2008 IEEE Workshop on
Nonlinear Circuit Networks, Tokushima, Dec 5-6, 2008

Chaotic BP Algorithm for Sine Wave Function Learning with Different Frequencies

Azian Azamimi and Yoshifumi Nishio
Dept. of Electrical and Electronic Eng.,
Tokushima University,
Tokushima, 770-8506 JAPAN
Email:azian, nishio@ee.tokushima-u.ac.jp

Yoko Uwate
Institute of Neuroinfomatics,
University / ETH Zurich, Switzerland
Email: yu001@ini.phys.ethz.ch

***Abstract*— Stochastic resonance is observed when noise added to a system improves the systems performance in some ways. Using this concept, we apply noise to neural networks in order to observe their effect on network performance. In our previous research, we have proposed a new modified back propagation (BP) learning algorithm, in which chaotic noise is added into weight update process [1][2]. Computer simulation results show that the addition of chaotic noise during weight update process can give a faster learning performance and a better convergence rate compared to the standard BP algorithm. Hence, in this study, we extend our investigation by simulating our proposed network to learn sine wave function with different frequencies as one learning problem. We observe and analyze the network performance in order to confirm the generality of this work.**

I. Introduction

Backpropagation (BP) learning 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 single or more of hidden layers and an output layer. The effectiveness of BP learning has been recognized in many engineering applications especially in pattern recognition, system control and signal processing. Although the BP learning has been a significant research area of artificial neural network, it also has been known as an algorithm with a poor convergence rate. Many attempts have been made to the algorithm to improve the performance on convergence speed and learning efficiency. For example, there have been a lot of reports on changing the learning rate and also the number of neurons in the hidden layer but this will lead to slight improvement only [3]-[5]. Not many studies have been made on modifying the algorithm structure in order to improve the learning performance.

On the other hand, chaos has gained much attention and some applications in neural network over this recent years. There have been many reports on the good performance of Hopfield neural network when chaos is inputted to the neurons as noise [6]. By computer simulations, it has been confirmed that chaotic noise is effective for solving quadratic assigment problem and gains better performance to escape out local minima than random noise. The authors [7] also have proposed the feedforward neural network with chaotically oscillating gradient of the sigmoid function and proved that the proposed network can find good solutions in early time. Hence, we consider that various features of chaos can give a good effect in neural network. Network performance can be improved when chaotic noise is applied to neural networks.

In this study, we conduct a further investigation by simulating our proposed network to learn sine wave function with different frequencies. Different frequencies of sine wave function may give different learning results and we analyze the network performance.

II. BP Learning Algorithm

A. BP batch learning algorithm

In the standard backpropagation learning algorithm, the errors of output neurons are backpropagated through the network during training. This standard learning algorithm was introduced in [8]. The error signal of output neuron can be defined by taking the difference between the target output and the actual output. However, 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 but the difference lies in timing of the weight update. The weight update of the standard BP is performed after each single input data, while for the batch BP the weight update is performed after all input data has been processed. The total error E of the network is defined as follows;

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

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 so that the total error E can be minimized. In order to minimize the total error E, the weights are adjusted according to the following equation:

$$w_{i,j}^{k-1,k}(m+1) = w_{i,j}^{k-1,k}(m) + \sum_{p=1}^{P} \triangle_p w_{i,j}^{k-1,k}(m), \tag{2}$$

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

where $w_{i,j}^{k-1,k}$ is the weight between ith neuron of the layer $k-1$ and the jth neuron of the layer k, m is the learning time and η is the learning rate. In this study, we add the inertia term to Eq.(3) where ζ denotes the inertia rate.

$$\triangle_p w_{i,j}^{k-1,k}(m) = -\eta \frac{\partial E_p}{\partial w_{i,j}^{k-1,k}} + \zeta \triangle_p w_{i,j}^{k-1,k}(m-1), \quad (4)$$

B. Chaotic BP learning algorithm

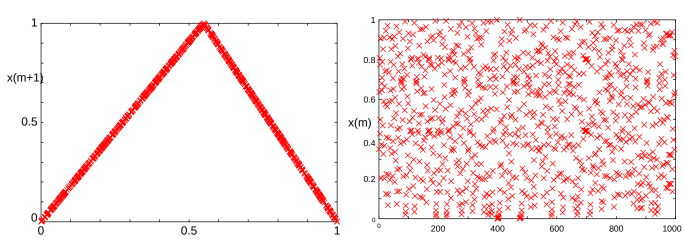
In our previous research, we have proposed a new modified BP learning algorithm, namely chaotic BP learning algorithm. This new algorithm can be expressed by a similar formula with the standard BP algorithm but the different lies in weight update process. Chaotic noise is added into weight update process during error propagation. The weight update process for chaotic BP learning algorithm can be shown as follows, which β denotes the noise amplitude.

$$\begin{aligned} \triangle_p w_{i,j}^{k-1,k}(m) &= -\eta \frac{\partial E_p}{\partial w_{i,j}^{k-1,k}} \\ &+ \zeta \triangle_p w_{i,j}^{k-1,k}(m-1) + noise_{i,j}(m), \end{aligned} \quad (5)$$

$$noise_{i,j}(m) = \beta_{i,j}(m)(x_{i,j}(m) - 0.5), \quad (6)$$

Chaos, $x_{i,j}(m)$ can be generated by various chaotic maps namely logistic map, skew tent map, Benoulli shift map and many others. Previously, we have investigated chaos features using logistic map and confirmed that our improved algorithm can give better convergence rate. In this study, we use other chaotic map which is skew tent map to generate chaos. The skew tent map and an example of time series obtained by Eq.(7) are shown in Fig. 1. α is the control parameter of chaos and one of the most important parameters for this work.

$$x_{i,j}(m+1) = \begin{cases} \frac{x_{i,j}(m)}{\alpha} & (0 \leq x_{i,j}(m) \leq \alpha) \\ \frac{x_{i,j}(m)-1}{\alpha-1} & (\alpha < x_{i,j}(m) \leq 1) \end{cases} \quad (7)$$



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

Fig. 1. Skew tent map.

III. Simulation Results

In this section, we show the effectiveness of this chaotic BP algorithm by testing it to a learning problem. Here, we consider the feed forward neural network produce outputs $0.5\sin(2\pi fx + \pi/6) + 0.5$ for input data x as one learning example. The sampling range of the input data is [-1.0, 1.0]. We carried out the chaotic BP learning algorithm by using the following parameters. The learning rate and the inertia rate are fixed as η = 0.1 and ζ = 0.001 respectively. The initial values of the weights are given between -1.0 and 1.0 at random. The learning iterations is set to 50000 and 8 neurons are prepared in the hidden layer (Fig. 2). For all simulation process, we use fixed value of chaos parameter (α=0.55) and noise amplitude (β=0.01) to investigate the learning perforamance for difference frequencies of sine wave function. The learning example of sine wave function $0.5\sin(2\pi fx + \pi/6) + 0.5$ is shown in Fig. 3 for $f = 1.0[Hz]$.

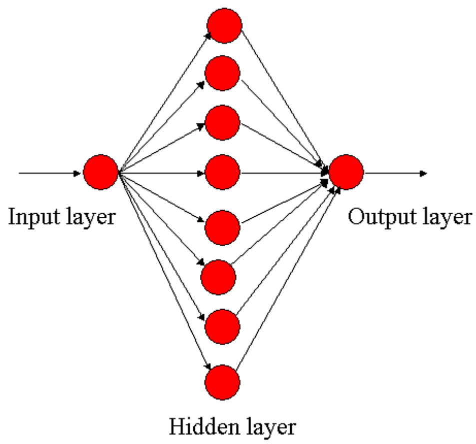


Fig. 2. Network structure.

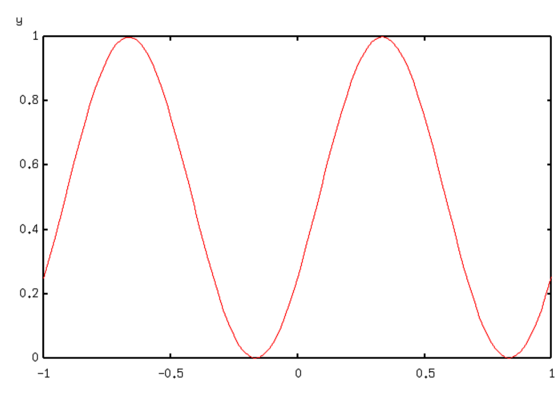


Fig. 3. Learning example of sine wave function $0.5\sin(2\pi x + \pi/6) + 0.5$.

It should be noted that although there maybe more suitable values of these parameters, it is difficult to set them theoret-

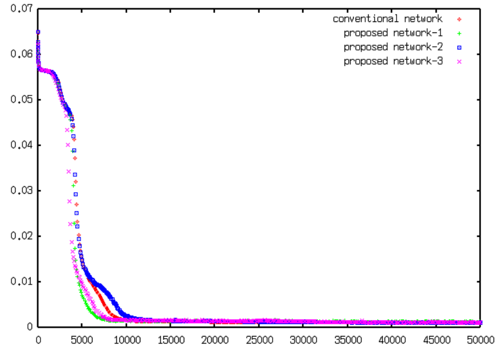


Fig. 4. Learning example of sine wave function $0.5 \sin(2\pi x + \pi/6) + 0.5$.

ically. Hence, we use the same parameters during all process to make it easier to analyze the simulation results.

A. Noise added to different weight update positions

First, we investigate the learning efficacy by adding chaotic noise into different position of weight update. As we may know, in BP learning algorithm, the main purpose of weight update is to reduce the error value between output and desired target. Here, we choose one sine wave function with fixed frequency ($f = 1.0[Hz]$) to see the effectiveness of noise adding into different position of weight update. We add chaotic noise into three different positions of weight update ; a) Proposed network-1 (from input layer to hidden layer only), b) Proposed network-2 (from hidden layer to output layer only) and c) Proposed network-3 (both of them). We also compare the learning performance with the standard BP algorithm which there is no chaotic noise application at all (conventional network). Figure 4 show the result performance of all proposed and conventional network for sine wave function learning example. The horizontal axis is iteration time and the vertical axis is error value.

From this figure, we confirm that our proposed network which chaotic noise is added into the weight update gains better performance than the conventional network when chaos parameter of α is set to 0.55 and noise amplitude β is fixed as 0.01. The addition of chaotic noise during weight update can help the learning process to find a good solution in early time compared to the standard BP algorithm but the improvement rate is very small. Hence, we investigate the learning performance if different frequencies of sine wave function are used. Different frequencies of sine wave function may give different learning results and we analyze the network performance.

B. Different frequencies of sine wave function

We observe the network performance by simulating it to various sine wave function with different frequencies. Figure 5 show the learning performance when $f[Hz]$ is set to 0.5, 1.0, 2.0, 3.0, 4.0 and 5.0 respectively.

From these results, we can see that our proposed network give better improvement rate when $f[Hz]$ is set to 2.0 and 3.0. If $f[Hz]$ is set to 0.5, there is no improvement at all between our proposed network and conventional BP algorithm. If frequency value is too large, the learning performance become not stable and cannot find a good solution in early time. We can see the bad learning results if $f[Hz]$ is set to 5.0. The more difficult a learning function is, the more complicated the learning process will be. Hence, we show the result of all frequency values for proposed network-3 (Fig. 6). We understand that a large value of frequency need more time to solve the learning problem compared to small frequency values.

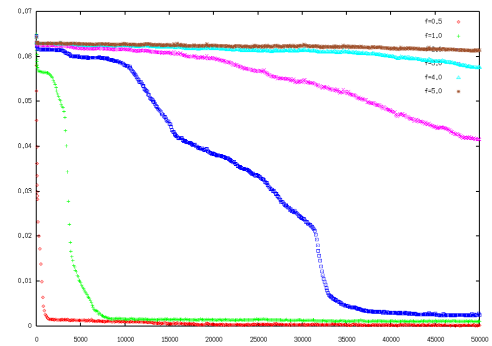


Fig. 6. Learning performance of sine wave function with different frequencies for proposed network-3.

IV. Conclusions

In this study, we have conducted a further investigation of our proposed network to learn sine wave function with different frequencies. From simulation results, we confirmed that our proposed network can improve the learning performance compared to the standard BP algorithm. We also discovered that learning process will take a longer time if more difficult learning function is used.

References

[1] A. Azamimi, Y. Uwate and Y. Nishio, “Good Learning Performance of Backpropagation Algorithm with Chaotic Noise Features,” *Proceedings of SSJW”08*,pp. 36-38, Aug. 2008.

[2] A. Azamimi, Y. Uwate and Y. Nishio, “An Analysis of Chaotic Noise Injected to Backpropagation Algorithm in Feedforward Neural Network,” *Proceedings of IWVCC’08*, pp. 70-73, Nov 2008.

[3] A. Sperduti and A. Starita, “Speed up learning and network optimization with extended back-propagation,” *Neural Networks*, vol.6, pp.365-383, 1993.

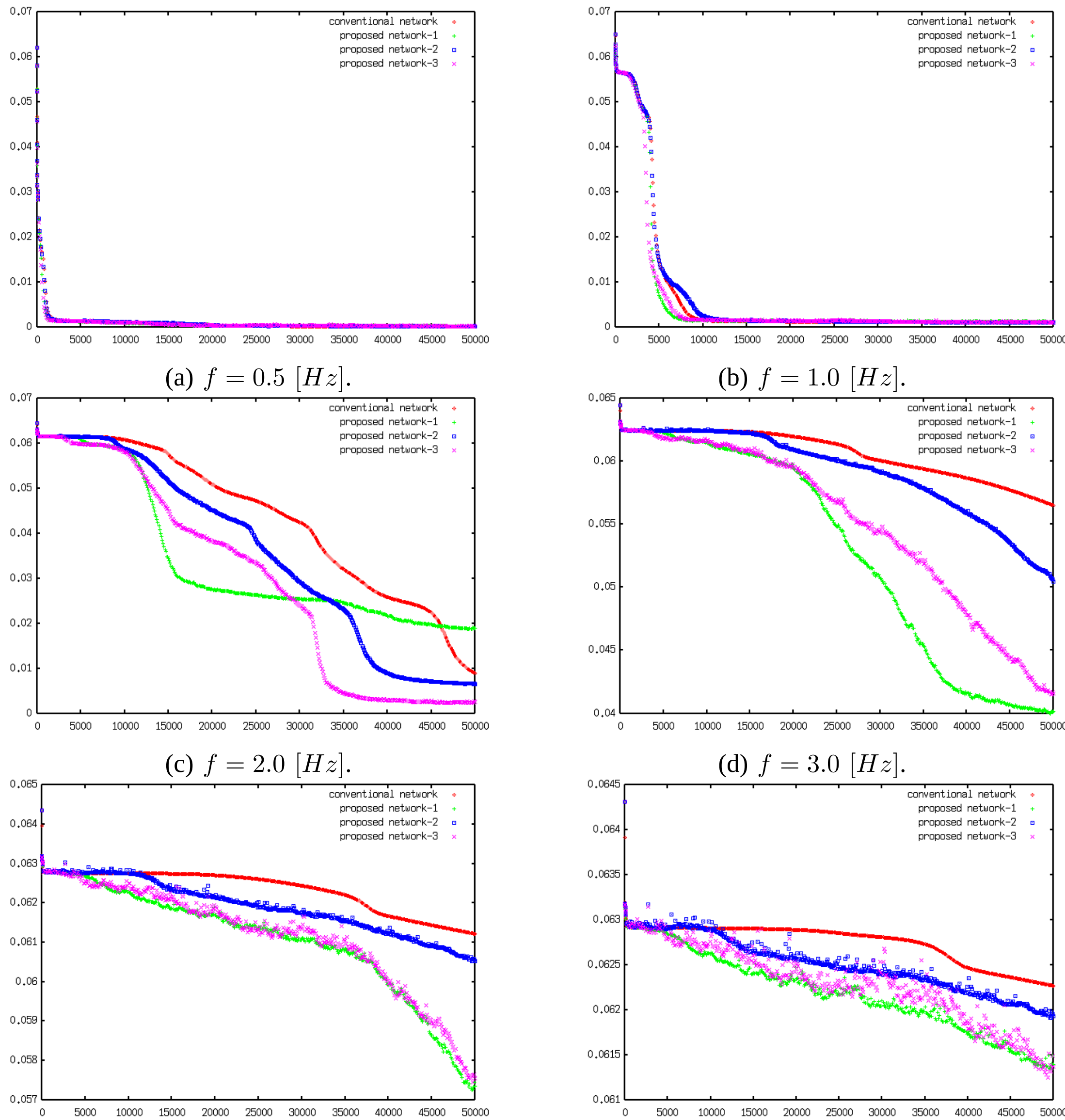
[4] A. Miniani and R.D. Williams, “Acceleration of backpropagation through learning rate and momentum adaption,” *Proceedings of IJCNN’90*,San Diego,CA, pp.676-679, 1990.

[5] R.A. Jacobs, “Increased rates of convergence through learning rate adaption,” *Neural Networks*, vol.1, pp.295-308, 1988.

[6] T. Ueta, Y. Nishio and T. Kawabe, “Comparison between Chaotic Noise and Burst Noise on Solving Ability of Hopfield Neural Networks,” *Proceedings of NOLTA’97*, vol.1, pp.409-412, 1997.

[7] Y. Uwate and Y. Nishio, “Chaotically Oscillating Sigmoid Function in Feedforward Neural Network,” *Proceedings of NOLTA’06*, pp.215-218, 2006.

[8] 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.



(a) $f = 0.5$ $[Hz]$. (b) $f = 1.0$ $[Hz]$.

(c) $f = 2.0$ $[Hz]$. (d) $f = 3.0$ $[Hz]$.

(e) $f = 4.0$ $[Hz]$. (f) $f = 5.0$ $[Hz]$.

Fig. 5. Learning performance of proposed and conventional network for sine wave function with different frequencies.