2009 International Workshop on Nonlinear Circuits and Signal Processing
NCSP'09, Waikiki, Hawaii, March 1-3, 2009

An Improvement in Pattern Recognition Problem using Chaotic BP Learning Algorithm

Azian Azamimi†, Yoko Uwate‡ and Yoshifumi Nishio†

†Tokushima University
2-1 Minami-Josanjima, Tokushima, Japan
Phone/FAX:+81-88-656-7501
Email: azian, nishio@ee.tokushima-u.ac.jp

‡Institute of Neuroinfomatics,
University / ETH Zurich, Switzerland
Phone/FAX:+41 (0)44 6353053
Email: yu001@ini.phys.ethz.ch

Abstract

Pattern recognition is one of the most difficult problems and somehow impossible to be solved by conventional computers. However, it can be solved by using a feed forward neural network that has been trained accordingly. In our previous research, we have proposed a new modified BP algorithm, namely Chaotic BP learning algorithm [1][2]. Simulated results show that our proposed algorithm obtain a good solution in early time when simple learning examples are used. Hence, in this study, we implement our proposed algorithm to learn a pattern recognition task which is an example of difficult learning examples. By computer simulations, we confirmed that our proposed algorithm can give a better recognition rate and less iteration time compared with standard BP algorithm.

1. Introduction

Back Propagation (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]-[8]. 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. Hence, we consider that various features of chaos can give a good effect in neural network.

In our previous research, we have proposed a new modified BP learning algorithm (Chaotic BP), in which chaotic noise is added into weight update process during error propagation. 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 with standard BP algorithm. Hence, in this study, we extend our investigation by using our proposed algorithm to learn a pattern recognition problem. We also investigate on various factors such as the role of hidden neurons and certain parameters which might give different results to the learning performance.

2. Chaotic BP Learning Algorithm

The chaotic BP algorithm can be expressed by a similar formula with the standard BP algorithm but the different lies in weight update process. In the standard BP learning algorithm, the errors of output neurons are backpropagated through the network during training. This standard learning algorithm was introduced in [9]. The error signal of output neuron can be defined by taking the difference between the target output and the actual output as shown in equation below:

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

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:

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

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

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.

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

However, in our proposed algorithm, we add chaotic noise 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.

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

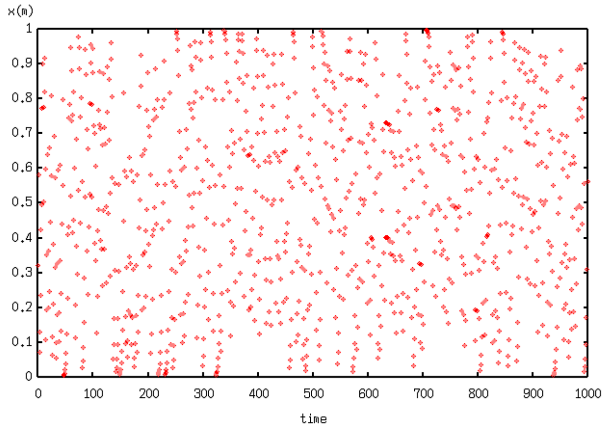
$$noise_{i,j}(m) = \beta_{i,j}(m)(x_{i,j}(m) - 0.5), \tag{6}$$

2.1. Chaotic noise

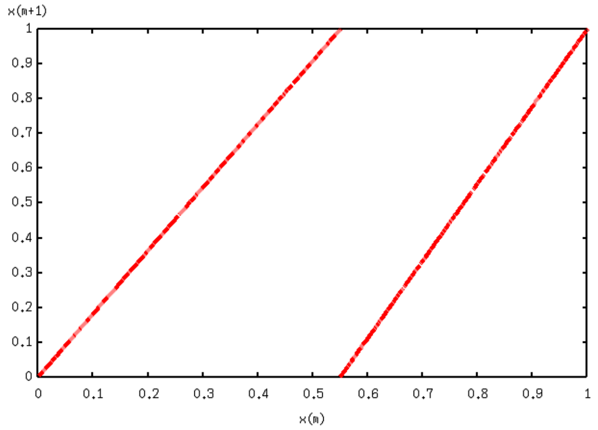
Chaos can be generated by using various chaotic maps. In our previous research, we have introduced logistic map and skew tent map, which are well-known as one-dimensional simple chaotic map to generate chaos. In order to confirm the effectiveness of our proposed algorithm, we introduce other chaotic map as well. Hence, in this study, we use another one-dimensional chaotic map which is Bernoulli shift map to generate chaos.

An example of time series and Bernoulli shift map obtained by equation below are shown in Fig. 1. One of the most important parameters for this work is α, the control parameter of chaos.

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



(a) Time series of Bernoulli shift map.



(b) Bernoulli shift map.

Figure 1: Bernoulli shift map.

3. Simulation results

Here, we consider a pattern recognition task, as one example of difficult problems. The learning example of pattern recognition is 10 patterns of numeric characters $(1, 2, 3, 4, 5, 6, 7, 8, 9, 0)$ and each pattern is composed of 7x5 neurons (see Fig. 2.) This numeric characters are fed into the neural network for recognition. In this case, the number of neurons in the input layer is 35 and the number of neurons in output layer is fixed to 10. We choose 16 neurons in the hidden layer. For recognition, a set of 10 patterns shifted 1 bit from each original pattern was prepared, leading a set of 100 patterns to be recognized.

We carried out the chaotic BP learning algorithm by using the following parameters. The learning rate and the inertia rate are fixed as $\eta = 0.1$ and $\zeta = 0.001$ respectively. The initial values of the weights are given between -1.0 and 1.0 at random. The learning iterations is set to 20000 steps. Here, chaos parameter of α is set to 0.55 and noise amplitude β is fixed as 0.01.

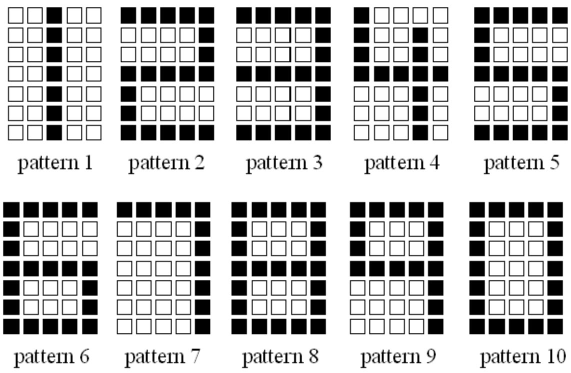


Figure 2: Pattern recognition.

3.1. Noise added into 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. 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). The digram model of our proposed network can be shown in Fig. 3.

We make a comparison between our proposed network and the conventional BP algorithm where there is no noise adding at all. Figure 4 shows the learning performance of all proposed and conventional BP network for pattern recognition problem in term of error rate calculation. We conduct 10 trials by using different initial conditions and we obtain reliable averages of error rate. The final value of error rate at iterations=20000 can be summarized as below.

Table 1: Final value of error rate.

Conventional	Network-1	Network-2	Network-3
0.5362454	0.2259438	0.3859357	0.1982767

From this result, we confirm that our proposed network which chaotic noise is added into the weight update process can give better learning performance than the conventional network for pattern recognition problem. The addition

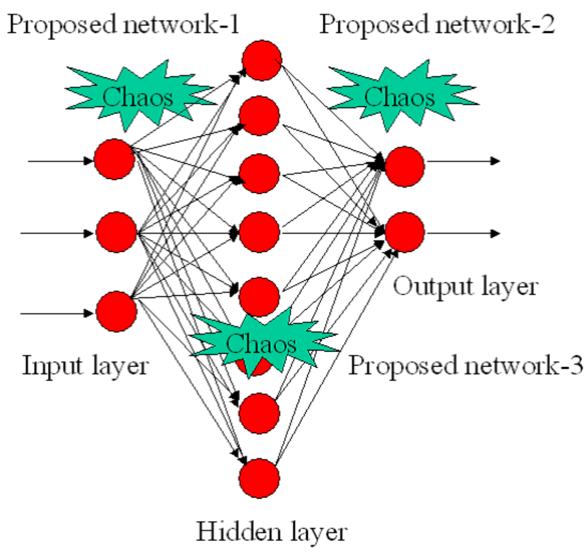


Figure 3: Digram model of all proposed network.

of chaotic noise during weight update process can help the learning process to find a good solution in early time compared with the conventional BP algorithm although the improvement rate is very small.

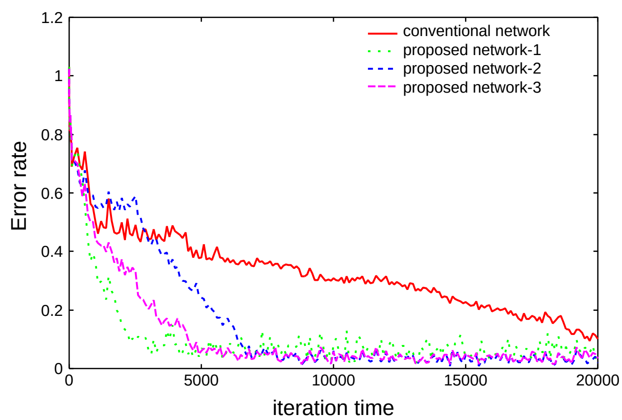


Figure 4: Learning curve of all conventional and proposed networks for pattern recognition.

3.2. Role of hidden neurons

During network training, the number of hidden neurons are crtitical. It is difficult to set an accurate number of hidden neurons. Therefore, in this study we conduct a further investigation on how to optimize the hidden layer neurons.

Figure 5 displays the learning curve of proposed network-3 when the number of hidden neurons are changed. From this figure, we can relate the network learning ability with hidden layer neurons. Upon increasing the number of hidden neurons, the learning ability also increase, showing that the hidden neurons play an important role during learning process. But if the number of hidden neurons are too large, it will lead the learning performance to oscillate. Here, the optimum number of hidden neurons are 16.

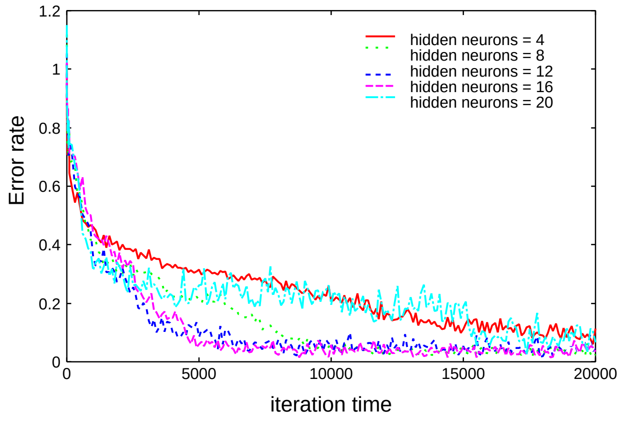


Figure 5: Learning curve of proposed network-3 when the number of hidden neurons are changed.

4. Conclusions

In this study, we have applied our proposed algorithm to learn a pattern recognition which is an example of difficult problems. From simulation results, we confirmed that our proposed network can improve the learning performance compared with the conventional BP algorithm. We confirmed that the addition of chaos noise into weight update process during error propagation can give a better recognition rate and less iteration time. We also discovered that the learning process will take a longer time if more difficult learning examples are used compared with the easy problems.

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 back-propagation 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. 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.

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

[9] 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.