Good Learning Performance of Backpropagation Algorithm with Chaotic Noise Features

Azian Azamimi†, Yoko Uwate‡ and Yoshifumi Nishio†

†Department of Electrical and Electronic Engineering
Tokushima University
2-1 Minami-Josanjima, Tokushima 770-8506, Japan
‡Institute of Neuroinfomatics,
University / ETH Zurich, Switzerland
Email: {azian, nishio}@ee.tokushima-u.ac.jp, yu001@ini.phys.ethz.ch

Abstract—Over the years, many improvements and modifications of the backpropagation learning algorithm have been reported. In this study, we propose a new modified backpropagation learning algorithm by adding the chaotic noise into weight update process. By computer simulations, we confirm that the proposed algorithm can gives a better convergence rate and can find a good solution in early time compared to the conventional backpropagation algorithm. Weight update position, noise amplitude and control parameter of chaos can give a big effect on the backpropagation learning performance.

1. Introduction

In the area of artificial neural networks, the backpropagation (BP) learning algorithm has proved to be efficient in many engineering applications especially in pattern recognition, signal processing and system control. The BP learning operates with a feedforward neural network which consists of an input layer, a single or more of hidden layer and an output layer. Although the BP learning has been a significant research area of neural network, it also has been known as an algorithm with a very poor convergence rate. Many attempts have been made to the algorithm to improve the performance on convergence speed and learning efficiency.

On the other hand, chaos has gained much attention and some applications in neural network during recent years. There have been many reports on the good performance of Hopfield neural network when chaos is inputted to the neurons as noise [1],[2]. By computer simulations, it has been confirmed that the chaotic noise is effective for solving quadratic assigment problem and gains better performance to escape out local minima than random noise. The authors [3] 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. Considering this, we try to apply some of chaotic features to the backpropagation learning algorithm.

In this study, we propose a new modified backpropagation learning algorithm by adding the chaotic noise into weight update process. The chaotic noise is generated by logistic map and is controlled by noise amplitude.

2. BP learning algorithm

2.1. 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 [4]. 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;

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

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 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:

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

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

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.

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

2.2. New modified BP learning algorithm

Slow convergence rate is the most common disadvantage of backpropagation learning algorithm. In the backpropagation algorithm, the weights updates are proportional to the error propagating from output through the sigmoid function and through the weights. The convergence rate of the learning process can be improved by changing how the error propagates back through the network. We believe that the weight update calculation is the main factor to improve the learning process. Considering this, we add the chaotic noise into weight update process during error propagation. The new modified weight update equation for BP algorithm can be shown as follows where β is the noise amplitude.

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

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

The chaotic noise is generated by logistic map and α is the control parameter of chaos. An example of time series obtained by Eq.(7) can be shown as Fig. 1.

$$x_{i,j}(m+1) = \alpha x_0(m)(1 - x_0(m)), \quad (7)$$

3. Simulation results

In this section, we show the effectiveness of chaotic noise added into weight update process in BP algorithm by testing it to the learning example. Here, we consider the feedforward neural network produce outputs x^2 for input data x as one learning example. The sampling range of the input data is [-1.0, 1.0]. We carried out the new modified BP learning algorithm by using the following parameters. The learning rate and the inertia rate are fixed as $\eta = 0.02$ and $\zeta = 0.002$ respectively. The initial values of the weights are given between -1.0 and 1.0 at random. The learning iterations is set to 10000 and 6 neurons are prepared in the hidden layer. The network structure and learning example are shown in Fig. 2

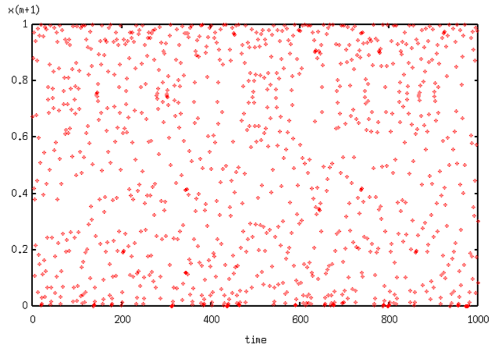
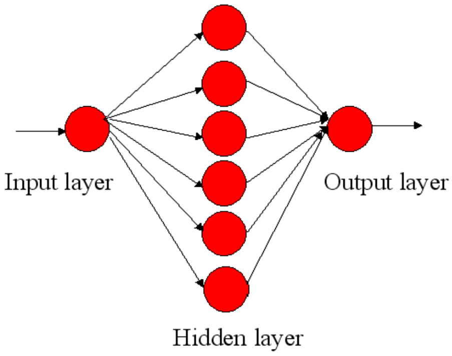


Figure 1: Time series obtained from logistic map (α=4.00).

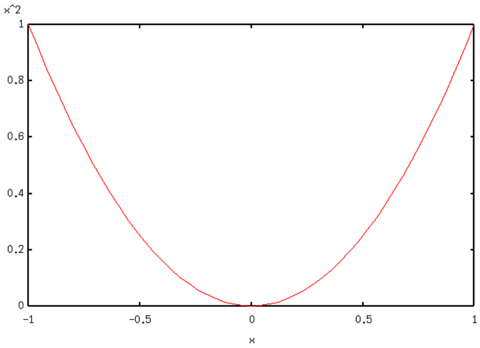
It should be noted that although there maybe more suitable values of these parameters, it is diificult to set them theoretically. Hence, we use the same parameters during all simulations process to make it easier to compare between the new proposed algorithm and the conventional method.

First, we investigate the learning efficiency by adding the chaotic noise into different position of weight update process. As we may know, in backpropagation learning algorithm, the main purpose of weight adjustment is to reduce the error value between the output and desired target. Thus, we apply the chaotic noise to different position of weight update in order to see the effectiveness of noise adding. We add chaotic noise to 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 conventional method where there is no noise application at all. Figure 3 shows the learning performance of all proposed and conventional network respectively. The horizontal axis is iteration time and the vertical axis is error value.

From this figure, we can confirm that the new proposed algorithm where chaotic noise is added into weight update process gains better performance than the conventional algorithm when the chaos parameter of α is set to 4.0 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 conventional backpropagation algorithm. Futhermore, the different position



(a) Network structure.



(b) Learning example of x^2.

Figure 2: Network structure and learning example.

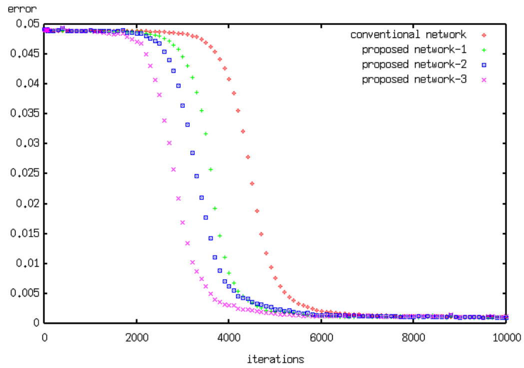


Figure 3: Learning performance for proposed and conventional network.

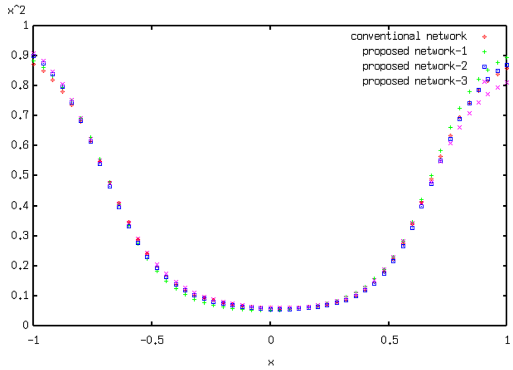


Figure 4: Output results for proposed and conventional network.

of weight update also give the different performance. From the simulation result, we can confirm that when the chaotic noise is applied to both sides of weight update (from input layer to hidden layer and from hidden layer to output layer), the learning process gain better performance compared to others. Figure 4 also show the output results for all proposed and conventional method.

4. Conclusions

In this study, we proposed a new modified backpropagation learning algorithm by adding the chaotic noise during weight update. By computer simulations, we confirmed that addition of chaotic noise to the weight update in learning algorithm can gives a better convergence rate and can find a good solution of the learning process in early time compared to the conventional backpropagation algorithm. Different weight update position, the amplitude of noise and the control parameter of chaos can give a big effect on the backpropagation learning performance.

References

[1] 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.

[2] Y. Tada, Y. Uwate and Y. Nishio, “Performance of Chaotic Switching Noise Injected to Hopfield NN for Quadratic Assignment Problem,” *Proceedings of ISCAS'06*, pp.5519-5522, 2006.

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

[4] 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.