Applied Mathematics and Computation 120 (2001) 65–77
APPLIED MATHEMATICS AND COMPUTATION
www.elsevier.com/locate/amc

Learning performance of a neurocomputer for nonlinear dynamical system identification

Masanori Sugisaka *, Masayo Nagasaki [1]

Department of Electrical and Electronic Engineering, Oita University, 700 Oaza Dannoharu, Oita 870-1192, Japan

Abstract

This paper investigates the learning performance of a RICOH neurocomputer RN-2000 for the identification problem of input and output map of a discrete nonlinear dynamical system. The results obtained show capability of on-chip learning, which is essential for many neural applications such as machine learning and control where real-time adaptation is required. In this paper, the method to use a neurocomputer is briefly presented for a nonlinear identification problem. The main significance of this research is to obtain a further guideline for designing a primitive artificial brain for robotics.

1. Introduction

System identification [1,2] plays an important role to design a control system. The identification is generally treated as a problem for estimating unknown parameters involved in models using optimization methods under the assumption that the structures of the models are known. However, although the true parameters in the system can be estimated for the case where the structure of the model assumed coincides with that of the system, the characteristics or structure are unknown and/or too complicated in many cases.

* Corresponding author. Tel.: +81-975-54-7841; fax: +81-975-54-7831.
E-mail address: msugi@cc.oita-u.ac.jp (M. Sugisaka).
[1] Presently NTT.

PII: S0096-3003(99)00288-X

In addition, the system may be nonstationary and nonlinear. Consequently, the accuracy of the estimated parameters is not quite satisfactory. In other words, the identifiers stated above are not robust and, therefore, the performances of a control law based on them are not satisfactory.

A new technique, which can be used for real identification problems has to be employed in order to circumvent the above difficulties. The artificial neural network system (denoted as ANNS) [3–6] is considered as one of the identification technologies, which is able to model the systems with unknown structures, nonlinearities, and disturbances. It is known that the ANNSs have the ability to learn the relationships between the inputs and outputs of the systems.

We have proposed new neural identifiers utilizing the ANNSs with tapped delay line added to the input layer of the ANNSs in order to learn dynamic maps or systems [7] and also have developed a new model predictive controller (denoted as MPC) based on the neural identifiers [8]. The neural identifiers proposed use the back propagation method in order to minimize the errors between the outputs of the ANNS and the outputs of the system to be identified. It was shown from numerical experiments that the neural identifiers have robustness in the change of operational conditions and circumstances and hence the control performances of the MPCs are quite satisfactory.

However, the neural identifiers based on the software constructed in the Neumann-type computers require huge computational time for training the ANNS using the input and output data of the systems to be identified. Hence, it is very hard to control mechanical systems with unknown structures, etc., in on-line fashion using the neural identifiers based on the software.

For the purposes of developing on-line controllers for mechanical systems such as a mobile vehicle guided with a CCD camera, robots, etc., we developed a new identification method [9] based on a hardware, viz., a RICOH neurocomputer RN-2000, which is constructed from seven digital neural network VLSI chips, for general dynamical systems. The VLSI chip RN-200 [10] has 16 neurons and totally 256 synapses are integrated in a $13.73 \times 13.73\ \mathrm{mm}^2$ VLSI chips fabricated by RICOH 0.8 μm CMOS technology. This chip can perform 5.12 giga pulse operations per second. It corresponds to effective neural computing rate of 40M CPS or 40M CUPS.

The feedforward process to obtain one output signal from many input signals in the neurocomputer RN-2000 is performed based on the McCulloch and Pitts model [11]. Namely, the process is as follows:

1. Calculate the product of each input signal and synapse weight.
2. Calculate the sum of products.
3. Perform nonlinear (sigmoid function) processing of the sum in order to obtain the output signal from a neuron.

In the case of processing the above calculations by a digital circuit, the processing speed by the usual floating decimal point operation is slow and the execution in parallel is difficult. The feedforward process in a neuron model is performed by AND and OR operations. The pulse density manipulation in forward process is shown in Fig. 1.

On the other hand, the learning process is based on the back-propagation method proposed by Rumelhart [12]. The calculations of numerical values in the back-propagation method are replaced by the logic operations, namely, AND and OR [13].

In the paper given by Sugisaka et al. [9], we presented the hardware-based neural identification method to learn the characteristics or structures of dynamical systems and explained how to implement the neural identification in both learning and feedforward processing (recognizing) of the linear discrete dynamical system, using the neurocomputer RN-2000. Based on the results obtained in the paper given by Sugisaka et al. [9] this paper presents the results of nonlinear system identification using the neurocomputer RN-2000 [14].

As the input into a nonlinear discrete dynamical system, we employed two types of inputs, viz., rectangular pulse signal and sinusoidal signal. The following three types of neural network structures are considered to investigate identification performances of the neurocomputer RN-2000:

1. Eight neurons in the input layer and one neuron in the output layer.
2. Seven neurons in the input layer and one neuron in the output layer.
3. One neuron in the input layer and one neuron in the output layer.

The number of neurons in the remaining first and second intermediate layers are fixed at 16, respectively. The neural network system has four layers. The numerical values obtained from the input and output in the system are normalized such that the maximum value corresponds to the digit 127 (which is equal to 1) and the minimum value corresponds to the digit 0 (which is equal to 0) according to the hardware specification. In the identification of the

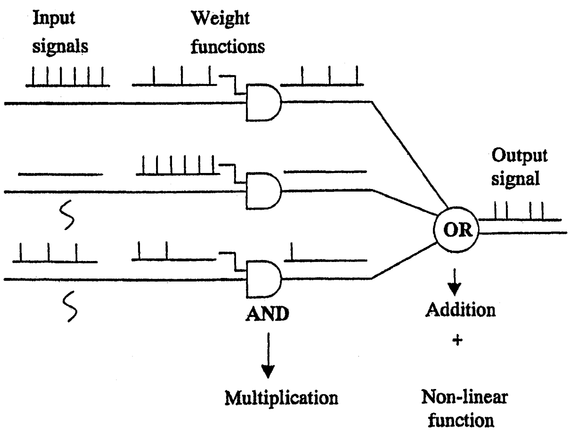

Fig. 1. Circuitry for the feedforward operation of a multi-input neuron.

dynamical system, the input is applied to the system and the output value of the system is obtained at each sampling time. The output values of the system are transformed into the binary numbers (the BCD code) using the bit transformation technique.

Then the learning of the unknown system dynamics is performed in the RN-2000 by changing the learning numbers. The results obtained from the neural calculation verified on-chip learning capability which is essential for many neural applications, such as machine learning and controls, running of mobile vehicles, inspection in the manufacturing line, etc., where real-time and continuous adaptation is required or off-line learning (stand-alone or embedded application) are avoided.

In Section 2, the model of the system to be identified is presented. Section 3 shows the structure of the ANNS based on the software and the identification performance of the neurocomputer RN-2000 by changing the structure of neural network and the results of the neural identification follow and, finally, the conclusion is presented in Section 4.

2. System model

In order to investigate identification performance of the neurocomputer RN-2000, the simple nonlinear dynamical model specified by

$$y_{i+1} = 0.8\sin(y_i) + 1.2u_i \qquad (1)$$

is used as one of the nonlinear systems where u and y are the input and the output, respectively. Here i means iT_s and T_s denotes the sampling time. For simplicity, we used the simple nonlinear dynamical model stated above in order to obtain the training or learning data for a neurocomputer.

3. Identification using neurocomputer

For the purpose of facilitating a better understanding of the neural identification method using the neurocomputer, we briefly explain the structure of the conventional ANNS which is used for identifying the structure of the model based on the software constructed in the Neumann-type microcomputer [7,8].

Fig. 2 [7,8] shows a nonrecurrent ANNS, which consists of four layers. The input layer has two neurons, which receive both the input and the output of the model at ith sampling time. The output layer has one neuron that produces the output of the model at $(i+1)$th sampling time. The first and second hidden layers have five and two neurons, respectively, and are constructed in the Neumann-type microcomputer. The structure shown in Fig. 2 is used to identify the unknown structure of the model because the Neumann-type

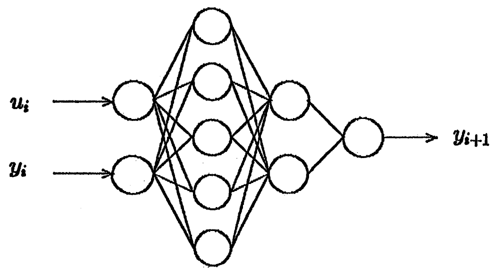

Fig. 2. Structure of ANNS based on software.

microcomputer is able to process real values. As stated in Section 1, training the ANNSs by the Neumann-type microcomputer requires huge computational time and, therefore, the training by neurocomputers, which consist of hardware, is required in order to shorten the training time. However, since the neurocomputer RN-2000 can process only integers as the data to the neurons in the input layer and can process integers as the data from the neurons in the output layer, the ANNSs which have other structures have to be constructed in the neurocomputer.

In the following we show the structures of the ANNS which are constructed in the neurocomputer and the identification performance using the proposed structures. The neurocomputer consists of four layers where each layer has 16 neurons. We use two types of input to the model. As stated in Section 1, the maximum value of the output corresponds to the digit 127 and the minimum value of the output corresponds to the digit 0 according to the hardware specification of the neurocomputer. In other words, the digit one used for the calculations (training and recognition) in the neurocomputer corresponds to 127 pulses in the input sequences and the digit zero corresponds to no pulses. Hence, in order to express the maximum value 127 by the binary expression seven bits are needed.

3.1. Identification performance of structure 1

Structure 1 of the neurocomputer RN-2000 is illustrated in Fig. 3, where the input layer has eight neurons and the output layer has one neuron. The inputs to the eight neurons in the input layer are 0 (0) or 1 (127). The conceptual data structure and their patterns used in the neurocomputer are illustrated in Fig. 4. In Fig. 4 the first one bit or neuron is used for the input data and the remaining seven bits or neurons are used for the output data in the input layer and the one bit or neuron is used for the output data in the output layer. The pairs of $(N+1)$ patterns used as the training data are shown in Fig. 4.

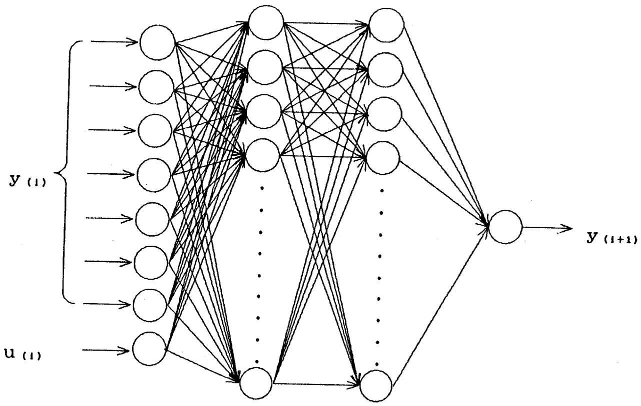

Fig. 3. Structure of neurocomputer RN-2000 (structure 1).

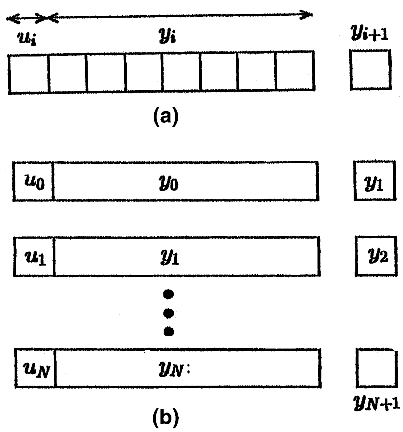

Fig. 4. (a) Data structure for identification; (b) $(N+1)$ pairs of patterns for training.

The flowchart of the identification method using the neurocomputer is given in Fig. 5. In the flowchart the output data from the model whose inputs are rectangular pulse and sinusoidal functions are calculated, then the data are normalized by using the maximum and minimum values of the data and, thereafter, the normalized values are transformed into the integers between 0 and 127. The input data for graphics are stored in the data form specified by $*$.dat. The input and output data for the simulations by the neurocomputer are calculated by using the bit transformation technique and are stored in the data form specified by $*$.rnd.

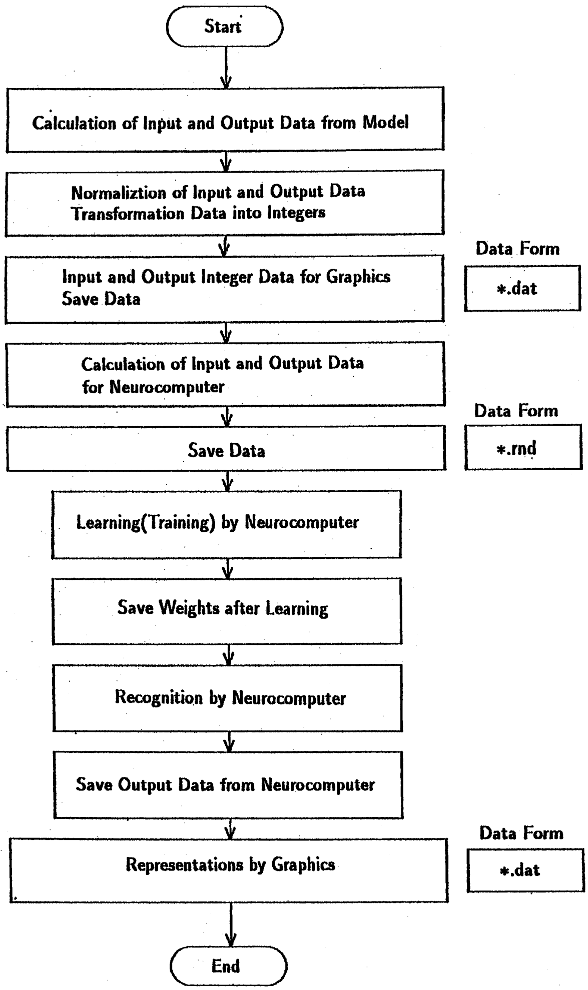

Fig. 5. Flowchart of identification using neurocomputer.

On making use of the data obtained from the above processes, the learning or training process is performed in the neurocomputer and, thereafter, the data of the synapsis weights are stored after the learning process is completed. Using the data of the weights obtained we perform the recognition or forward process. The output data are stored in the same data form as before, after the recognition process has completed. Thereafter, both the input and output data of the model and the output data from the output layer in the neurocomputer are illustrated by using the graphics and finally the identification process is completed.

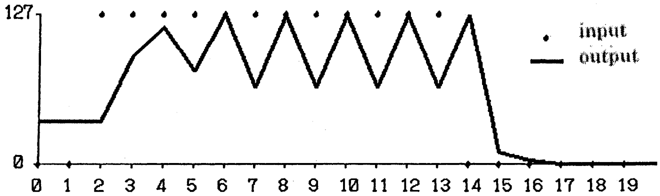

Fig. 6. Learning data for RN-2000 (input and output data).

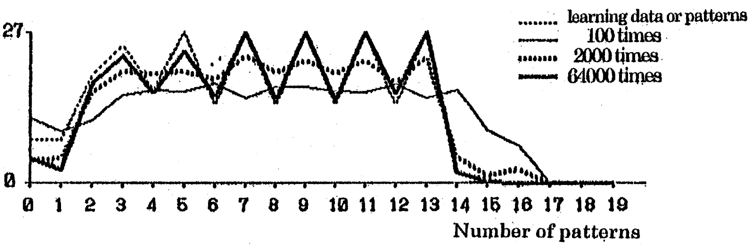

Fig. 7. The learning data and the outputs learnt by RN-2000.

For the numerical calculation, we used the rectangular pulse as the input applied to the model in order to obtain 20 pairs ($N = 20$) of learning data for RN-2000

$$u_i = \begin{cases} 1, & 2 \leqslant i \leqslant 13, \\ 0, & \text{others}. \end{cases} \tag{2}$$

The learning data or pattern is shown in Fig. 6. The output results learnt by RN-2000 are illustrated in Fig. 7, where the number of learning are 100, 2000 and 64,000. The sum of mean square error in the learning process, given by

$$E_\mathrm{p} = \sum_{i=0}^{19} (z_i - y_i)^2/2, \tag{3}$$

is illustrated in Fig. 8. From this figure we can see that the sum of mean square errors converges after the number of learning for one pattern passed over 100.

3.2. Identification performance of structure 2

Structure 2 of RN-2000 is shown in Fig. 9. The input layer has seven neurons. The inputs to the seven neurons in the input layer are 0 (0) or 1 (127). The first and second intermediate layers have 16 neurons, respectively. The output layer has one neuron. The input to the nonlinear dynamical system given by (1) is the following sinusoidal function:

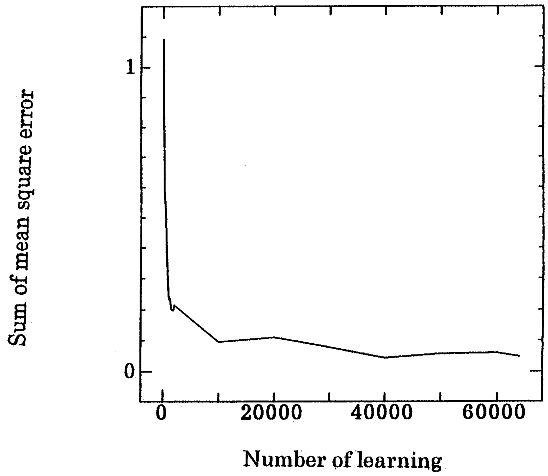

Fig. 8. Sum of mean square error.

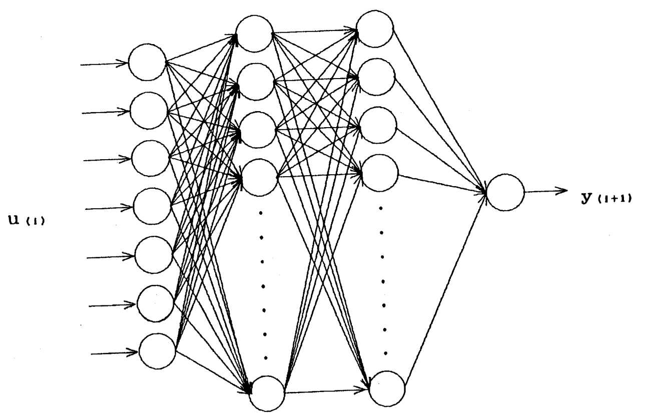

Fig. 9. Structure of neurocomputer RN-2000 (structure 2).

$$u_i = \sin(2\pi i/20)t + 1. \tag{4}$$

The input and output data for simulation are illustrated in Fig. 10. We used a part of learning data, namely, the stationary data for $35 \leqslant i \leqslant 54$.

The results obtained from learning 20 patterns or data ranging from 35 to 54 were not satisfactory. Therefore, the data were divided into two parts. One part is the data for $35 \leqslant i \leqslant 44$. The other part is the data for $45 \leqslant i \leqslant 54$. The data in each part were learnt independently. The learnt results for one period are shown in Fig. 11 by changing the number of learning.

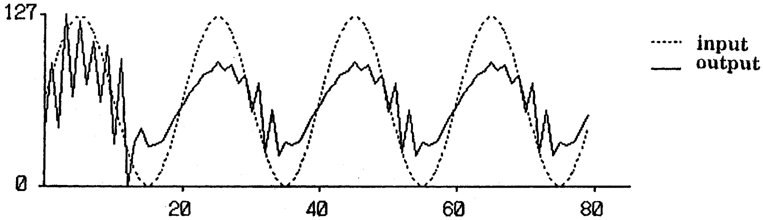

Fig. 10. Learning data for RN-2000 (input and output data).

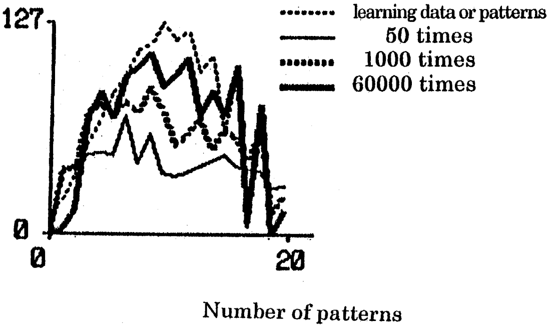

Fig. 11. Learning data and outputs learnt by structure 2 in RN-2000.

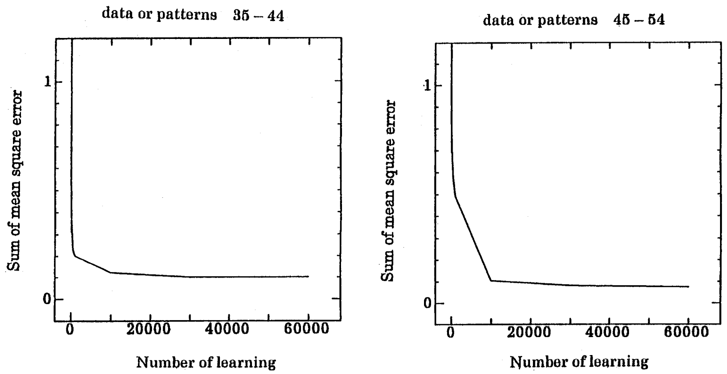

Fig. 12. Sum of mean square error.

The sums of mean square error for both the data from 35 to 44 and the data from 45 to 54 are shown in Fig. 12. From this figures we see that the sums of mean square error converge after the number of learning for one pattern passed over 1000. And also the accuracy of learning is not satisfactory as before.

3.3. Identification performance of structure 3

Structure 3 of the neurocomputer RN-2000 is shown in Fig. 13 where the input layer has one neuron and the output neuron has one neuron. The input to the one neuron in the input layer takes digits ranging from 0 to 127. The input to the nonlinear dynamical system is the same as the one given by Eq. (4). Also the output in the structure 3 of RN-2000 takes digits ranged from 0 to 127. The learnt results for one period are shown in Fig. 14 and the sums of mean square error for both the data from 35 to 44 and the data from 45 to 54

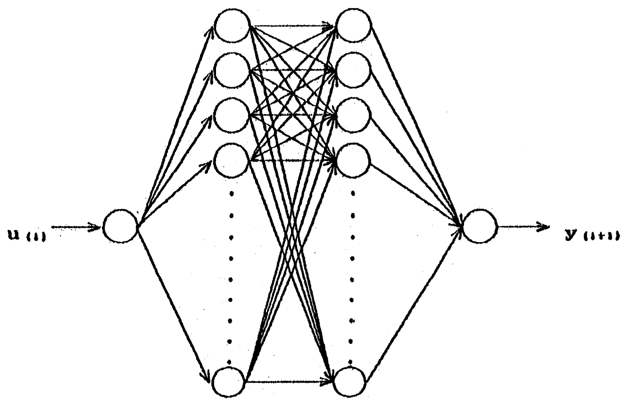

Fig. 13. Structure of neurocomputer RN-2000 (structure 3).

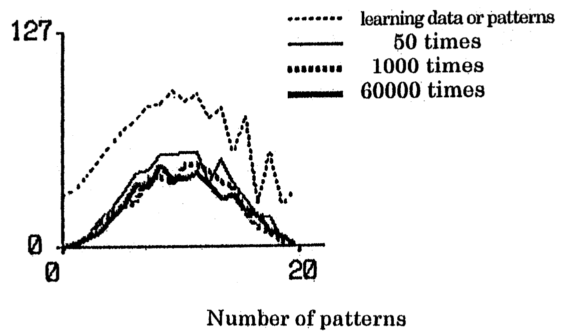

Fig. 14. Learning data and outputs learnt by structure 3 in RN-2000.

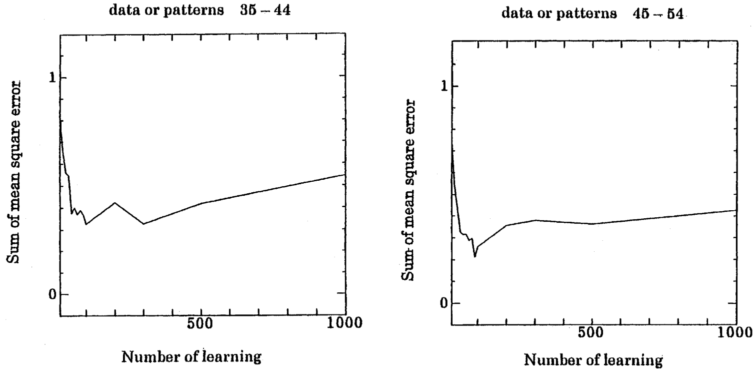

Fig. 15. Sum of mean square error.

are shown in Fig. 15. From this figure we see that the accuracy of learning is not satisfactory. The result obtained for structure 3 of the RN-2000 is worst compared with the other two structures.

4. Conclusion

In this paper we investigated the learning performance of the RICOH neurocomputer RN-2000. We considered three different types of structure for the artificial neural network in the neurocomputer RN-2000. From the simulated results we conclude that the structures 1 and 2 provide us with the relatively better performance of learning capability. If we use the integer ranging from 1 to 127 as the value into a neuron in the input layer, we are not able to obtain the high performance of learning.

From our simulated results we see that the neurocomputer RN-2000 has on-chip learning capability which is essential for many neural applications such as machine learning and controls, running of mobile vehicles, inspection in manufacturing line, etc., where real-time and continuous adaptation is required and off-line learning is avoided.

We note that the idea of artificial brain for robotics was proposed recently by the author and an artificial brain was developed for tracking moving objects [15]. The artificial brain consists of a neural network hardware or a neurocomputer, Von Neumann-type computer, and interface etc. In this research, learning performance of a neurocomputer for identification problems was investigated. The main significance of this research is that we obtained the guideline of how to construct the structure of a neural network when we design an artificial brain for robotics.

References

[1] P. Eykoff, System Identification-Parameter and State Estimation, Wiley, London, 1974.

[2] H.H. Kagiwada, System Identification Methods and Application, Addison-Wesley, Reading, MA, 1974.

[3] K. Nakano, K. Iinuma, Neuron-net-group, S. Kiritani, Nyumonn to Jishyu, Neurocomputer, Gizyutuhyoronsha, Tokyo, 1989 (in Japanese).

[4] T. Kikuti, Nyumonn Neurocomputer, Ohmsha, Tokyo, 1990 (in Japanese).

[5] M. Sugisaka, Y. Kodama, S. Sagara, S. Ueno, H.H. Kagiwada, Characters recognition via artificial neural network systems, in: Proceedings of the International Conference on Fuzzy Logic and Neural Networks, Iizuka, Japan, July 1990, pp. 311–314.

[6] M. Sugisaka, S. Sagara, S. Ueno, Pattern recognition via artificial neural network systems, in: Proceedings of the 1990 Korean Automatic Control Conference, Seoul, Korea, October 1990, pp. 929–932.

[7] M. Sugisaka, M. Ino, A neuro identifier for linear and nonlinear systems, System Science XI, International Conference on Systems Science, Technical University of Wroclaw, Poland, September 1992 (also Systems Science 20 (3) (1994) 55–64).

[8] M. Sugisaka, M. Ino, Model predictive control based on neural identification method, Computers and Mathematics with Applications 20 (3) (1994) 83–93.

[9] M. Sugisaka, S. Motomura, T. Kitaguchi, H. Eguchi, Hardware based neural identification: Linear dynamical systems, in: Proceedings of International Workshop on Intelligent Systems and Innovative Computations – The Sixth Bellman Continuum, Tokyo Metropolitan University, Tokyo, Japan, August 1994, pp. 124–131 (also Artificial Life and Robotics 1 (3) (1997) 151–155).

[10] S. Oteki, A. Hashimoto, T. Furuta, T. Motomura, T. Watanebe, D.G. Stork, H. Eguchi, A digital neural network VLSI with on-chip learning using stocastic pulse encording, in: Proceedings of 1993 International Conference on Neural Networks, Nagoya, Japan, October 1993, pp. 3039–3045.

[11] W.S. McCulloch, W.H. Pitts, A logical calculus of the ideas immanent in nervous activity, Bullet. Math. Biophysics 5 (1943) 115–133.

[12] D.E. Rumelhart, J.L. McClelland, PDP Research Group (Eds.), Parallel Distributed Processing, MIT Press, Cambridge, MA, 1986.

[13] H. Eguchi, T. Furuta, H. Horiguchi, S. Oteki, Neural network hardware and learning function utilizing pulse-density modulation, Transactions of the Institute of Electronics, Information and Communication Engineers J. 74-C II (5) (1991) 369–376.

[14] M. Sugisaka, Hardware based neural identification: Nonlinear dynamical systems, in: Proceedings of the Second Asia-Pacific Conference on Control & Measurement, Wuhan-Chongqing, China, June 1995, pp. 255–259.

[15] M. Sugisaka, Neurocomputer control in an artificial brain for tracking moving objects, Artificial Life and Robotics 1 (1) (1997) 45–51.