirmed the learning ability is enhanced with nonmonotonic neurons [9]. BM learning can be easily implemented with a simple stochastic logic circuit. Main idea related to Hebbian learning by stochastic logic, which is discussed in the following study, has been proposed by Kondo *et al.* [13]. We have modified the circuit in order to perform signed arithmetic. The synaptic weight update rule is given by the following equation:

$$\Delta w_{ij} = \frac{\varepsilon}{T}\sum_{\alpha}\left(x_i^{\alpha}x_j^{\alpha}\right)_{\mathrm{cl}} - \left(x_i^{\alpha}x_j^{\alpha}\right)_{\mathrm{uncl}} \tag{9}$$

where ε is a learning rate, T is a gain parameter, α is the index of desired patterns, and cl and uncl mean clamp and unclamp phases, respectively. The circuit realizing these operations are implemented with stochastic multipliers and up/down counters, in the same manner discussed in the above. Fig. 4 shows the block diagram of a new learning circuit. The simultaneous firing probability x_ix_j is obtained with the AND gate. A control pulse sequence is applied externally in order to obtain proper updating rate ε/T as the third input to the AND gate. Counting up or down is selected by the output of the XOR gate according to the phase (0:clamp/1:unclamp) and the sign of x_ix_j.

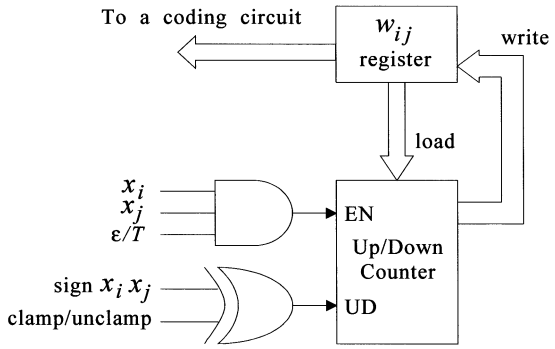


Fig. 4. Block diagram of a learning circuit.

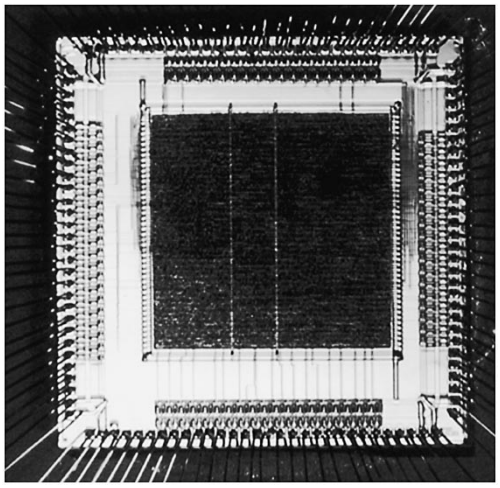


Fig. 5. Photograph of a stochastic neurochip. It has been fabricated through VDEC, in double-polysilicon, triple-metal, and 0.6 μm rule CMOS technology. The chip size is 4.5 mm × 4.5 mm.

III. Hardware Implementation

Fig. 5 shows a photograph of the stochastic neurochip which has been fabricated through VDEC 0.6 μm rule CMOS technology. Fifty neurons have been integrated with 5 M-sequence random number generators. Each random number generator outputs the 200-bit M-sequence. The bit length of a random number required for neuron operation is 20. Therefore, one random number generator is shared by ten neurons. This configuration was chosen after numerical simulations in which we confirmed that the correlation of noises was sufficient small for stochastic logic neural operation. The occupied silicon area by random number generators is 6.2% of the total chip area.

The bit lengths of u_i and w_{ij} are 14 and eight, respectively. Memories which store synaptic weights are provided externally. Fig. 6 shows the multichip configuration of the proposed chips. All neurons are connected together via the x_i bus. The neurons broadcast their outputs in order under the control of the host PC. The size and topology of the network can be defined externally. Though the synaptic weights and the memory size should be chosen properly, this architecture keeps good scalability. The fabricated chip has been tested using the host PC which generates the required control signals. We have confirmed that the neurochip operates successfully. A rather low clock frequency of 100 kHz was used for the measurements. This was limited by the interface card of the PC. However, we confirmed the chip can operate with over 30 MHz by Verilog simulations, in which propagation delay was considered. The CPS and CUPS of the proposed chip are estimated as follows:

$$\mathrm{CPS} = \frac{fN}{12 + N_{\mathrm{a}}},$$
$$\mathrm{CUPS} = \frac{fN}{2(\tau(N_{\mathrm{a}} + 12) + (24 + 512\epsilon/T))} \tag{10}$$

where f and τ are the clock frequency, and the number of the state update until the network finds a certain fixed point, respectively. Other coefficients are related to the operations such as synaptic weight loading or counting neuron output pulses for learning. Supposing typical values $f = 30$ MHz, $N = 1000$ (20 chips), $\tau = 100, \epsilon/T = 0.05$, and $N_{\mathrm{a}} = 10$, we obtain 1.37 GCPS and 6.67 MCUPS. The power consumption per chip is 330 mW at 30 MHz.

A. Learning Results

The learning performance as a Boltzmann machine has been evaluated. Parity problem, in which the output required is $+1/-1$ if the input vector contains odd/even number of $+1/-1$'s, $-1/+1$ else, is considered. Fig. 7 shows the learning results for the four-parity problem obtained with either nonmonotonic neurons or monotonic neurons. The networks have the same three layers, and nonmonotonic neurons are placed only in the hidden layer. The monotonic function is obtained by setting either R_1 or R_2 to the maximum. The number of required neurons for achieving successful learning decreases by the use of nonmonotonic neurons. We have confirmed that the stochastic nonmonotonic neurons operated properly. Please note that the results have been given by the discrete time dynamics. However, they agree well with the previous results obtained in the continuous time dynamics by Kinjo *et al.* [9]. A monotonic neuron can divide its input space by a certain plane perpendicular to its weight vector. On the other hand, a nonmonotonic neuron has three dividing hyperplanes. Thus a nonmonotonic network can obtain desirable input–output relation with less neurons.

B. Application to an Optimization Problem

Another advantage of the stochastic neurochip is to generate coding noises as shown in (3). When N_{a} is small, the circuit shows stochastic behavior influenced by coding noises. The steepest descent method of the energy function of a Hopfield network [20] works well for retrieving memorized patterns together with simulated annealing as demonstrated by Geman *et al.* [21]. One can achieve simulated annealing by coding noises with scheduled N_{a}. The five-city traveling salesman problem (TSP) [20] was solved either with or without simulated

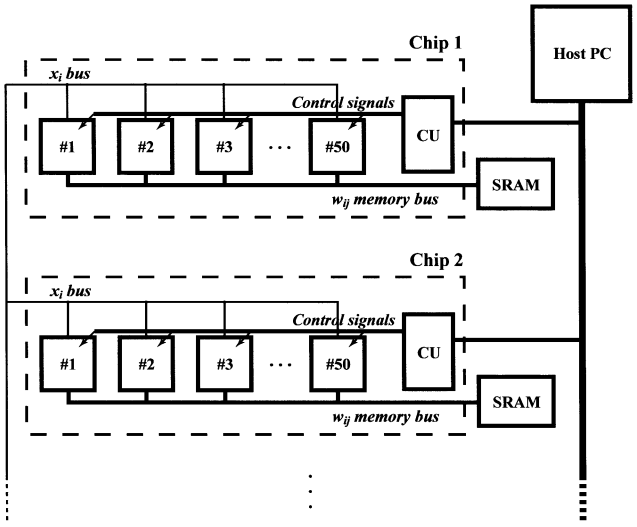


Fig. 6. Multichip network controlled by a host PC. Each chip comprises 50 neurons and one control unit (CU). Synaptic weights are stored in external SRAM's.

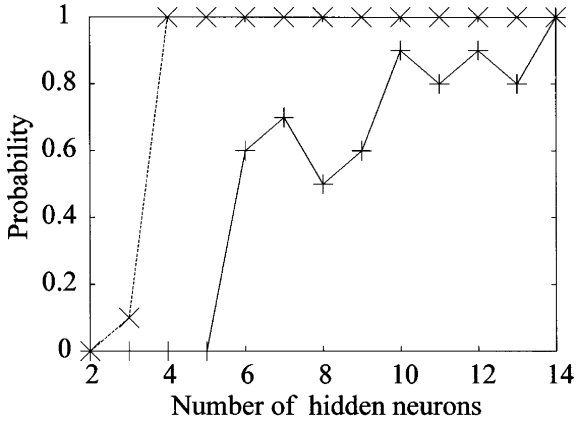


Fig. 7. Probability of successful learning as a function of the number of neurons in the hidden layer. × and + denote the nonmonotonic and monotonic cases, respectively. The split distribution in Fig. 1(c) is used. The parameters for the circuit operations are $a = 200, b = 300, U_{\max} = 500, N_{\mathrm{a}} = 1000$, and $\varepsilon/T = 0.048$.

TABLE I
PROBABILITY OF THE BEST SOLUTION OF THE FIVE-CITY TSP

N_{a}	Monotonic	Non-monotonic
450→600	29.5%	36.5%
600 (no scheduling)	28.7%	26.0%
∞(deterministic)	16.0%	17.5%

TABLE II
COMPARISON OF DIGITAL NEUROCHIPS

Institution (year)	Silicon Area per Neuron	Process Rule	Learning	Resolution	CPS	Power Consumption
NTT (1997)[23]	2.87mm^2	0.5μm	NA	16 bit	18G@30MHz	3.2W
Mitsubishi (1997)[24]	22.62mm^2	0.5μm	BP,Hebb	24 bit	81M@50MHz	4.0W
Proposed Chip	0.41mm^2	0.6μm	Hebb	14 bit	7M@30Mhz	330mW

annealing. The scheduling of noises was done by changing $N_{\rm a}$ as follows [13]:

$$N_{\rm a}(t) = N_{\rm a0}(1 + t/\tau_s)^2 \tag{11}$$

where $N_{\rm a0}$ is the initial value and τ_s is the time constant of the scheduling. Note that the gain of the activation function increases with $N_{\rm a}$. Therefore, the sharpening effect is also expected together with the annealing. The results have been obtained by asynchronous updating. This is because we avoid oscillatory behavior caused by synchronous updating on a Hopfield network. Table I shows the probability of the best solution. Both cases with and without simulated annealing have been evaluated with the parameters, $\tau_s = 100, a = 200, b = 4N_{\rm a} - 200$, and $U_{\max} = 4N_{\rm a}$. We chose 600 as the maximum $N_{\rm a}$ in order to avoid the overflow of u_i registers. The deterministic case was evaluated by numerical simulations. No significant difference between monotonic and nonmonotonic neurons has been found except for the case of simulated annealing. It can be seen that simulated annealing works well especially for nonmonotonic neurons. Also, even in the case of $N_{\rm a} = 600$ and without the scheduling, the probability is high. This is because there exists stationary noise. The probabilities increased for both cases greatly in comparison with the deterministic case. We have confirmed the coding noises work efficiently for the optimization problem.

IV. Discussion

Let us discuss the advantage and disadvantage of the proposed chip. The various arithmetic circuits can be implemented with less transistors by the stochastic logic. It is possible to integrate more neurons in a chip than a conventional digital circuit. Our rough estimation indicates that ten times integration can be achieved [22]. The comparison between the proposed chip and other digital neurochips is shown in Table II. The proposed circuit is suitable for a neural network requiring a large number of neurons on a single chip. On the other hand, the processing speed is not so fast due to pulse accumulating operation. With same clock frequency, the conventional digital circuit operates $N_{\rm a}$ times faster. However, we can control the amplitude of the noise by adjusting $N_{\rm a}$, and the coding noise is helpful for improving the network performance. Furthermore, $N_{\rm a}$ can be decreased in inverse proportion of the number of neurons N, since the variance of the neuron output is in proportion of $1/(N_{\rm a}N)$. This is a suitable feature for implementing a huge number of neurons.

Another interest should be considered here is the continuous time operation by the stochastic logic, which has not been discussed in previous sections. Kondo *et al.* have suggested that the stochastic logic can realize pseudocontinuous-time dynamics by implementing the following updating rule of u_i [13]:

$$u_i(t + \delta t) = u_i(t) + \frac{\delta t}{\tau} \sum_j^N w_{ij} x_j(t) \tag{12}$$

where τ is the decay time constant and δt is time step. This equation has been obtained after time discretization of the original equation under the condition $\tau \gg \delta t$. It is not difficult to implement this rule since all the operations appeared in (12) are familiar for the stochastic logic. The continuous time dynamics is much important for associative memories with nonmonotonic neurons [2]–[6]. The improved retrieving characteristic of stored patterns are obtained with nonmonotonic neurons in continuous time operation. We have tested the updating rule by numerical simulations. However, we have not confirmed any improvement. This is because decay process is omitted in (12), and such decay is crucially important for a nonmonotonic neuron to find fixed points. It is clear that u_i increases continuously even over the fixed point given by the original continuous time dynamics. In the monotonic case, this is not a problem since such behavior of u_i does not affect to the output x_i if u_i is sufficiently large. On the other hand, the improvement of the memory capacity given by nonmonotonic characteristic is related closely to the decay of u_i. A new circuit including decay process should be studied for the application to associative memories.

V. Conclusion

We have designed and fabricated the stochastic neurochip comprising nonmonotonic neurons. The nonmonotonic function has been obtained with the new circuit including two coding circuits and an XOR gate. The network having Boltzmann machine learning ability has been integrated using 0.6 μm CMOS technology. We have confirmed that the fabricated chip including 50 neurons operates successfully. The advantages of the new neurochip are easy integration with less transistors, implementation of the nonmonotonic function, and the stochastic operation. Networks composed of over 10^3 neurons are necessary in order to be applied to practical applications. Such a network is achieved by connecting multichips together. The proposed chips are easy to be connected since the global control signals are given by the host PC, and the size and topology of the network can be defined externally.

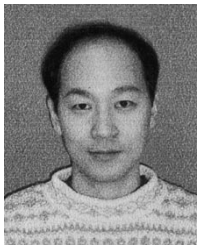
Acknowledgment

The authors wish to thank Y. Kondo, H. F. Yanai, Y. Hayakawa, and Y. Sawada for fruitful discussions, K. Sugawara for technical assistance, and T. Haga for programming asistance. The VLSI chip in this study has been fabricated in the chip fabrication program of VLSI Design and Education Center (VDEC), University of Tokyo, with collaboration by Rohm Corporation and Toppan Printing Corporation.

References

[1] J. E. Dayhoff, *Neural Network Architectures: An Introduction*. New York: Wiley, 1990.
[2] M. Morita, "Associative memory with nonmonotone dynamics," *Neural Networks*, vol. 6, pp. 115–126, 1993.
[3] S. Yoshizawa, M. Morita, and S. Amari, "Capacity of associative memory using a nonmonotonic neuron model," *Neural Networks*, vol. 6, pp. 167–176, 1993.
[4] T. Fukai, J. Kim, and M. Shiino, "Retrieval properties of analog neural networks and the nonmonotonicity of transfer functions," *Neural Networks*, vol. 8, pp. 391–404, 1995.
[5] H.-F. Yanai and S. Amari, "Auto-associative memory with two-stage dynamics of nonmonotonic neurons," *IEEE Trans. Neural Networks*, vol. 7, pp. 803–815, July 1996.
[6] M. Okada, "Notions of associative memory and sparse coding," *Neural Networks*, vol. 9, pp. 1429–1458, 1996.

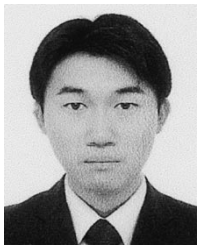
[7] M. Morita, "Memory and learning of sequential patterns by nonmonotone neural networks," *Neural Networks*, vol. 9, pp. 1477–1489, 1996.
[8] ——, "Computational study on the neural mechanism of sequential pattern memory," *Cogn. Brain Res.*, vol. 5, pp. 137–146, 1996.
[9] M. Kinjo, S. Sato, and K. Nakajima, "Hardware implementation of a DBM network with nonmonotonic neurons," *IEICE Trans. Inform. Syst.*, vol. E85-D, pp. 558–567, 2002.
[10] D. E. van den Bout and T. K. Miller III, "A digital architecture employing stochasticism for the simulation of Hopfield neural network," *IEEE Trans. Circuits Syst.*, vol. 36, pp. 732–738, May 1989.
[11] A. F. Murray, D. D. Korso, and L. Tarassenko, "Pulse-stream VLSI neural networks mixing analog and digital techniques," *IEEE Trans. Neural Networks*, vol. 2, pp. 193–204, June 1991.
[12] G. Moon, M. E. Zaghloul, and R. E. Newcomb, "VLSI implementation of synaptic weighting and summing in pulse coded neural-type cells," *IEEE Trans. Neural Networks*, vol. 3, pp. 394–403, Mar. 1992.
[13] Y. Kondo and Y. Sawada, "Functional abilities of a stochastic logic neural network," *IEEE Trans. Neural Networks*, vol. 3, pp. 434–443, May 1993.
[14] S. Sato, M. Yumine, T. Yama, J. Murota, K. Nakajima, and Y. Sawada, "LSI neural chip of pulse-output network with programmable synapse," *IEICE Trans. Electron.*, vol. E78-C, pp. 94–100, 1995.
[15] Y. Hirai and M. Yasunaga, "A PDM digital neural network system with 1 000 neurons fully interconnected via 1 000 000 6-bit synapses," in *Proc. 1996 Int. Conf. Neural Information Processing*, 1996, pp. 1251–1256.
[16] T. Morie, J. Funakoshi, M. Nagata, and A. Iwata, "An analog-digital merged neural circuit using pulse width modulation technique," *IEICE Trans. Fund. Electron. Commun. Comput. Sci.*, vol. E82-A, pp. 356–363, 1999.
[17] B. D. Brown and H. C. Card, "Stochastic neural computation I: Computational elements," *IEEE Trans. Comput.*, vol. 50, pp. 891–905, Sept. 2001.
[18] B. R. Gains, *Advances in Information Systems Science*. New York, NY: Plenum, 1969. Stochastic Computing Systems.
[19] G. E. Hinton and T. J. Sejnowski, *Parallel Distributed Processing*. Cambridge, MA: MIT Press, 1986. Learning and Relearning in Boltzmann Machines.
[20] J. J. Hopfield and D. W. Tank, "Neural computation of decisions in optimization problems," *Biol. Cybern.*, vol. 52, pp. 141–152, 1985.
[21] S. Geman and D. Geman, "Stochastic relaxation, Gibbs distribution, and the Baysian restoration of images," *IEEE Trans. Pattern Anal. Machine Intell.*, vol. PAMI-6, pp. 721–741, Nov. 1984.
[22] K. Nemoto, M. Kinjo, S. Sato, and K. Nakajima, "A nonmonotonic neurochip using stochastic logic," in *Proc. 2001 Int. Symp. Nonlinear Theory Applications*, 2001, pp. 605–608.
[23] K. Aihara, O. Fujita, and K. Uchimura, "A sparse memory access architecture for digital neural network LSIs," *IEICE Trans. Electron.*, vol. E80-C, pp. 996–1002, 1997.
[24] S. Komori, Y. Arima, Y. Kondo, H. Tsubota, K. Tanaka, and K. Kyuma, "A 3.2 GFLOPS neural network accelerator," *IEICE Trans. Electron.*, vol. E80-C, pp. 859–867, 1997.



Shigeo Sato received the B.S., M.S., and Ph.D. degrees from Tohoku University, Sendai, Japan, in 1989, 1991, and 1994, respectively.

In 1996, he joined the Research Institute of Electrical Communication, Tohoku University. His research interests are in LSI implementation and application of neural networks.

Dr. Sato is a member of the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan and Japanese Neural Network Society.

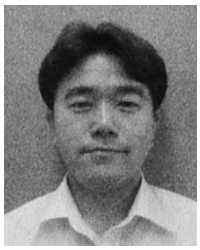


Ken Nemoto received the B.E. and M.E. degrees from Tohoku University, Sendai, Japan, in 2000 and 2002, respectively.

He was engaged in the research of digital neuron systems and their architectures. In 2002, he joined East Japan Railway Company. His research interests are in traffic control and information systems.

Shunsuke Akimoto received the B.E. degree in communication engineering from Tohoku University in 2002. He is currently pursuing the M.S. degree in the Graduate School of Information Sciences, Tohoku University.

His current research interests are hardware implementation of artificial neural networks and their applications.



Mitsunaga Kinjo received the B.E. degree in electrical and information engineering from University of the Ryukyus, Okinawa, Japan, in 1996 and the M.E. degree in electrical engineering and Ph.D. degree in information science from Tohoku University, Sendai, Japan, in 1998 and 2001, respectively.

Since April 2001, he has been a Researcher in the PRESTO program of the Japan Science and Technology Corporation (JST) at Tohoku University. His research interests are in quantum computation and its hardware implementation.

Koji Nakajima received the B.E., M.E., and Dr.Eng. degrees from Tohoku University, Sendai, Japan, in 1972, 1975, and 1978, respectively.

Since 1978, he has been working on Josephson junctions and soliton physics at the Research Institute of Electrical Communication, Tohoku University, except for a ten-month period in 1983, when he worked as a Visiting Assistant Research Engineer at the University of California, Berkeley. He is currently engaged in the study of VLSI implementation of neural networks, and Josephson junction devices for digital applications as a Professor at the Research Institute of Electrical Communication, Tohoku University.

Dr. Nakajima is a member of the Japan Society of Applied Physics, the Institute of Electrical Engineers of Japan, and Japanese Neural Network Society. He received the Excellent Paper Award from the Institute of Electronics, Information and Communication Engineers (IEICE) of Japan in 2000.