A New Neural Network with Genetic Algorithm in Searching Nonlinear Function Extremum

Huiyan Qu1,*, Sining Cheng2

1School of Information Technology, Jilin Agricultural University, Changchun, Jilin Province, 130118, China
2Haikou College of Economics, Haikou, Hainan Province, 571127, China

*Corresponding author e-mail: quhuiyan68@163.com

Abstract. In order to more accurate for nonlinear function extreme, this paper used improved particle swarm optimization neural network combining with genetic algorithm method to solve the problem. In view of the particle swarm optimization algorithm is easy to appear "premature" faults, introducing the adaptive threshold, initializing particles if they were under the constraint conditions, making particles jump out to the optimal value of the position in previous search. Through the experiment, contrasts to the genetic neural network algorithm and traditional BP neural network, this method is faster in convergence and has the smallest prediction error. Finally, combining with genetic algorithm, calculating the extreme value of nonlinear function by using the above three kinds of neural network trained forecast as an individual output fitness value. The adaptive particle swarm optimization neural network proves the most close to the theoretical calculation. It shows that the method is effective.

1. Introduction

Optimization technique is a modern design method and technology which is formed by the theory of optimum and the combining of computer technology and engineering technology. The application will faster to realize the optimal design, improve the design efficiency and the design quality. In recent years, particle swarm optimization (PSO) [1], artificial neural network (ANN) [2] and genetic algorithm (GA) [3], and other modern optimization method has been widely development and application. But PSO is easy to fall into local optimum, the most extensive network model-back propagation neural network (BPNN) has slow convergence speed and easy to fall into the local extreme value point extensive network model, GA sometimes is random to optimal operation, and it is difficult to get the optimal solution [4-6].

According to the above three kinds of the characteristics of the algorithm, this paper proposes an adaptive particle swarm optimization neural network combining with genetic algorithm optimization method, in view of the particle swarm optimization algorithm is easy to appear "premature" faults, the introduction of the adaptive threshold, for greater than adaptive threshold value particles, for more than adaptive threshold value particles, updating of the operation, that is for some variables to a certain probability to initialize. The update operation expanded in the iteration of the dwindling populations search space, make the particle can jump out to the previous search optimal value of the position, and in the larger space launch, and maintain search population diversity, improve the
algorithm to find the possibility of more optimal value. Then using the adaptive particle swarm optimization to optimize the BP neural network, and trained the BP neural network model output as individual fitness value. Finally combining with the nonlinear optimization ability of genetic algorithm, find the function extreme.

2. Adaptive particle swarm optimization neural network genetic algorithm

2.1 Basic PSO algorithm

PSO produces the initial population, that is, random initial a group of particles in the feasible solution space, every particle for a feasible solution of optimization problems, and the objective function is determined to a fitness value [7]. According to follow the principle of optimal particle, particle $x_i$ will press (1), (2) type change speed and position.

$$v_i^{(t+1)} = v_i^{(t)} + \frac{c_1 r_1 (p_i^{(t)} - x_i^{(t)}) + c_2 r_2 (p_g^{(t)} - x_i^{(t)})}{m_d},$$

$$x_i^{(t+1)} = x_i^{(t)} + v_i^{(t+1)}$$

where $d = 1,2,\ldots,n$, $i = 1,2,\ldots,m$, $m$ represents for the population scale, $t$ is the current evolution algebra, $r_1$ and $r_2$ are distributed in the random number in $[0,1]$, $c_1$ and $c_2$ are accelerate constant. In addition, in order to make the particle velocity from too much, setting the speed limit $V_{max}$, that is when the (1) type $v_i < V_{max}$, $v_i = V_{max}$, when $v_i < -V_{max}$, $v_i = -V_{max}$.

2.2 Back-propagation neural network

BP network error adjusting direction is always proceeding along the fastest decline of the error, the conventional method of three BP network weights and threshold adjustment formula is as follows:

$$w_{ij}^{(t+1)} = -\eta \frac{\partial E}{\partial w_{ij}} + w_{ij}(t)$$

$$w_{jk}^{(t+1)} = -\eta \frac{\partial E}{\partial w_{jk}} + w_{jk}(t)$$

$$B_{jk}^{(t+1)} = -\eta \frac{\partial E}{\partial B_{jk}} + B_{jk}(t)$$

where $E$ is the error sum of squares between network output and actual output samples; $\eta$ is the network learning rate, or weights adjustment range; $w_{ij}(t)$ is for the $t$ time connection weights between input layer $i$ neurons and hidden layer $j$ neurons; $w_{ij}(t+1)$ is for the $t+1$ time connection weights between input layer $i$ neurons and hidden layer $j$ neurons; $w_{jk}(t)$ is for the $t$ time connection weights between hidden layer $j$ neurons and output layer $k$ neurons; $w_{jk}(t+1)$ is for the $t+1$ time connection weights between hidden layer $j$ neurons and output layer $k$ neurons; $B$ is neurons threshold value, its subscript meaning is the same as weights.

According to the specific neurons excitation function expression, specific to find out the analytical formula about $\frac{\partial E}{\partial w_{ij}}$ and $\frac{\partial E}{\partial w_{jk}}$. Network through the negative gradient descent learning rule to fixed weights and threshold enhances the accuracy is getting smaller and eventually to achieve the ideal error.
2.3 Adaptive particle swarm optimization optimizes neural network

PSO training BP network, defining the position of the particles is all the threshold of the weight and node in all the connection structure, its weight range as \((-1,1)\). Fitness function chooses the reciprocal of error sum squares in the test data set, namely:

\[
f(X) = \frac{1}{\sum_{i=1}^{n} (\hat{t}_i - t_i)^2}
\]

where \(\hat{t}_i\) is the predictive value of the test data set, \(t_i\) is the true value of the test data set, \(n\) is the Sample number of the test data set. Basic PSO algorithm is easy to have "premature" faults in optimal process. In order to solve this problem, this paper introduces into adaptive threshold during the iteration of the PSO algorithm. Using update operation when some particles are greater than adaptive threshold value, that is, to initialize some variables by a certain probability. The update operation expands the dwindling populations search space during the iteration process, makes the particle jump out to the optimal value position by the previous search, and makes it is available to search in a larger space, at the same time it keeps the diversity of population, increases the possibility of the algorithm finding more optimal value.

(1) Randomly generated \(N\) group of neural network structure parameters and the initial velocity of particles;
(2) For this \(N\) group to particle PSO iteration, calculating the global optimal particle position gbest and the single optimum particle position pbest
(3) According to the type of 1, 2 update each particle type speed and position in space;
(4) Set adaptive threshold \(\lambda\), by computer program, producing a between 0 ~ 1 uniform distribution of random number \(\text{rand}(0,1)\), if \(\text{rand}(0,1) > \lambda\), then turn to 5, or jump to 6;
(5) Operation initialization to all particles which is above adaptive threshold value, updating the global optimal particle position gbest a particle and the optimum position pbest;
(6) Calculating whether the global optimal particle position gbest fitness value and the largest number of default training fits for the default conditions. If yes, then gbest is needed for results; Otherwise, jump to (2).

2.4 Genetic algorithm optimizes the neural network

In order to contrast and test to the adaptive PSO neural network algorithm, this paper also uses genetic algorithm optimizes the neural network. The algorithm is described below:

Genetic algorithm (GA) is put forward by the American scholars J.H.Holland. It is highly parallel, random, and adaptive optimization algorithm which simulates natural selection and natural genetic mechanism in the biosphere. The algorithm just need to search a minority of structure can reflect large area in the searching space. Through the simple choice, crossover and mutation operators, it uses of fitness value information of the group to find the global optimal solution by a large probability. Especially the genetic algorithm is suitable for solve the complicated nonlinear programming problems which can’t be solved by traditional search method. It has developed into a self-organizing, adaptive, comprehensive technology. Among them, the purpose of the choice is to choose the excellent individual from the current group, makes them have a chance to multiply to next generation children as father. The basis of selection is the individual with strong adaptability having a large probability to contribute for the next generation. We can get a new generation of individuals through the crossover operation. The new individual has the character with its fathers. Each of the individual should match into pairs randomly in the group. For each individual exchanges their part of the chromosome with crossover probability.

The basic idea of GA optimizes the neural network is introducing GA into BP neural network. It uses global search characteristics of GA to train the network weights. Then using BP algorithm to calculate the exact solution, get the purpose of the global optimization, quickly and accurately. It is
mainly consisted by three parts: ensuring BP neural network results, GA optimizing and BP neural network forecast. Among them, in the part of ensuring BP neural network, making sure the BP neural network structure according to the input/output parameters number of fitting function, then determines the length of GA individual. In the process of optimizing, each individual in the group contains all the weights and thresholds in one network. They search the optimal value through selection, crossover and mutation operators. BP neutral network gets initialized weights and threshold value assignment of the optimal value according to GA. The network outputs the predictive function after training. The basic procedure of the algorithm is as follows:

1. Setting up the network topology structure and calculation parameters in BP, and initialize weights and threshold in the group.
2. Getting the errors by training BP neural network as the fitness value.
3. Using GA to optimize the algorithm through selection, crossover and mutation operators.
4. If meets stopping rules, it will be treated as a possible solution of the neural network learning. Get the optimal weights and threshold values. Turn to (5), otherwise, turn to (3).
5. Calculating the errors of network, updating the weights and threshold value of network.
6. If it meets the convergence conditions, completing the training times, output, otherwise, turning to (5).

2.5 Combining with genetic algorithm seeking the function extreme

Genetic algorithm is a kind of calculation model simulating biological evolution process on nature. Only need to determine the objective function on the search direction and the corresponding fitness function, it stresses the probability conversion rule, not ensuring the rules of conversion. Owing to we don't know the scope of variables of some problems before the test, in the application of GA, it need to randomly optimization operation, and it is difficult to get the optimal solution. This paper combines adaptive PSO neural network and genetic algorithm, using the adaptive PSO neural network training results as the genetic algorithm individual fitness value. It improved the faults of random optimizing in GA, and then getting the optimization extreme of nonlinear function. The process of the combination algorithm is as follows:

1. Initializing control parameters of GA. Just as the size of group $N$, mutation probability $P_m$, crossover probability $P_c$.
2. Random generating initial solution group $P(t) = \{P_0, P_1, \cdots, P_n\}$. The number of the individual is fixed, and each individual represents the gene code of chromosomes.
3. Training the data in adaptive PSO neural network. Use the adaptive PSO neural network training results as the genetic algorithm individual fitness value.
4. Selecting regeneration individuals to copy according to the individual fitness. Individual with large adaptive function value has large probability to copy.
5. Taking crossover operation in the current solution group according to a certain crossover probability and crossover method, and generating new individuals.
6. Taking mutation operation when the individual crossed according to a certain mutation probability and mutation method, and generating new individuals.
7. Generating a new group after crossover and mutation operation. If it meets the convergence condition, the algorithm terminated. Otherwise, turning to (3).

In the comparison experiments, we also respectively use GA neural network combining with GA and traditional BP neural network combining with GA to calculate the optimal value of nonlinear function. Just exchange the algorithm in procedure (3).
3. Algorithm achievements

3.1 Neural network optimization

This experiment for the following nonlinear function extremum [9], it expressions for:

\[ y = x_1^2 + x_2^2 \]

Be sure to BP neural network structure for three layers: 2 nodes in the input layer, 5 nodes in the hidden layer, 1 node in the output layer. In this paper, respectively using the particle swarm optimization neural network and genetic algorithm optimizes the BP neural network, at the same time use of traditional BP neural network optimize and train data. In the Adaptive particle swarm optimization neural network process, population size is 10, evolution number is 10, speed update parameters \( c_1 = c_2 = 1.49445 \), maximum and minimum values of individual are 5, -5, maximum and minimum value of speed are 1, -1; In the genetic algorithm optimization process, population size is 10, evolution number is 10, crossover probability is 0.3, the variation probability for 0.1; In the traditional BP neural network algorithm process, maximum iterating times for 100, learning rate is 0.1, the mean square error of 0.00001. Respectively getting adaptive particle swarm optimization algorithm, genetic algorithm optimizes the BP neural network of fitness comparison figure 1, three algorithm prediction error comparisons in figure 2.

![Algorithm fitness comparison](image1)

![Prediction error comparison](image2)

From the above results can be seen, based on the adaptive particle swarm optimization algorithm to optimize BP neural network getting the smallest prediction error, adaptive PSO algorithm to optimize the BP neural network prediction accuracy effect is better than GA algorithm to optimize the BP neural network prediction accuracy.

3.2 Adaptive PSO-BP neural network combines with genetic algorithm seeking the function extremum

Using the Adaptive PSO-BP neural network fitting data, the predictive output and expected output contrast as shown in figure 3 shows. From the adaptive particle swarm neural network prediction results can be seen, adaptive particle group of neural network can accurately predict the nonlinear function, can put the network forecast output as real output function approximation.

Adaptive particle swarm neural network training ended, using the adaptive PSO neural network training results as the genetic algorithm individual fitness value, combining with genetic algorithm looking for the extremum of the nonlinear function. The minimum iteration number is 100, the population scale is 20, crossover probability is 0.4, the variation probability is 0.2, using the floating-point coding, and individual length is 2. we use adaptive PSO neural network, GA neural network and traditional BP neural network combining with GA to calculate the optimal value of nonlinear function. In the process of optimizing the optimal individual fitness value curve is shown in figure 4.
4. Conclusions

This paper proposes an adaptive particle swarm optimization neural network combining with genetic algorithm optimization method, in view of the particle swarm optimization algorithm is easy to appear "premature" faults, the introduction of the adaptive threshold, for greater than adaptive threshold value particles, updating of the operation, making particles can jump out to the previous search optimal value of the position and avoiding the local optimum. Through the experiment, the adaptive particle swarm neural network algorithm contrasts to the genetic neural network algorithm and traditional BP neural network. It is faster in convergence and has the smallest prediction error.

5. Acknowledgments

This work was financially supported by Key Project of "The thirteenth-Five" science and technology research of Jilin Provincial Department of Educationon Optical Motion Capture Technology "Rapid image generation and detection technology based on virtual reality",JJKH20190907KJ and supported by Provin-cial Natural Science Foundation of China, "large-scale complex scenarios based on virtual reality is rapid real-time collision detection technology research" Item number: 619QN247

References
[1] Kou Xiao-Li, Liu San-yang. Particle swarm algorithm based on simulated annealing to solve constrained optimization. Journal of Jilin University(Engineering and Technology Edition), 2007,37(1), 136–140.
[2] Tang Xian-lun, Zhuang Ling, Li Yin-guo ect. Optimizing structure and parameters of feedforward neural networks using hybrid particle swarm optimization. Application Research of Computers, 2017, 24(12), 91–93.
[3] Teschner, M., Kimmerle, S., Heidelberger, B., Zachmann, G., Raghupathi, L., Fuhrmann, A., Cani, M., Faure, F., Magenat-Thalmann, N., Strasser, W., et al. (2015). Collision detection for deformable objects. In Computer Graphics Forum, volume 24, pages 61–81. Wiley Online Library.
[4] Jiang ZY, Cai ZX, Wang Y. Hybrid self-adaptive orthogonal genetic algorithm for solving global optimization problems. Journal of Software, 2010,21(6), 1296–1307.
[5] Zhang Chang-sheng, Sun Ji-gui, Ou Yang Dan-Tong ect. A Self-adaptive hybrid particle swarm optimization algorithm for flow shop scheduling problem. Chinese Journal of Computers, 2009, 32(11), 2137–2145.
[6] Yu Wan-xia, Zhang Wei-cun, Zhang Hong-xing. Genetic and particle swarm algorithm-based optimization solution for high-dimension complex functions. Computer Engineering and Applications, 2007, 43(36), 31–33.
[7] Liu Xiaohua, Lin Jie, Deng Ke. Scheduling optimization in re-entrant lines based on a GA and PSO hybrid algorithm. Journal of TongJi University(Natural Science), 2011,39(5), 726–729.
[8] Li Jian-Yong. The study on particle swarm optimization algorithm, Chemical and Materials Institute of Zhejiang University, March 2014, 5-7
[9] Wang Xue-fei. A Non-linear time-varying system simulation modeling method based on neural networks. Journal of Computer Research and Development, 2006, 43(7), 1167–1172.