2006 RISP International Workshop
on Nonlinear Circuits and Signal Processing (NCSP'06)
Waikiki Beach Marriott, Honolulu,
Hawaii, USA, March 3-5, 2006

Performance of Feedforward Neural Network with External Influence Function for Back Propagation Learning

Yoko Uwate †, Thomas Ott ‡, Yoshifumi Nishio † and Ruedi Stoop ‡

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

‡ Institute of Neuroinfomatics, University / ETH Zurich
Winterthurerstrasse 190, 8057 Zurich, Switzerland
Email: {tott, ruedi}@ini.phys.ethz.ch

1. Introduction

The human brain contains at least 100 billion neurons, each with the ability to influence many other cells. Clearly, highly sophisticated and efficient mechanisms are needed to enable communication among this astronomical number of elements. Such communication is made possible by synapses, the functional contacts between neurons. The synapses have a major part to play for learning, memory and control in the functional brain. The connection strengths of the synapses correspond to weights of neurons in artificial neural networks.

Back Propagation (BP) learning [1] is one of engineering applications of artificial neural networks. The BP learning operates with a feedforward neural network which is composed of an input layer, a hidden layer and an output layer, and the effectiveness of the BP learning has been confirmed in pattern recognition, system control, signal processing, and so on [2]-[4]. It is an optimization produce based on gradient descent that adjusts weights to reduce the system error or cost function. The errors are used as inputs to feedback connections from which adjustments are made to the synaptic weights layer by layer in a backward direction.

In this study, we investigate the performance of the BP learning if the action of synaptic weights changing by external influence. We introduce periodically sine wave as external influence function to adjustment of the weights. By computer simulation, the proposed network gains the good performance for learning efficiency. Furthermore, the characteristics of the neurons in hidden layer are investigated.

2. Back Propagation Algorithm

The standard BP learning algorithm was introduced in [1]. The BP is the most common learning algorithm for feedforward neural networks. In this study, we use the batch BP learning algorithm. The batch BP learning algorithm is expressed by similar formula of the standard BP learning algorithm. The difference lies in the timing of the update of the weight. The update of the standard BP is performed after each single input data, while the update of the batch BP is performed after all different input data. The total error E of the network is defined as the following equation.

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

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 weights between all layers of the network to minimize the total error E. In order to minimize E, the weights are adjusted according to the following equation:

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

where $w_{i,j}^{k-1,k}$ 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, we introduce inertia term in the 2nd term of the right-hand side of Eq. (4) to improve the rate of convergence.

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

where ζ denotes the inertia rate. This can be accomplished by adding a fraction of the previous weight change to the current weight change. The addition of such a term can help smooth out the descent path by preventing extreme changes in the gradient due to local anomalies. It can act as an average effect which smooths the trajectory of the gradient as it moves downhill.

3. External Influence Function

We investigate the performance of BP learning when the adjustment of the weights changes periodically by external influence. The adjustment of weights equation with external influence function is given by the following equation.

$$\begin{aligned} &\Delta_p w_{i,j}^{k-1,k}(m) \\ &= f_E(m) \times \left(-\eta \frac{\partial E_p}{\partial w_{i,j}^{k-1,k}} + \zeta \Delta_p w_{i,j}^{k-1,k}(m-1) \right) \end{aligned} \tag{4}$$

where f_E is the external influence function as shown in Fig. 1. In this study, we consider that the external influence function is sine wave. From this figure, A_1 and A_2 denote the amplitude of this function, and T is the period of the sine wave. We set $T = 1000$.

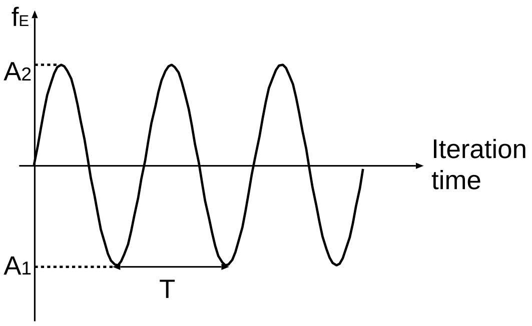


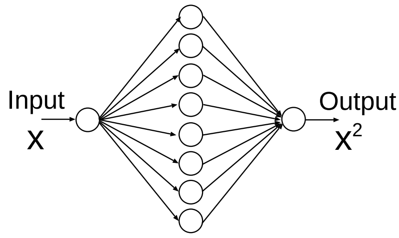
Figure 1: External influence function.

4. Simulated Results

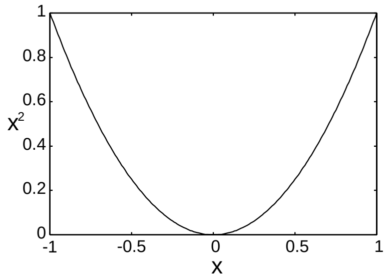
We consider the feedforward neural network producing outputs x^2 for inputs data x as one learning example. The sampling range of the input data is $[-1.0,\ 1.0]$ and the step size of the input data is set to be 0.01. We carried out the BP learning by using the following parameters. The learning rate and the inertia rate are fixed as $\eta = 0.1$ and $\zeta = 0.02$, respectively. The initial values of the weights are given between -1.0 and 1.0 at random. The learning time is set to 50000, and the 8 neurons are prepared in the hidden layer. The network structure using this study and learning example are shown in Fig. 2.

4.1. Performance of learning process

We investigate the convergence speed and the learning efficiency as the total error between the output and the desired



(a) Network structure.



(b) Learning example.

Figure 2: Network structure and learning example.

target. We define "Average Error E_{ave}" by the following equation as mean square error.

$$E_{ave} = \frac{1}{P} \sum_{p=1}^{P} \left\{ \frac{1}{2} (t_p - o_p)^2 \right\} \tag{5}$$

Figure 3 shows one example of the simulation results when the amplitude of the external influence functions are changed. The horizontal axis is iteration time and the vertical axis is E_{ave}. From this figure, we can confirm that the proposed network with the external influence function gains better performance than the conventional network. The learning curve converges more quickly when the external influence has large amplitude. E_{ave} of the proposed network and the conventional network are summarized in Tab. 1. The results show that the proposed network gains a good performance when the amplitude of the external influence function is large.

Table 1: E_{ave} of the after BP learning process.

	A_1, A_2	E_{ave}
Proposed	0.1, 2.0	0.026374
	0.1, 5.0	0.000987
	0.1, 10.0	0.000826
	0.1, 20.0	0.000750
Conventional		0.028067

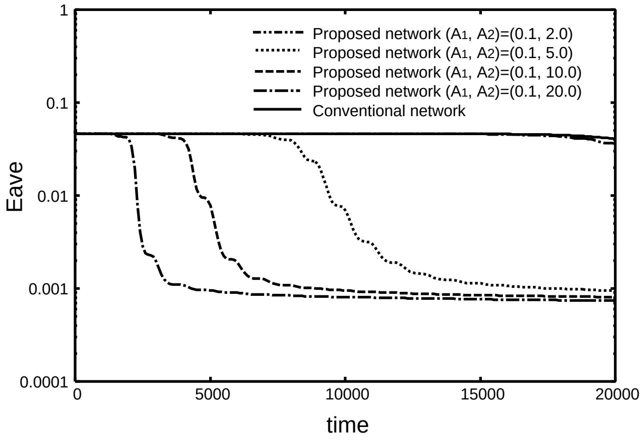


Figure 3: One example of learning process.

4.2. Stress and satisfaction

Next, the adjustments of the weights of the proposed network and the conventional network are compared. We define that the network feel stressed if the adjustment of the weights of the proposed network is smaller than the conventional network. On the other hand, the network feel satisfied if the adjustment of the weights of the proposed network is larger than the conventional network. The definition of the stress and the satisfaction are shown in Fig. 4.

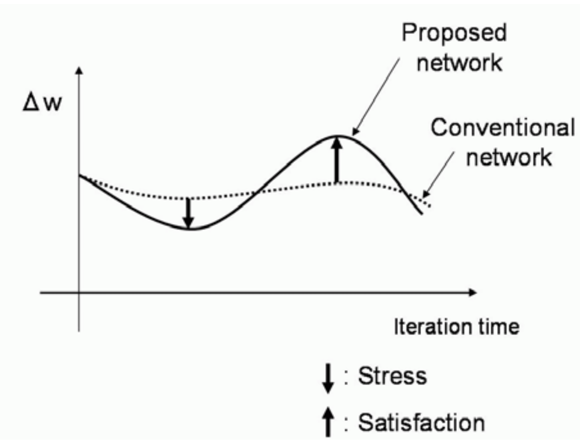
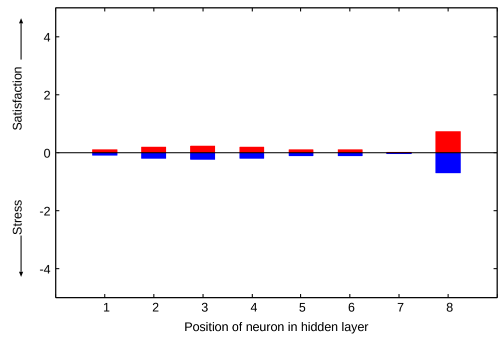
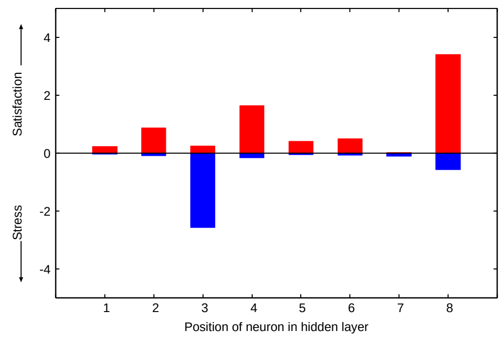


Figure 4: Stress and satisfaction of the proposed network.

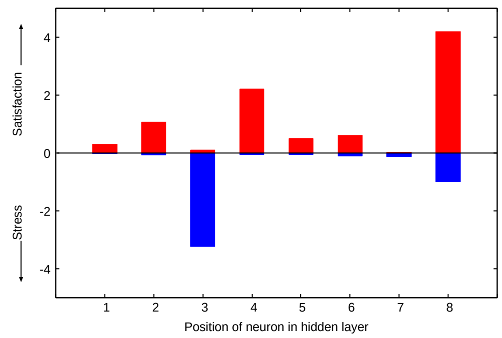
The simulated results compared the area ratio of the stress and the satisfaction are shown in Fig. 5. In this simulation, we use the same initial conditions of the weights. We confirm that the area of the stress of the satisfaction is small when the amplitude of the external influence function is small (Fig. 5 (a)). And the some neuron's area of the stress or the satisfaction becomes large by increasing with amplitude of the external influence function (Figs. 5 (b), (c) and (d)).



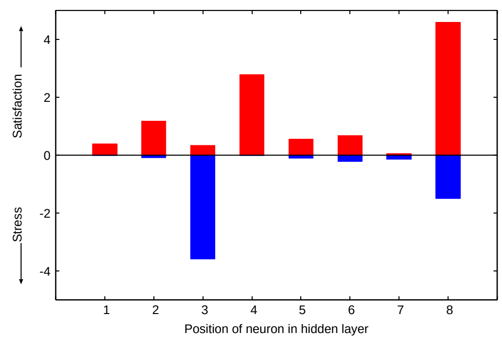
(a) A_1=0.1, A_2=2.0.



(b) A_1=0.1, A_2=5.0.

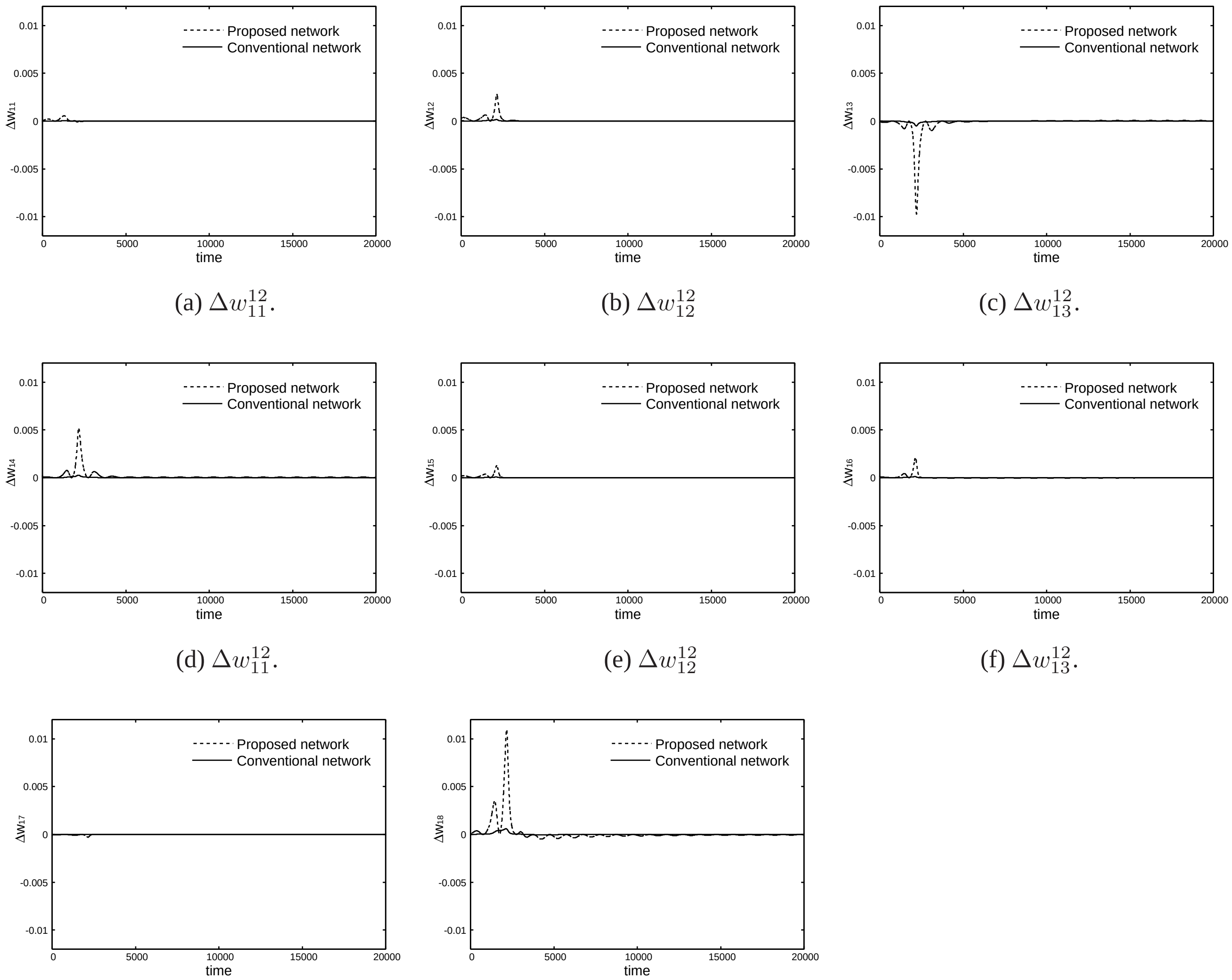


(c) A_1=0.1, A_2=10.0.



(d) A_1=0.1, A_2=20.0.

Figure 5: Simulated results of stress and satisfaction.



(a) Δw^{12}_{11}. (b) Δw^{12}_{12} (c) Δw^{12}_{13}.

(d) Δw^{12}_{11}. (e) Δw^{12}_{12} (f) Δw^{12}_{13}.

(g) Δw^{12}_{11}. (h) Δw^{12}_{12}.

Figure 6: Adjustment of weights (A_1=0.1, A_2=20.0).

One example of varying adjustments of the weights between the input and the output layers is shown in Fig. 6. The adjustments of some weights of the proposed network change widely when the iteration time is around 2000.

5. Conclusions

In this study, we investigated the performance of the BP learning if the action of synaptic weights changing by external influence. We introduced the periodically sine wave as external influence function to the adjustment of the weights. By computer simulation, the proposed network gains the good performance for learning efficiency. Furthermore, the characteristics of the neurons in the hidden layer were investigated.

References

[1] 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.

[2] Y. LeCun, B. Boser, J.S. Denker, et al, "Handwritten Digit Recognition with a Back-Propagation Network," *Advances in Neural Information Processing Systems*, vol. 2, MIT Press, MA, 1990.

[3] M. Kawano, Y. Uno, M. Isobe and R. Suzuki, "A Hierarchical Neural Network Model for Voluntary Movement with Application to Robotics," *IEEE Control Systems Magazine*, vol. 8, pp. 8-16, 1988.

[4] T.J. Sejnowski and C.R. Rosenberg, "Parallel Networks that Learn to Pronounce English Text," *Complex Systems*, vol. 1, pp. 145-168, 1987.