2010 International Symposium on Nonlinear Theory and its Applications
NOLTA2010, Krakow, Poland, September 5-8, 2010

Scale-Free Property of Affordable Neural Network

Yoshifumi Nishio and Yoko Uwate

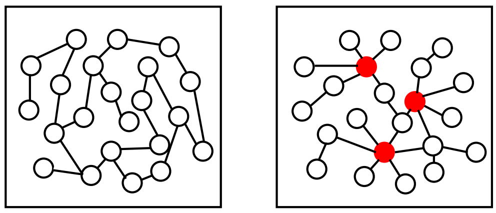
Dept. of Electrical and Electronic Engineering, Tokushima University
2-1 Minami-Josanjima, Tokushima, Japan
Email: {uwate, nishio}@ee.tokushima-u.ac.jp

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. We have confirmed that the AfNN gains good performance both of the generalization ability and the learning ability. In this study, we focus on the firng frequency of neurons in the hidden layer and the amount of weight changes during the learning process. By computer simulations, we confirm that the AfNN generates scale-free properties.

1. Introduction

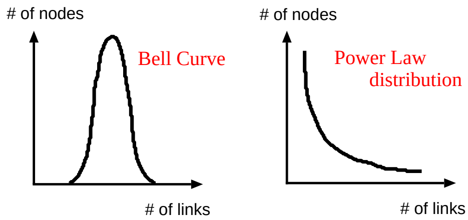
Since the scale-free networks were discovered by Barabasi et al. [1], studies assessing the influence of this property on the efficiency of networks have been carried out in various fields [2]-[4]. One way of how to characterize the difference between random and scale-free networks is by means of the distribution of the number of links which a node has. From Fig. 1, where we contrast the two network types, it is evident that in the scale-free network, although most nodes only have few connections, some nodes act as highly connected hubs. This distinction is captured in a more quantitative way by the distribution of the number of links vs. the number of nodes, as shown in Fig. 2. Random networks display a bell-shaped curve, implying that most nodes have the same number of links, and no highly connected nodes (see Fig. 2(a)). Scale-free networks, in contrast, often have many nodes with a few links only, whereas quite a few hubs exist that have a large number of links. Mathematically, scale-free networks are characterized by power law distributions (Fig. 2(b)). Because scale-rules emerge in many areas and disciplines of science (e.g. engineering, economics, social sciences and so on), we expect that also in the development of the science of complex networks, they will play an important role.

In a previous study on artificial neural networks, we proposed a new network structure with affordable neurons in the hidden layer, for efficient BP-learning [5]. We christened this network "Affordable Neural Network (AfNN)." In this network, some extra neurons are inserted into the hidden layer. By computer simulations [5], the AfNN has been confirmed to achieve an improved performance over conventional networks for BP-learning, in terms of speed of convergence and of learning efficiency. Moreover, we



(a) Random network. (b) Scale-free network.

Figure 1: Example of random and scale-free network.



(a) Random network. (b) Scale-free network.

Figure 2: Comparing random and scale-free distribution.

have investigated the performance of the AfNN for noise-polluted input data. We found that the AfNN is able to generate noise-cleaned outputs, which leads to the conclusion that the AfNN has the generalization property. Furthermore, the AfNN has a kind of durability, because the AfNN still performs well even if some of neurons in the hidden layer are damaged after learning process. We have confirmed that the AfNN can operates with keeping its efficiency against damaging neurons [6]. However, we believe that many advantageous characteristics of the AfNN are yet to be unveiled, and we also believe that the operation of the AfNN embodies important general features of the BP-learning process.

In this article, we investigate the scale-free property of the AfNN, which can be regarded as a complex network with time-varying connections, in order to clarify the relationship between the network topology and its information processing ability. First, we investigate the characteristics of the firng frequency of neurons in the hidden layer and the amount of weight changes during the learning process. Next, we introduce the scale-rule selection of affordable neurons and its effect is examined by computer simulations.

2. Affordable Neural Network (AfNN)

In [5], we introduced the AfNN to reflect important properties of the brain. During BP-learning, not all of the neurons in the hidden layer are used at every updating: some of the neurons are selected for the learning and the rest of the neurons are deactivated. The underlying network model is sketched in Fig. 3. The affordable neurons are selected by random every updating time (see Fig. 4). The definition of the affordable neurons is described as follows.

- The output of affordable neurons is set to zero.
- The weight vectors connected to the affordable neurons are not updated.

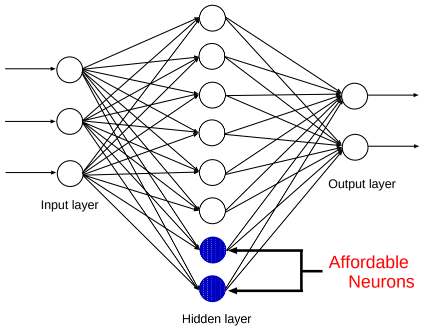


Figure 3: Affordable neural network.

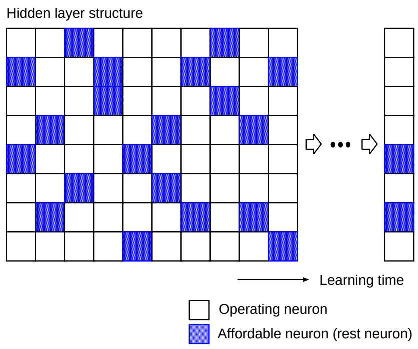


Figure 4: Random selection of affordable neurons.

3. Pattern Recognition by Random Selection

In this study, we use the batch BP-learning algorithm. The batch BP-learning algorithm can be expressed similarly to the standard BP-learning algorithm [7], with the difference lying in the timing of the weights updating.

As the first task, we consider the pattern recognition where 10 numeric characters (Fig. 5) are fed into the neural network for recognition.

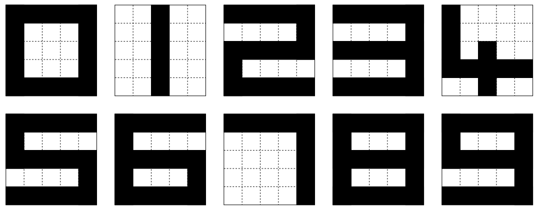


Figure 5: Pattern recognition.

In this case, the number of neurons in the input layer is 25, and we choose 100 hidden layer neurons. The number of neurons in the output layer is 10. The network learns these 10 numeric characters with 100000 iterations and the error curve converges to almost zero. The network parameters are fixed to η (learning rate)=0.05 and ξ (proportionality factor)=0.002. After learning process, the recognition rate is investigated by using the set of 10 patterns shifted 1 bit from each original pattern. We confirm that the learned networks can obtain around 70 to 90 percent recognition rate.

Next, we investigate the following characteristics of the AfNN:

1. Firing frequency for each neuron in hidden layer.
2. Amount of weight changes between input/hidden and hidden/output layers.

Figure 6 shows the simulation result of the frequency of firing neurons in the hidden layer by using multi trial. The threthold of firing is set to 0.5. This graph is shown with double logarithmic plot. From Fig. 6, the graph curve almost follows the straight line which means this graph has some scale-free properties.

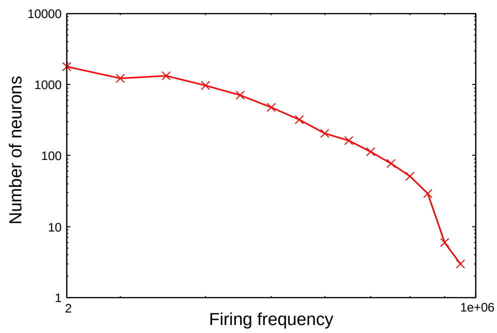


Figure 6: Frequency distribution for firing neurons in hidden layer.

Figure 7 shows the frequency distribution for total adjustment of weight vectors between input and hidden layers. In this case, the graph curves does not have any rules.

While in the case of he total adjustmen of weight vectors between hidden and output layers as shown in Fig. 8, the graph curve shows the straight line.

The total adjustment of weight vectors between input and hidden layers corresponds with the input pattern deeply. Then, the scale-free property does not occur for the total adjustment of weight vectors between input and hidden layers.

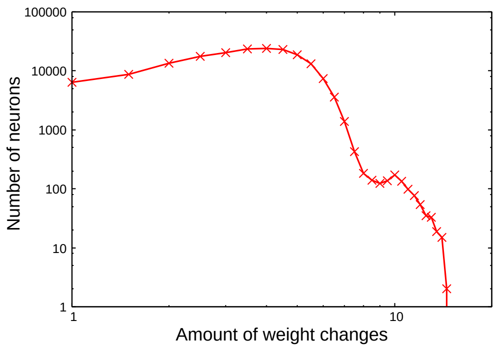


Figure 7: Frequency distribution for total adjustment of weight vectors between input and hidden layers.

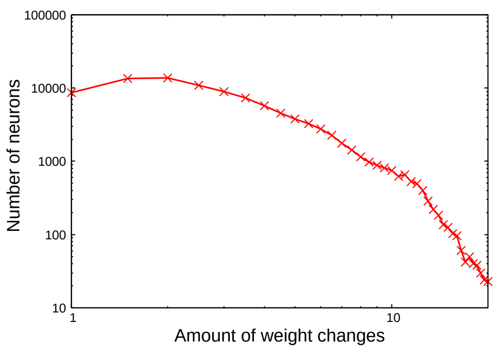


Figure 8: Frequency distribution for total adjustment of weight vectors between hidden and output layers.

4. Scale-Rule Selection of Affordable Neurons [7]

In this section, we introduce the simulation results of past study in [7]. The scale-rule selection of affordable neurons has been investigated. Our scale-rule selection procedure is described in terms of a parameter denoted by a vector S. The dimension s of S equals the number of neurons present in the hidden layer; each component of S corresponds to one single neuron indexed by i. The values of the components are evaluated in each update by

$$S_i = random()/i^2 \tag{1}$$

where $random()$ means the uniform random function producing values from 0.0 to 1.0. This implies that the neuron with the highest index will generally have a small value, whereas the first neuron will – unless the random function states something different – have a larger entry. Note that the values of the entries follow a power-law distribution. Using these values, we select in each update the set of active neurons according to the values of S. From the s neurons in the hidden layer, exactly the k neurons with the smallest entries are chosen as the affordable neurons. Figure 9 illustrates this scale-rule selection of the affordable neurons, where the number of the neurons in the hidden layer is set to be 100 and the number of affordable neurons is 20, 40 and 60, respectively. Our simulations will be based on 100000 updates. In this Figure, the horizontal axis indicates the neuron number, whereas the vertical axis displays the number of times the corresponding neuron was in the set of operating neurons. By this Figure it is confirmed that the operation time decreases gradually with the neuron number of the hidden layer. Furthermore, histograms of the number of neurons that have a given operation time are shown in Fig. 10. The resemblance with the scale-free distribution of Fig. 2(b) is evident, although, when inspected in details, the distribution is not of a simple power law type. For comparison let us also consider random selection of affordable neurons.

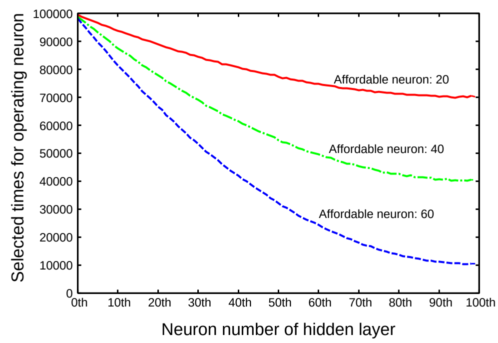


Figure 9: Scale-rule selection (Hidden: 100).

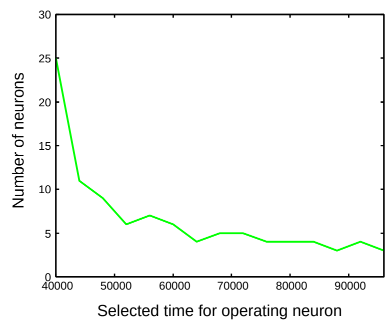
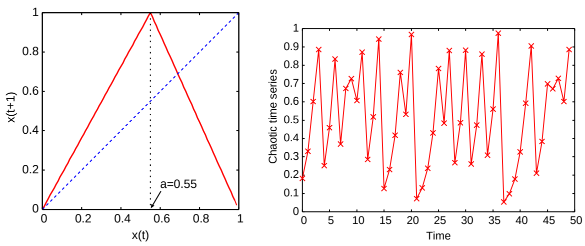


Figure 10: Distribution of scale-rule selection (Affordable neuron: 40, Hidden: 100).

For our simulations we want to teach our network to generate typical time series of the skew tent map. To this end, the network is trained – using time series of the tent map – to output, starting from given initial conditions, the same time series as the tent map would have generated.

The skew tent map and an example of time series are shown in Fig. 11. The length of chaotic time series is set to 50 steps; the size of the set of learning patterns is 20. In our approach, this requires the network to have 50 nodes in the input and the output layers. Each data is inputted to each node in the input layer. We carried out BP-learning by using the following parameters. The parameter of the learning rate and the inertia rate are fixed at $\eta = 0.05$ and $\zeta = 0.02$, respectively. The initial values of the weights are chosen between -1.0 and 1.0 at random. The learning time is set to 5000.



(a) Skew tent map. (b) Time series.

Figure 11: Skew tent map.

We investigate the learning efficiency as the average of the total error between the output and the desired target, when the network structure of the hidden layer is changed. The "Average Error E_{ave}" for this learning example is defined by the following equation.

$$E_{ave} = \frac{1}{P}\sum_{p=1}^{P}\left\{\frac{1}{2}(t_p - o_p)^2\right\} \tag{2}$$

We consider the case that the hidden layer consists of 100 neurons. The number of the affordable neurons is varied from 10 to 70. The results of this simulation are shown in Fig. 12, where the horizontal axes are the number of the affordable neurons and the vertical axes are E_{ave} for the pattern learning. From this Figure, we can confirm that the scale-rule selection method achieves a better performance if compare to the random selection. It is also seen that the difference between the errors of the scale-rule and the random selection networks increases with the number of affordable neurons. Even when the number of affordable neurons becomes large, the scale-rule selection network continues to show good learning ability. From this result, we can conclude that the scale-rule selection method of affordable neurons could play an important role for learning processes, in particular in biological systems.

5. Conclusions

In this study, we have investigated the characteristics of the firng frequency of neurons in the hidden layer and the amount of weight changes during the learning process. By

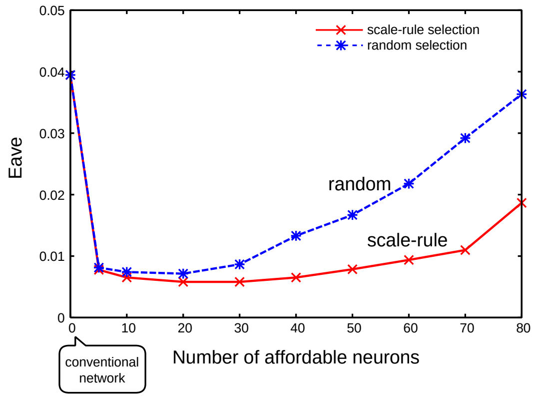


Figure 12: Learning ability by changing the number of affordable neurons (Number of neurons in hidden layer: 100).

computer simulations, we have confirmed that the AfNN could generate scale-free properties.

Acknowledgment

This work was partly supported by JSPS Grant-in-Aid for Scientific Research 22500203.

References

[1] Barabasi, Albert-Laszlo and A. Reka, "Emergence of Scaling in Random Networks," *Science*, 286:509-512, Oct. 1999.

[2] W.-X. Wang, B.-H. Wand, C.-Y. Yin, Y.-B. Xie and T. Zhou, "Traffic Dynamics Based on Local Routing Protocol on a Scale-Free Network," *Physical Review E*, vol. 73, no. 2, 026111, Feb. 2006. -

[3] W. Tam, F.C.M. Lau and C.K. Tse, "Complex-Network Modeling of a Call Network," *IEEE Transactions on Circuits Syst. I*, vol. 56, no. 2, pp. 416-429, Feb. 2009. -

[4] X. Zheng, F.C.M. Lau and C.K. Tse, "Constructing High-Rate Scale-Free LDPC Codes," *Proc. ISCAS'10*, pp. 3781-3784, May 2010.

[5] Y. Uwate and Y. Nishio, "Performance of Affordable Neural Network for Back Propagation Learning," *IEICE Transactions on Fundamentals*, vol. E89-A, no. 9, pp. 2374-2380, Sep. 2006.

[6] 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.

[7] 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.

[8] Y. Uwate, Y. Nishio and R. Stoop, "Scale-Rule Selection of Affordable Neural Network for Chaotic Time Series Learning," *Proc. ECCTD'07*, pp. 811-814, Aug. 2007.