

Research on Hybrid Algorithm Based on BP Neural Network and Genetic Algorithm

Yongxin Wang*

Chongqing College of Electronic Engineering, Chongqing, China

Abstract: BP neural network can approximate nonlinear functions in any degree of accuracy. However, it exhibits the shortcoming of slow convergence, and it is easy to fall into local minimum value. The genetic algorithm has academic advantages both at home and abroad. On the basis of analysis on features of genetic algorithm and BP neural network, the explanations on the necessity of combining genetic algorithm and neural network are provided. In order to improve the structure and setting parameters of neural network, a hybrid algorithm which can optimize BP neural network based on genetic algorithm is proposed. Hybrid algorithm optimizes the weights of BP neural network by an improved genetic algorithm, and makes up the neural network with global random search, which has the capability of genetic algorithm and likely falls into the local optimal solution. Meanwhile, the changes of traditional mechanisms on the same generation crossover in genetic algorithm, and the use of parent and offspring cross to avoid premature loss of evolution ability of genetic algorithm are also discussed in this paper. The algorithm is validated by using standard data sets wine and letter-recognition of UCI database. The results show that the hybrid algorithm proposed in this paper can improve the convergence speed of BP neural network and speed up the training process of this network.

Keywords: BP neural network, genetic algorithms, globally optimal, hybrid algorithm, weights.

1. INTRODUCTION

Neural network is a field that develops with the pace of neurobiology. Artificial Neural Network (Artificial Neural Network, ANN) is a neural network, which achieves some functions that are generated under humans' understanding of brain network. Since the artificial neural network has high degree of parallelism, distributed storage, adaptive learning ability and other characteristics, it has been widely used in pattern recognition, artificial intelligence, forecasting, evaluation, signal processing, nonlinear control and other fields [1-4]. Among these, BP neural network is most widely used. BP neural network has a network of simple structure, stable working conditions and self-learning ability. Nevertheless, it also has some shortcomings such as the slow convergence of learning algorithm. It also easily falls into local minimum value. The genetic algorithm, however can solve these problems precisely. Genetic algorithm can avoid falling into local minimum, and speed up the convergence rate. In genetic manipulation, the classic models are represented by standard genetic algorithm [5], niche genetic algorithm [6], quantum genetic algorithm [7] and hybrid genetic algorithm [8]. They choose the same characteristics of chromosomes from parent groups $P(t)$ through the following process. The first step consists of the selection of operator to constitute mating pool, and then the chromosomes in mating pool generate new groups $P(t+1)$ by crossover operation. Finally, the parent groups $P(t)$ are eliminated by the new groups $P(t+1)$. In other words, in the iteration process of this

classic model, only chromosomes that belong to the same generation can be cross-operated. However, from a genetic point of view, the offspring that has the same generation of chromosomes produced by genetic manipulation may not have genetic advantages. The new generation of cross-operating might be no better than the best individuals in its parent chromosome group. On the contrary, it may cause premature convergence for the algorithm. Therefore, this paper proposes a new crossover operation. Firstly, the best chromosomes are inhibited from the initial population, thus the restricting the best parent chromosomes to cross-operate with the offspring. Then, the best individuals are chosen and restricted from the inhibited parent chromosomes and the new generated generation. Thus, optimal chromosome is ensured to be involved in genetic manipulation. Under these constraints, which can also meet the requirements of selection operator, this operation ensures that the genetic operations are carried out from the best individuals, thus improving the search speed of the algorithm. Experimental results from many cases have shown that these constraints of the same generation crossing cause a rapid decrease of the diversity in groups, and then the genetic algorithm prematurely loses its ability to evolve. But the method of using the best individuals for parent participation in genetic manipulation avoids the problems which manifest in traditional algorithms.

2. BP NEURAL NETWORK AND ITS LIMITATIONS

2.1. BP Neural Network Model

BP neural network was proposed by a group of scientists who were led by Rumelhart and McClelland in 1986. It is

*Address correspondence to these authors at Chongqing College of Electronic Engineering, Chongqing, China; Tel: +8615803002011; E-mail: 82346654@qq.com

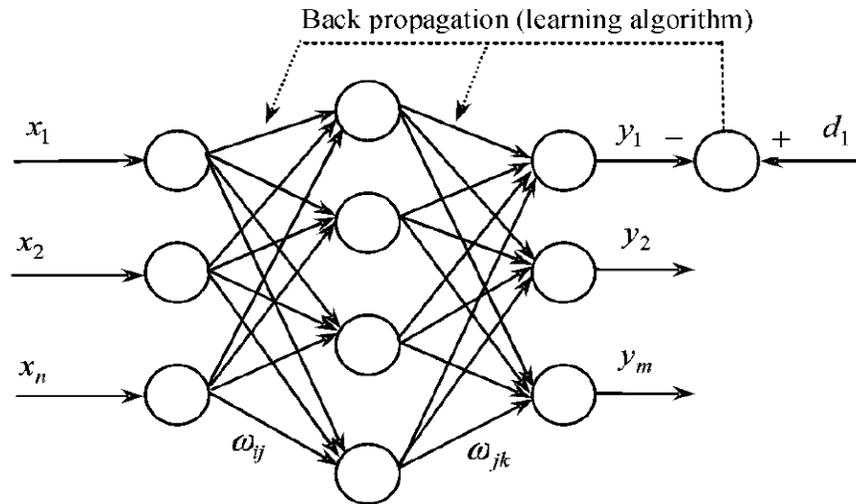


Fig. (1). The topology of three-layer BP network.

constituted by an input layer, some hidden layers, and an output layer. A three-layer BP network can convert any N-dimensional mapping [9] to M-dimensional. Theoretically, BP neural network can simulate any set of data as long as adequate training is given. As BP's English name is Back Propagation of a BP neural network algorithm can also be called as error back-propagation algorithm. Specific learning of this algorithm is composed of two processes [10]: the forward propagation process and the back-propagation process. The forward propagation process is the learning process of passing the sample's input vector from the input layer to the hidden layer and then to the output layer. It then reverses the spread if the output error with expected value is obvious. This involves making the output error propagate through the hidden layer to move into the input layer, and successively modify the weights. These two processes repeat alternately until convergence is reached. Fig. (1) represents a typical three-layer BP network model. In this model, the first layer is the input layer; the second layer is the hidden layer; and the third layer is the output layer.

2.2. The Limitations of BP Neural Network

Although the application of BP network is extremely extensive, it still has following shortcomings [11-12]:

- 1) Learning rate is fixed, and results in slow network convergence and difficult grasp of training.
- 2) Even though BP algorithm makes the connection weights converge, it may only get local extreme value. Since the squared error function easily falls into the local minimum.
- 3) Currently, there is no theoretical basis for the number of hidden layers, the nodes of each hidden layer in the network and the settings of learning parameters. These data are usually obtained through experience or experiments. Although some scholars have summarized some empirical formulas which have a supporting role, the practical applications of these results are unsatisfactory.

- 4) The learning and memory of network are unstable. The entire training set should be submitted by BP algorithm to the network at once. Then only it can join the weight's adjustments. The previously trained network needs to start learning again if the training sample is modified.

3. IMPROVED GENETIC ALGORITHM

So far, there is no evidence to prove that the same genetic generation is better than atavistic. Moreover, the same genetic generation easily reduces the diversity of populations, making the entire algorithm prematurely lose the ability to evolve. Therefore, this paper proposes an improved genetic algorithm using atavistic approach. It proceeds as follows. First, the best chromosome T_{\max} is chosen in the initial chromosome pool T , and then T_{\max} is crossed with the chromosome in the initial chromosome pool to generate new populations. All chromosomes in the new populations are sorted out, and the best chromosome T'_{\max} is selected in order to be compared with T_{\max} . T_{\max} is then replaced with T'_{\max} if T'_{\max} is superior to T_{\max} . Simultaneously, the better parts are chosen in the new populations to generate a new chromosome pool T' , then original chromosome pool T is replaced with T' . In this iterative process, mutation is put on the part of chromosome. This iterative process is completed until the end condition is satisfied.

3.1. Chromosome Coding

Genetic algorithm uses encoding methods to replace the parameters in this problem. One corresponding relationship between the representations of all the practical problems with coded bits must be constituted. It also requires multiple encoding and decoding manipulations during operations. For the weight optimization of BP neural network, this paper encodes each layer of the network weights and thresholds in a real-coded way. The multidimensional matrix W_1, B, W_2, S which represented BP network weights and thresholds were represented in the form of one-dimensional matrix. Each

gene of the chromosome could represent a weight or threshold of the network. The encoded form of three-layer network institutions shown in Fig. (1) is as follows:

$$w_{111}, w_{112}, \dots, w_{1m}, b_1, b_2, \dots, b_l, w_{211}, w_{212}, \dots, w_{2lm}, s_1, s_2, \dots, s_m \quad (1)$$

3.2. Generating Initial Population and Local Optimum Chromosome

Initial weights of BP neural network take value from the random decimal in a uniform distribution (-1,1). The length of chromosome is $L = W_1 + B + W_2 + S + 1$. The last bit of each chromosome is the fitness for this chromosome. This bit does not involve in crossover and mutation operations in genetic operation. Herein, using the function *Initalizega* of initial population to generate an initial pool of chromosome and local optimal chromosome T_{max} , specific functions are as follows:

$$[T, T_{max}] = \text{Initalizega}(P_{size}, aa, \text{gabpEbal}) \quad (2)$$

In the above formula, T is the initial chromosome group, T_{max} is the optimal chromosome of initial chromosome group, P_{size} is the size of initial population, aa is the value range of each chromosome gene, and *gabpEbal* is the fitness function.

3.3 Selecting Operators

The optimized results are passed to the system-level by every sub-discipline. Under the condition of consistent inequality constraints, system-level is used to apply intelligently optimized algorithm to maximize (small) the target. The selection of genetic algorithm refers to searching individuals with higher fitness in the whole population, in order to generate the initial mating pool. The most common methods of selection are adapting proportional selection, best retain selection, sorting selection and expectations selection. This paper adopts the adapting proportional selection method, and chooses the proportion between the fitness of individuals and the average fitness of the entire population. First the fitness of each individual is calculated in the population; then the ratio is assessed of this fitness account with regard to the total population's fitness. If there are larger proportions of the individual accounts, greater probability is selected; if not, then the smaller probability is selected instead.

For the group with given size $n P = \{a_1, a_2, \dots, a_n\}$, the fitness of individual $a_j \in P$ is $f(a_j)$, with the selected probability being

$$p_s(a_j) = \frac{f(a_j)}{\sum_{i=1}^n f(a_i)} \quad (j = 1, 2, \dots, n) \quad (3)$$

3.4 Fitness Function

Genetic algorithm simulates biological *survival of the fittest* mechanism by the fitness function. In the genetic algorithm, the first steps are calculation of the individual's fitness through fitness function, and the reflection of different indi-

vidual degrees in the population through the size of individual's fitness. Then, whether the individual has the right to conduct the genetic manipulation of new generation is determined. The fitness function the performance evaluation function for the BP neural network in this article--of genetic algorithm is the only standard to guide the search of genetic algorithm. Its selection is the key for the quality of algorithm. The fitness function needs to effectively guide the search through the optimal parameters combination and gradually approaches the optimal solution; it also needs to ensure that the search can converge and fall into local optimum. For BP neural network, the smaller the training error, the more gifted is the chromosome. Therefore, this article takes all individuals and the inverse of sum on error squares from all corresponding target samples as the fitness of chromosomes. Specific functions are as follows:

$$[\text{fitness}] = \text{gabpEbal}(T) \quad (4)$$

Here, the fitness is the fitness of chromosome. Each chromosome is decoded when calculating the output value of the neural network, then the weight is calculated for W_1, W_2 and thresholding B, S of each corresponding neuron in neural network. The matrix Y could be calculated which represents the output values of corresponding network for each sample according to formula (4).

$$Y = f_2(W_2 \cdot f_1(W_1 \cdot X, B), S) \quad (5)$$

Here, the three-tier structure neural network is taken as an example; $f_1(), f_2()$ represent the transfer functions of the neural network. According to the formula

$$E = \sum_{k=1}^N (T_k - Y_k)^2$$

is the sum on error squares from samples of each group, where N represents the number of samples to be calculated. Finally, the fitness of chromosomes is calculated as $\text{fitness} = 1 / E$.

3.5 Crossing Operators

The purpose of the genetic algorithm in the operation of crossover is to generate new individuals in the next generation. The searching capabilities of genetic algorithm improve drastically by operations of crossing and restructuring. Crossing and restructuring chromosome in genetic algorithm are main methods to obtain new superior individuals. At the same time, operation of crossing has important impact on the quality of genetic algorithm. Therefore, this article selects the optimal solution T_{max} from the parent pool to cross with the chromosome from the offspring pool. Thus, the algorithm would not prematurely lose the ability to evolve. Commonly used methods of crossover are one-point cross, two-point crossover, multiple-point crossover and uniform crossover. This paper uses the method of multiple-bits intersection in a single gene. Specific algorithm proceeds as follows:

$$\text{For } i = 1 : P_{size}$$

$a = \text{floor}(\text{rand}() * n)$; %n represents the length of chromosome

```

T_max1 = T_max ;
T_j = T(i,:);
for k=1:3
    e = T_max1(a + 2 * k);
    T_max1(a + 2 * k) = T_j(a + 2 * k + 1);
    T_j(a + 2 * k + 1) = e;
end
T_1(2 * i - 1,:) = T_max1; %T1 is the total group of new chromosomes after crossover
T_1(2 * i,:) = T_j;
end
    
```

3.6 Mutation Operators

The mutation rate in the calculation of variation usually takes $p_m = 0.1$, randomly selects the k -th gene t_{ik} in a single chromosome $T_i(t_{i1}, t_{i2}, \dots, t_{iL-1})$, and then adds a small perturbation on t_{ik} to achieve the mutation of chromosome. Since the best individuals T_{max} retention policy is used, researches can increase the probability of search in the vicinity of current best individual to get the optimal solution in the operation of mutation, and there is no need to worry about the damaging of the fine chromosome that already exists. So, taking of a good part chromosome of the population can offer variability.

4. THE PRECONDITIONING OF DATA AND ALGORITHMIC FLOW

4.1 The Normalized Processing of Data

When data is calculated, the attributes of the sample are diverse, and there are great differences in magnitude between some attributes. The imbalance of these data in the training process of network causes some characteristics of attributes to not fully reflect, and may even lead to non-convergence of network. Therefore, before network training and recognition, appropriate preconditioning of raw data is needed to make it more suitable for training neural network. Given here is the method of normalizing data. This method transforms the input data of training samples and the target data, so that data is distributed in the range of [-1,1]. Normalized function uses the function premnmx Specific formula:

$$P_n = 2 \times \frac{P - \min p}{\max p - \min p} - 1 \tag{6}$$

$$T_n = 2 \times \frac{T - \min t}{\max t - \min t} - 1 \tag{7}$$

Where, P is the raw inputted data, $\max p$ and $\min p$ are the maximum and minimum of P , P_n is the input data

after normalized process. T is the original target data, $\max t$ and $\min t$ represent the maximum and the minimum of the target data T , and T_n is the target data after normalized process.

4.2 The Basic Processes and the Basic Framework of Algorithm

The basic process of improving the genetic algorithm and optimizing the weights of neural network is shown in Fig. (2).

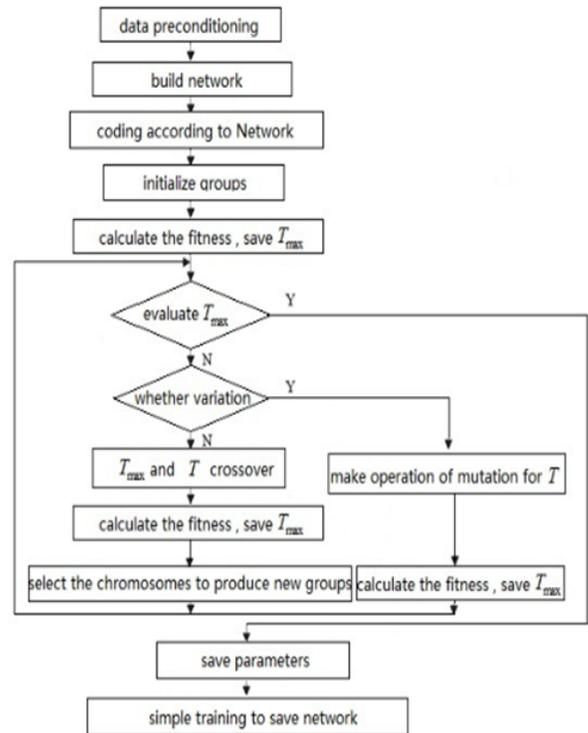


Fig. (2). The process of hybrid algorithm.

5. SIMULATION AND ANALYSIS

To test the performance of the algorithm, this paper uses standard data sets wine and letter-recognition of the UCI [13] database. These methods can verify the algorithm, and compare the value of fitness function of traditional algorithm with the value of the improved algorithm under the same number of iteration (experimental environment: CPU is Celeron1.6, memory: 1GB, operating system: Windows XP, programming languages: Matlab2009).

Experiment 1 uses the data of wine. The network converged faster in training since the amount of data is small. Fig. (3) and Fig. (4) respectively, show the curves of fitness of traditional algorithm and improved algorithm in 200 times iterations. Judging from changes in fitness, the improved algorithm is much better than the traditional algorithm. In the same 200 times iterations, the fitness of improved algorithm reaches about 0.7, while the traditional algorithm reaches only about 0.2. Fig. (5) and Fig. (6) present the curves of the number of network's convergence for the two algorithms. Under the same condition, the number of network training of the improved algorithm is significantly lower than the traditional algorithm.

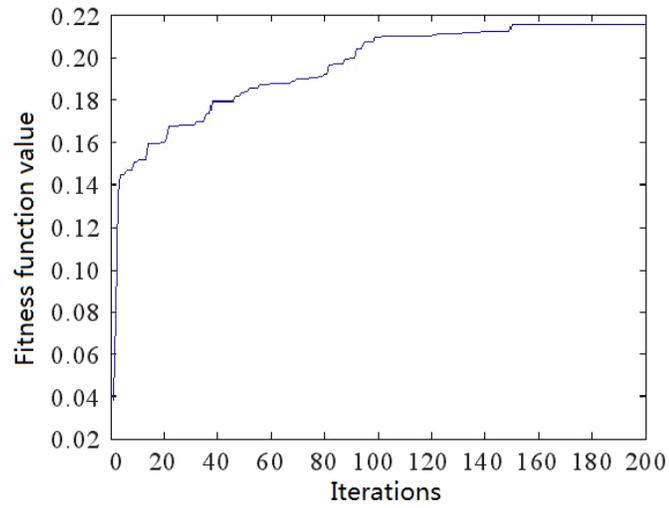


Fig. (3). The curve of fitness of traditional algorithm (wine).

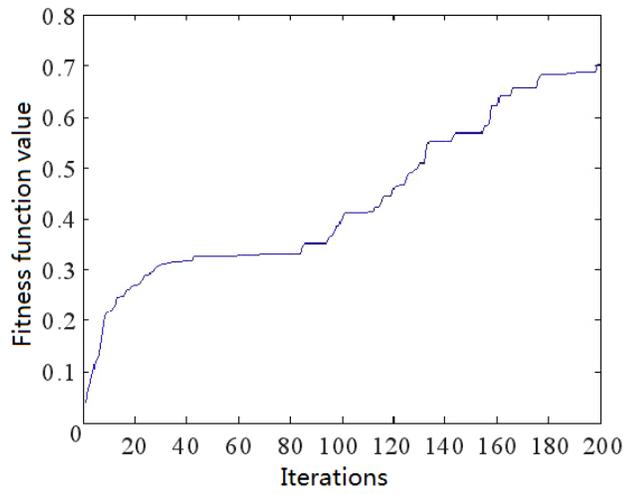


Fig. (4). The curve of fitness of improved algorithm (wine).

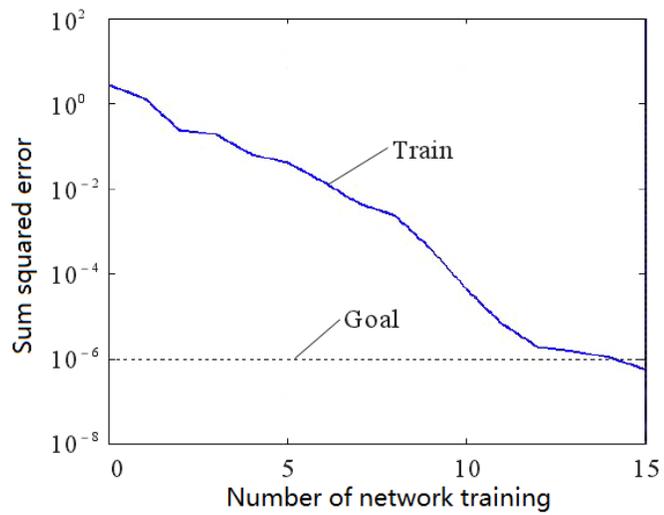


Fig. (5). The network training of traditional algorithm (wine).

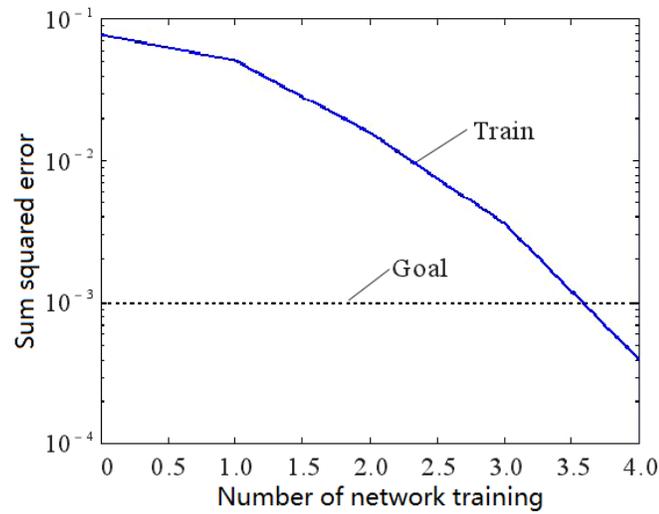


Fig. (6). The network training of improved algorithm (wine).

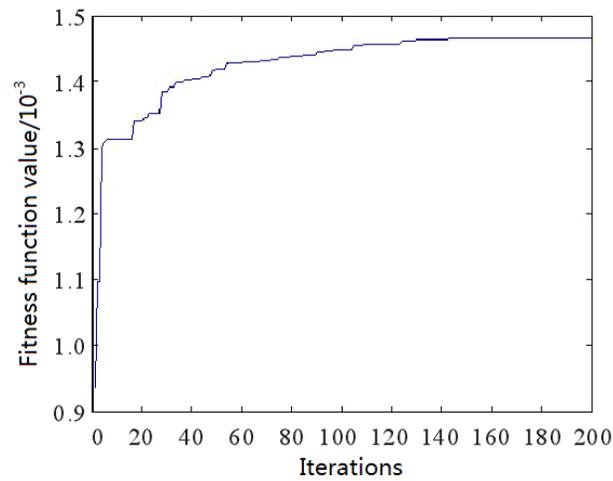


Fig. (7). The curve of fitness of traditional algorithm (letter-recognition).

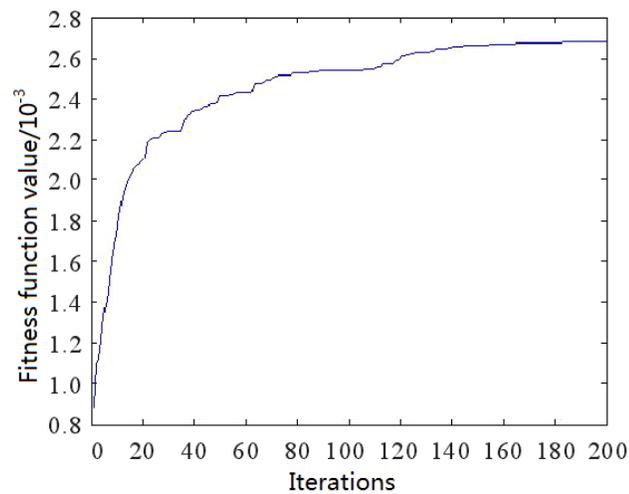


Fig. (8). The curve of fitness of improved algorithm (letter-recognition).

Experiment 2 uses the data of letter-recognition and selects 200 samples of each type from 26 classes of data to train. Figs. (7 and 8) represent the curves of fitness of tradi-

tional algorithm and improved algorithm in 200 times iterations. The effects of network training of the traditional algorithm and improved algorithm are shown in Figs. (9 and 10).

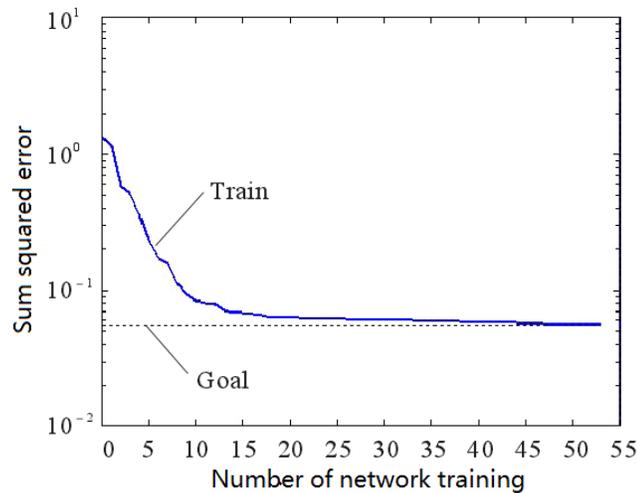


Fig. (9). The network training of traditional algorithm (letter-recognition).

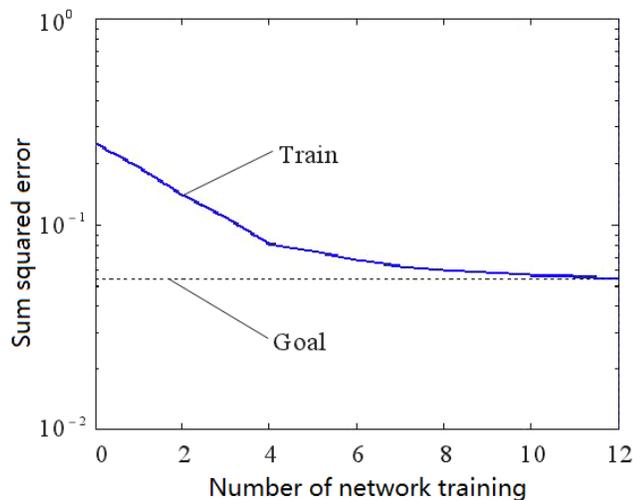


Fig. (10). The network training of improved algorithm (letter-recognition).

CONCLUSION

The genetic algorithm is widely used in optimizing, and the usage of genetic algorithm is very common in the field of optimization of the weights of neural network. However, most of the genetic algorithms cannot avoid the problem of premature convergence caused by same generation cross. The improved genetic algorithm proposed in this paper is helpful in avoiding this problem. Combining the advantage of global search of algorithm with the advantage of convergence speed of BP neural network will improve the convergence speed of BP neural network and accelerate the training process of the network. It provides a better way to solve some practical problems such as classification and analysis of principal component. The experimental results show that the effect of algorithm on the data from large size sample is more obvious. This method also shows its advantages in handling continuous data. However, some problems in dealing with discrete data still remain and further improvements are needed to be carried out.

CONFLICT OF INTEREST

The author confirms that this article content has no conflict of interest.

ACKNOWLEDGEMENTS

Declared none.

REFERENCES

- [1] B. Pradhan, and S. Lee, "Delineation of landslide hazard areas on Penang Island, Malaysia, by using frequency ratio, logistic regression, and artificial neural network models," *Environmental Earth Sciences*, vol. 60, no. 5, pp. 1037-1054, 2010.
- [2] Y. Shen, and A. Bax, "SPARTA+: a modest improvement in empirical NMR chemical shift prediction by means of an artificial neural network," *Journal of biomolecular NMR*, vol. 48, no. 1, pp. 13-22, 2010.
- [3] S. Goyal, and G. K. Goyal, "Cascade and feedforward backpropagation artificial neural networks models for prediction of sensory quality of instant coffee flavoured sterilized drink," *Canadian Journal on Artificial Intelligence, Machine Learning and Pattern Recognition*, vol. 2, no.6, pp. 78-82, 2011.

- [4] R. Taormina, K. Chau, and R. Sathi, "Artificial neural network simulation of hourly groundwater levels in a coastal aquifer system of the Venice lagoon," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 8, pp. 1670-1676, 2012.
- [5] C. R. Reeves, "A genetic algorithm for flowshop sequencing," *Computers & operations research*, vol. 22, no.1, pp. 5-13, 1995.
- [6] M. Jelasity, J. Dombi, "GAS, a concept on modeling species in genetic algorithms," *Artificial Intelligence*, vol. 99, no. 1, pp. 1-19, 1998.
- [7] L. Jiao, Y. Li, M. Gong, and X. Zhang, "Quantum-inspired immune clonal algorithm for global optimization," *Systems, Man, and Cybernetics, Part B: Cybernetics, IEEE Transactions on*, vol. 38, no. 5, pp. 1234-1253, 2008.
- [8] C. F. Tsai, C. W. Tsai, and C. C. Tseng, "A new hybrid heuristic approach for solving large traveling salesman problem," *Information Sciences*, vol. 166, no.1, pp. 67-81, 2004.
- [9] N. Mohsenifar, N. Mohsenifar, K. Mohsenifar, "Using Artificial Neural Network (ANN) for Estimating Rainfall Relationship with River Pollution," *Advances in Environmental Biology*, vol. 5, no. 6, pp. 1202-1208, 2011.
- [10] G. Li, J. Shi, "On comparing three artificial neural networks for wind speed forecasting," *Applied Energy*, vol. 87, no.7, pp. 2313-2320, 2010.
- [11] B. Pradhan, and S. Lee, "Landslide susceptibility assessment and factor effect analysis: backpropagation artificial neural networks and their comparison with frequency ratio and bivariate logistic regression modeling," *Environmental Modelling & Software*, vol. 25, no. 6, pp. 747-759, 2010.
- [12] V. K. Gupta, H. Khani, B. Ahmadi-Roudi, S. Mirakhorli, E. Fereyduni, S. Agarwal. "Prediction of capillary gas chromatographic retention times of fatty acid methyl esters in human blood using MLR, PLS and back-propagation artificial neural networks," *Talanta*, vol. 83, no.3, pp. 1014-1022, 2011.
- [13] C. J. Merz and P. M. Murphy, UCI repository of machine learning databases [EB/OL]. [2012-03-01]. <http://www.ics.uci.edu/~mllearn/MLRepository.html>.

Received: May 26, 2015

Revised: July 14, 2015

Accepted: August 10, 2015

© Yongxin Wang; Licensee *Bentham Open*.

This is an open access articles licensed under the terms of the Creative Commons Attribution-Non-Commercial 4.0 International Public License (CC BY-NC 4.0) (<https://creativecommons.org/licenses/by-nc/4.0/legalcode>), which permits unrestricted, non-commercial use, distribution and reproduction in any medium, provided that the work is properly cited.