Effectiveness of Artificial Neural Network with Time-Varying Coupling System

Yoko Uwate and Yoshifumi Nishio
Dept. of Electrical and Electronic Engineering,
Tokushima University
2-1 Minami-Josanjima, Tokushima, Japan
Email: {uwate, nishio}@ee.tokushima-u.ac.jp

***Abstract*— We have recently proposed a novel neural network structure called an "Affordable Neural Network" (AfNN), in which affordable neurons of the hidden layer are considered as the elements responsible for the robustness property as is observed in human brain function.**

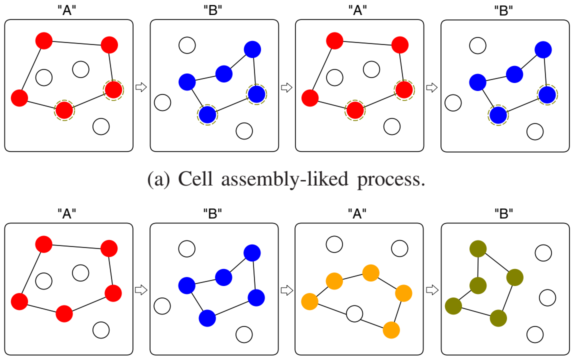
In this study, we investigate the mechanism of learning process of the AfNN to make clear the reason of that the AfNNs can perform well for learning and generalization abilities and operate as usually against damaging neurons.

I. Introduction

The mechanisms of many kinds of higher brain functions become clear with the advances in neuroscience technology. Therefore, the studies of bio-inspired artificial neural networks have been extensively reported to realize such intelligence systems applying for the future engineering applications. However, the performance of artificial neural networks still lags behind that of biological networks in many respects. In order to fill the gap between biological neural systems and artificial neural systems, it is important to apply these high functional mechanisms of the human brain to novel artificial neural networks.

In our previous work, we have proposed a new network structure of the feedforward neural network with affordable neurons in the hidden layer for Backpropagation (BP) learning [1]. We named this network "Affordable Neural Network (AfNN)." The AfNN was inspired by cell assembly [2]-[5]. which is one of explanations of multiple information processing in the brain and an information is represented by a firing space pattern of a group of plural neurons (see Fig 1(a)). In the AfNN, we prepare some extra neurons more than requires in the hidden layer. When the AfNN executes operating, all of the neurons in the hidden layer are not used at every updating. Namely, the AfNN is able to operate with high function by using different neuron patterns in the hidden layer. By computer simulations, the AfNN has been confirmed to gain better performance for the BP learning on both learning ability and generalization ability.

On the other hand, am important issue is the understanding how the mammalian brain is affected in its performance by the omnipresent death of their elements in the field of Neuroscience. It is also important to investigate the performance of the artificial neural network when some neurons are damaged. Several research groups have reported the studies of fault tolerance in the artificial neural networks [6]-[8]. In order to confirm that the AfNN can generate a kind of fault tolerance, we also have investigated the durability of the AfNN when some of the neurons in the hidden layer are damaged after the learning process [9]. We have confirmed that the AfNN can operates with keeping its efficiency against damaging neurons. However, the reason of what property of learning process of the AfNN affect good performance for the learning ability and durability has not yet understood in detail.



(a) Cell assembly-liked process.

(b) AfNN learning process.

Fig. 1. Learning process of neural networks for each input pattern.

We focus on the learning process of AfNN which used different neuron patterns for each input pattern. In the case of the AfNN, the affordable neurons are selected "at every updating time." The AfNN learns by using all neurons in the hidden layer with different pattern for every input patterns (see Fig. 1(b)). Therefore, we assume that the learning process of the AfNN is more complex and it has a possibility to create the high functions such as durability and flexibility. In order to confirm the effectiveness of the learning process of the AfNN, we consider different learning process with the AfNN. In this learning process, the affordable neurons are selected for "for each pattern." The network learns by using specific neurons in the hidden layer for each input pattern (see Fig. 1(a)).

In this study, we compare the performance of two types of the learning process when the AfNNs are applied for the pattern recognition. The results support the view that AfNN learned by all neurons with different pattern embody the important feature of durability.

978-1-4799-3432-4/14/$31.00 ©2014 IEEE

II. Affordable Neural Network (AfNN)

We have proposed a network with affordable neurons in the hidden layer of the three-layered feedforward neural network (one input layer, one hidden layer and one output layer) for the BP learning. The output characteristics of the neurons are implemented by a sigmoid function. We introduced the affordable neurons to reflect important properties of the brain. During the BP learning, all of neurons in the hidden layer are not used at every updating. Namely, some of the neurons are selected for the learning and the rest of the neurons are deactivated. The affordable neurons are selected by random at every updating time. The network model of the AfNN is shown in Fig. 2.

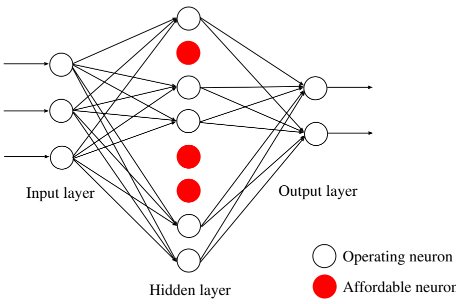


Fig. 2. Affordable neural network.

Here, we explain the learning process of the AfNN. In the case of the standard AfNN (abbr. AfNN-std), the affordable neurons are selected at every iteration time. Namely, the AfNN-std leans every input pattern by using the all neurons in the hidden layer with different pattern as shown in Fig. 3. We assume that this learning process is very effective to produce the durability. In order to confirm the effectiveness of the AfNN-std, we consider a different learning process to the AfNN. In a new learning process, same neurons patterns are used with respect to each input pattern. We call such AfNN "AfNN-ptn". Figure 4 shows operating mechanism of AfNN-ptn. In this study, we compare the the AfNN-std and the AfNN-ptn when the AfNNs are applied for the pattern recognition.

The standard BP learning algorithm was introduced in [10]. The BP is the most common learning algorithm for the feedforward neural networks. In this study, we use the batch BP learning algorithm. The batch BP learning algorithm is expressed by a formula similar to the standard BP learning algorithm. The difference lies in the timing of the weight. The update of the standard BP is performed after each single input data, while for the bath BP the update is performed after all input data has been processed. The total error E of the network is defined as

$$E = \sum_{p=1}^{P} E_p = \sum_{p=1}^{P} \left\{ \frac{1}{2} \sum_{i=1}^{N} (t_{pi} - o_{pi})^2 \right\}, \qquad (1)$$

where P is the number of the input data, N is the number of the neurons in the output layer, t_{pi} denotes the value of the desired target data for the pth input data, and o_{pi} denotes the value of the output data for the pth input data. The goal of the learning is to set the weights between the neurons so as to minimize the total error E. In order to minimize E, the weights are adjusted according to:

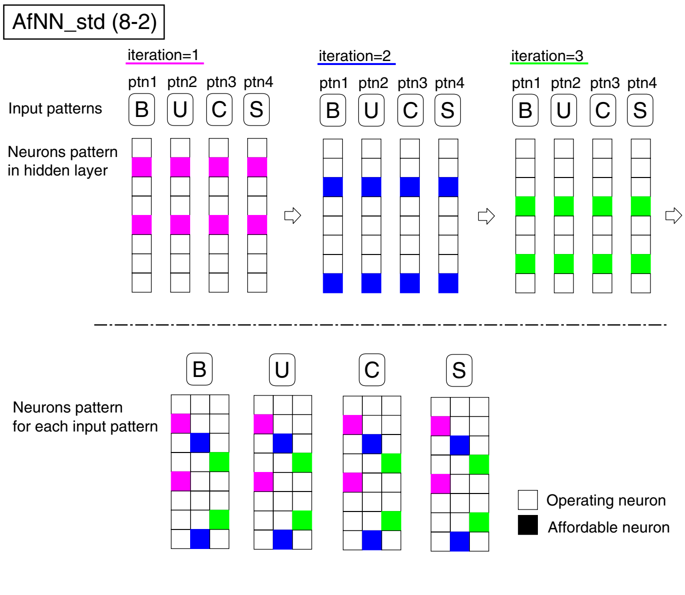


Fig. 3. Learning process of AfNN-std.

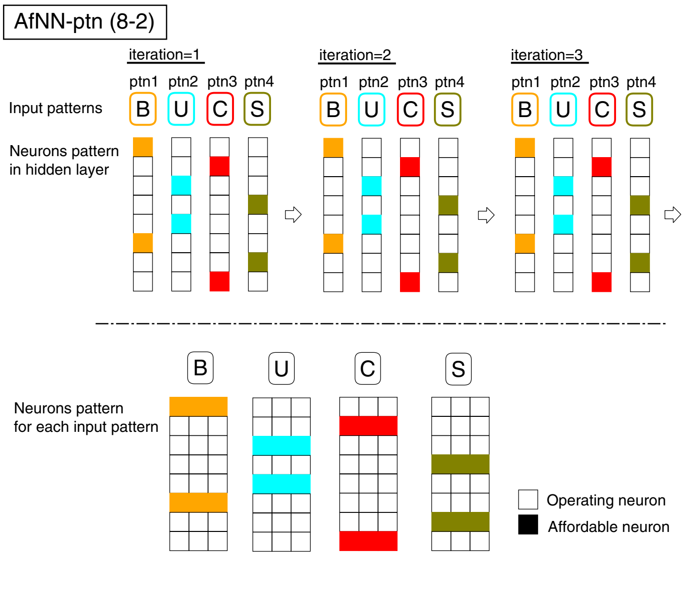


Fig. 4. Learning process of AfNN-ptn.

$$\begin{aligned} w_{i,j}^{k-1,k}(m+1) &= w_{i,j}^{k-1,k}(m) + \sum_{p=1}^{P} \Delta_p w_{i,j}^{k-1,k}(m), \\ \Delta_p w_{i,j}^{k-1,k}(m) &= -\eta \frac{\partial E_p}{\partial w_{i,j}^{k-1,k}}, \end{aligned} \qquad (2)$$

were $w_{i,j}^{k-1,k}$ is the weight between the ith neuron of the layer $k-1$ and the jth neuron of the layer k. m is the learning time, and η is a proportionality factor known as the learning rate. In this study, to the third line of Eq.(2) an inertia term was added, which leads to

$$\Delta_p w_{i,j}^{k-1,k}(m) = -\eta \frac{\partial E_p}{\partial w_{i,j}^{k-1,k}} + \zeta \Delta_p w_{i,j}^{k-1,k}(m-1), \quad (3)$$

where ζ denotes the inertia rate. The inertia term is introduced for efficient learning convergence.

III. Pattern Recognition

As more demanding example for the BP learning, we consider a pattern recognition task, where 4 alphabetical letters (**B**, **U**, **C**, **S**) are fed into the neural network for recognition (see Fig. 5). The black and white box correspond to 1.0 and 0.0. In this case, the number of neurons in the input layer is 35 The number of neurons in output layer is fixed to 4, and we choose 20 hidden layer neurons. For recognition, a set of 25 patterns shifted 1 bit from each original pattern was prepared, leading a set of 100 patterns to be recognized. The learning rate and the inertia rate were set to $\eta = 0.8$ and $\zeta = 0.08$, respectively, and the initial weights were uniformly randomly chosen from the interval $[-1, 1]$. The learning time was set to $m = 10000$ steps. By using 100 different network initial conditions, we obtained reliable averages characterizing the network performance in terms of the recognition rate R defined as

$$R = \frac{Number\ of\ success}{Total\ number\ of\ input\ patterns} \times 100[\%]. \quad (4)$$

For culculating this equation, we use the following equation.

$$\begin{aligned} o_{pi} &= 1.0 \qquad (o_{pi} > T_b) \\ o_{pi} &= 0.0 \qquad (o_{pi} > T_w) \end{aligned} \quad (5)$$

Where, we fix the threshold values ($T_b = 0.9$, $T_w = 0.1$).

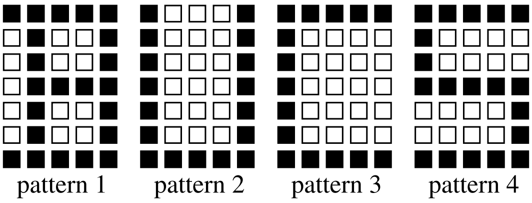


Fig. 5. Pattern recognition.

IV. Damaging Neurons (Zero output)

In this study, we investigate the generalization ability of AfNNs when some of neurons in the hidden layer are damaged. In order to realize damaging neurons, we consider that the connections of the damaged neurons to the output layer are temporally cut off (see Fig. 6). Namely, the damaged neurons do not operate. In this situation, we calculate the recognition rate for the several types of AfNNs.

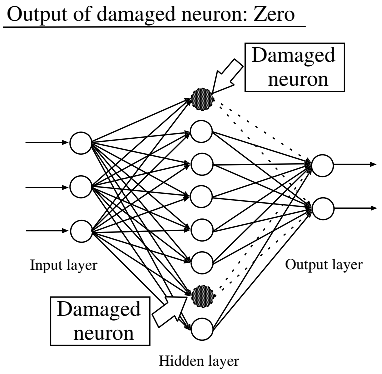


Fig. 6. Conceptual network model for damaging neurons (output: zero).

V. Simulation Results of AfNNs against Damaging Neurons

First, we explain the description of the network structure of AfNNs. For example, in the network (20-5), the neurons in the hidden layer is fixed with 20 and the number of the affordable neurons is 5.

We investigate the relationship between the performance of the AfNNs (AfNN-std and AfNN-ptn) and recognition rate when the number of affordable neurons is changed. We consider that 2 neurons are damaged (Zero output) after learning process. The simulation result of the AfNNs is shown in Fig. 7. The AfNN-std gains best performance when the number of the affordable neuron is 4. Also the AfNN-std obtains better results than the conventional network with wide range of the affordable neurons. While, the performance of the AfNN-ptn decrease by increasing the number of the affordable neurons.

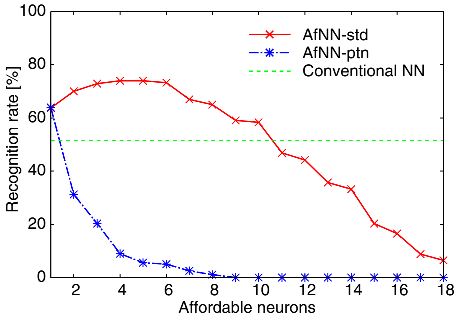
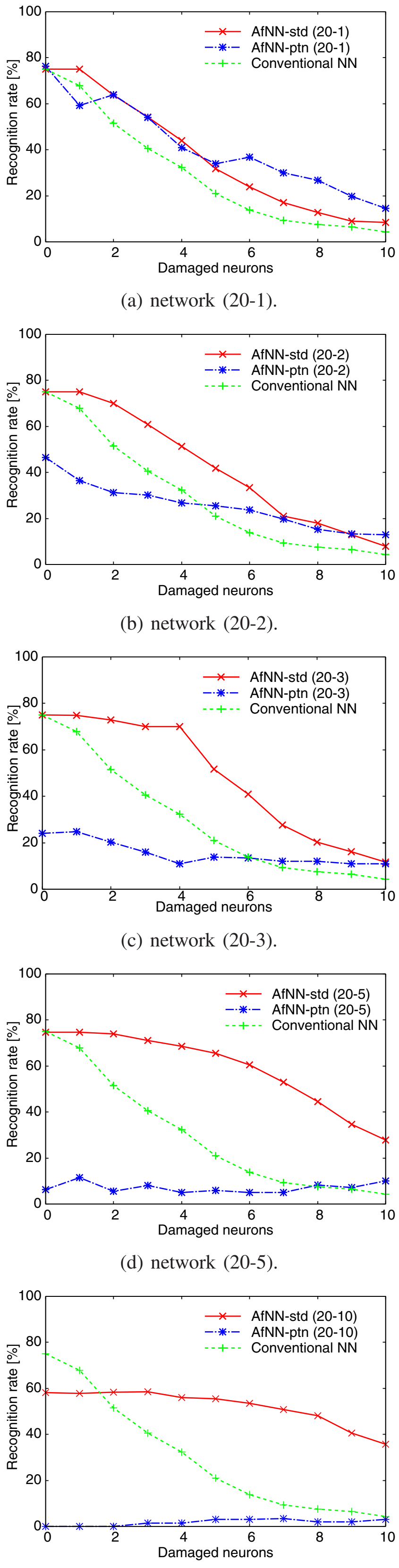


Fig. 7. Recognition rates by changing number of affordable neurons (Zero output).

Figure 8 shows the recognition rate of the several network structure of AfNN-std, AfNN-ptn and the conventional BP networks with the number of damaged neurons.

Figure 8(a) shows the recognition rate of the network structure (20-1). We infer that for this task, the recognition rates of the AfNNs and the conventional networks are similar, if only one neuron is damaged. Upon increasing the number of damaged neurons, the difference between the recognition rates,



(a) network (20-1).

(b) network (20-2).

(c) network (20-3).

(d) network (20-5).

(e) network (20-10).

Fig. 8. Recognition rate against damaging neurons (Zero output).

however, increases, leading to a noticeable higher recognition rate of the AfNNs, where the difference of recognition rate between the both networks is larger than 10 percent. When the number of damaged neuron is larger than 5, the recognition rate of AfNN-ptn is higher than AfNN-std.

Figures 8(b) and (c) show the recognition rate of the network structure (20-2) and (20-3). In these cases, we confirm that the AfNN-std can keep high recognition rate when the number of damaged neurons is smaller than 5. While, the performance of the AfNN-ptn becomes worse by increasing the number of affordable neurons.

The recognition rate of the network structure (20-5) and (20-10) is shown in Fig. 8(d) and (e). The AfNN-ptn can not recognize the input data even if the number of damaged neurons is small.

By comparison between the AfNN-std and the AfNN-ptn, we confirm that the performance of the AfNN-std is better than the AfNN-ptn and the conventional network. Namely, the learning process of the AfNN-std is cause of producing durability in the network.

VI. Conclusions

In this study, we have investigated the mechanism of learning process of the AfNN-std to make clear the reason of that the AfNN-std can perform well for learning and generalization abilities and operate as usually against damaging neurons. By computer simulations of the pattern recognition, we have confirmed that the learning process of the AfNN-std obtained better recognition rate than the AfNN-ptn.

For the future works, we apply the theoretical or systematic approaches to analyze the characteristics of the AfNNs and use more practical task as the learning example.

References

[1] Y. Uwate and Y. Nishio, "Performance of Affordable Neural Network for Back Propagation Learning," *IEICE Transactions on Fundamentals*, vol. E89-A, no. 9, pp. 473-478, Nov. 2005.

[2] Y. Sakurai, "Dependence of Functional Synaptic Connections of Hippocampal and Neocortical Neurons on Types of Memory," *Neuroscience Letters*, vol. 158, pp. 181-184, 1993.

[3] E. Vaadia, I. Haalman, M. Abeles, H. Bergman, Y. Prut, H. Slovin and A. Aertsen, "Dynamics of Neural Interactions in Monkey Cortex in Relation to Behavioral Events," *Nature*, vol. 373, pp. 515-518, Feb. 1995.

[4] Y. Sakurai, "Hippocampal and Neocortical Cell Assemblies Encode Memory Processes for Different Types of Stimuli in the Rat," *Journal of Neuroscience*, vol. 16, pp. 2809-2819, 1996.

[5] Y. Sakurai, "How Do Cell Assemblies Encode Information in the Brain?," *Neuroscience and Biobehavioral Reviews*, vol. 23, pp.785-796. 1999.

[6] C. H. Sequin and R. Clay, "Fault Tolerance in Artificial Neural Networks," *Proc. IEEE-INNS'90*, pp. 703-708, June. 1990.

[7] W. Hsieh and B. Sher, "Fault Tolerant Capability of Multi-Layer Perceptron Neural Network," *EUROMICRO'94*, pp. 644-650, 1994.

[8] V. Piuri, "Analysis of Fault Tolerance in Artificial Neural Networks," *Journal of Parallel and Distributed Computing*, vol. 61, pp. 18-48, Jan. 2001.

[9] Y. Uwate, Y. Nishio and R. Stoop, "Durability of Affordable Neural Networks against Damaging Neurons," *IEICE Transactions on Fundamentals*, vol. E92-A, no. 2, pp. 585-593, Feb. 2009.

[10] D.E. Rumelhart, G.E. Hinton and R.J. Williams, "Learning Internal Representations by Error Propagation," *Parallel Distributed Processing*, vol. 1, MIT Press, MA, pp. 318-362, 1986.