Improvements of ant colony algorithm and its applications in artificial neural network

Kuangfeng Ning*, Xiliang Zeng

Hunan University of International Economics, Changsha 410205, Hunan, China

Received 1 June 2014, www.cmnt.lv

Abstract

Ant colony algorithm (ACA) is a bionic intelligent optimization algorithm with positive feedback, distributed computing and heuristic search. As an important branch of computational intelligence and swarm intelligence, ACA has been successfully applied in solving many combinatorial optimization problems. Artificial neural network is a large-scale distributed parallel processing system with the characteristics of self-organization, self-study, self-adaptation and non-linear dynamic processing and it has a broad prospect in settling the complicated non-linear problems. This paper has proposed an algorithm used to solve multi-objective optimization problems and the applications of ACA.

Keywords: Ant Colony Algorithm; Neural Network

1 Introduction

Among the numerous swarm intelligence algorithms, the most remarkable is a self-organizational behaviour of the ant colony, namely that the ant can always find the shortest path between the ant nest and the food source in the foraging. Without visual capacity in finding the food, the ants can only find the path by releasing pheromone on the way. The ant individuals convey information through pheromone and choose the passing path according to the pheromone concentration. The higher pheromone concentration the path has, the higher probability the ants will choose this path. In this way, the process where the ants choose the path has formed a positive-feedback mechanism. The pheromone concentration of the optimal path increases gradually with time while those of the other paths decrease relatively and the ants will finally find the shortest path [1].

Dorigo M, an Italian scholar had first put forward a heuristic swarm intelligent bionic optimization algorithm, namely Ant Colony Algorithm (ACA) by simulating ant's behaviour in finding the shortest path. This algorithm has strong robustness, global convergence ability, distributed computing and feedback mechanism. In a short span of ten years, ACA has been extensively applied in a great number of optimization fields, including combinatorial optimization, network route, function optimization and robot path planning and it has been doing well especially in lots of discrete or continuous combinatorial optimization problems. At present, ACA has been used to settle a lot of practical problems such as Traveling Salesman Problem (TSP), the blind inspection problem of the signals, the assignment problem and Job Shop Scheduling Problem (JSSP) and it has achieved excellent effects [2].

ACA has a short period of development and there are still plenty of problems to be further researched and settled. It has been proved by the theoretical research and the practical applications in many fields that this algorithm has a bright development future. With the development of information technology and the advancement of human cognition, swarm intelligent bionic optimization algorithm will gradually become a useful tool of scientific knowledge; therefore, ant colony optimization algorithm and its applications will eventually be a long-term research focus and a forefront topic and it will arouse more attention from the public and have a broader development prospect [3].

2 Basic ant colony algorithm

Traveling Salesman Problem (TSP) is a typical combinatorial optimization problem and it is also the most successful example of ACA; therefore, ACA is always described with TSP as the application background. TSP can be described as follows: a traveling salesman needs to go back to the departure city after going through n cities only once apart from the starting city and TSP is supposed to find the shortest path among all the cities.

The ants in the natural world can always find the optimal path from the food to the cave or bypassing the obstacles when searching for food or encountering obstacles. The main reason is that the ants will release pheromone on the way and the subsequent ants can choose the next path to go according to the pheromone. The more pheromone a path has, the more times the path

* ***Corresponding author*** e-mail: ssss@dddd.com

is selected, the more excellent performance the path has and the higher probability the subsequent ants choose this path is; therefore, it forms a positive feedback of learning information and gets gradually closer to the optimal solution [4, 5].

TSP can be interpreted by the directed diagram G=(V,E), where V=(1,2…n) is the collection of the nods, E={(ij)} is the collection of the sides and D=(d_{ij}) is the Euclidean distance between i and j. Before using ACA to solve TSPs, it needs to restrict every artificial ant to only choose one city once along a path. After all the ants have searched a complete and legal path, update the corresponding pheromone to every side according to the paths the ants have gone through. In the searching process, the ant computes the probability according to the pheromone and the heuristic information in every path, based on which, the probability that the artificial ant moves from i city to j city at t in the next city is:

$$\rho_{ij}^k(t)=\begin{cases}\dfrac{[\tau_{ij}(t)]^{\alpha}[\eta_{ij}(t)]^{\beta}}{\sum\limits_{s\in D_k}[\tau_{is}(t)]^{\alpha}[\eta_{is}(t)]^{\beta}}, j\in D_k\\ 0,\text{Others}\end{cases}. \tag{1}$$

In this formula, $D_k=\{0,1\cdots n-1\}-tabu_k$ is the city collection to be chosen when the artificial ant arrives at kth city. With a memory function, the artificial ant records the cities it has passed and makes dynamic adjustment with the evolution via $tabu_k(k=1,2,\cdots m)$. As time passes, the information left before disappears gradually. $\eta_{ij}(t)$ is the visibility of side ij and $\eta_{ij}=\dfrac{1}{d_{ij}}$; $\tau_{ij}(t)$ is the pheromone of ij at the time of t; d is the relative importance degree of the pheromone and 1-p is the volatization of pheromone. After some time, the ant finishes one cycle and adjusts the pheromones in every path cording to Formula (2) and (3).

$$\tau_{ij}(t+1)=\rho\tau_{ij}(t)+\Delta\tau_{ij}(t), \tag{2}$$

$$\Delta\tau_{ij}(t)=\sum_{k=1}^{m}\Delta\tau_{ij}^k(t). \tag{3}$$

In the above formula, $\Delta\tau_{ij}^k(t)$ is the pheromone left on ij by k ants in this cycle and $\Delta\tau_{ij}$ is the pheromone increment left on ij in this cycle.

$$\Delta\tau_{ij}^k\begin{cases}\dfrac{Q}{L_k},\text{If kth ant pass through the ij in this loop}\\ 0,\text{Else}\end{cases}. \tag{4}$$

If the k^{th} ant passes ij in this cycle, otherwise.

In Formula (1)-(4), the pheromone intensity Q is a constant and L_k is the total length of the path the k^{th} ant takes in this loop.

At the initial moment, $\tau_{ij}(0)=C$, $\Delta\tau_{ij}=0\ (i,j=0,1\ldots,n-1)$ [6, 7].

3 The applications of ant colony optimization algorithm in artificial neural network

3.1 ARTIFICIAL NEURAL NETWORK

Artificial Neural Networks (ANNs), also called Neural Networks (NNs), is a mathematical model by simulating Biological Neural Networks (BNNs) to conduct information processing. Based on the psychological research achievements of brain, it is aimed to simulate some mechanism of the brain and realize some specific functions. Currently, ANNs have been widely used in many fields.

The basic information processing unit of ANNs is artificial neuron, which is the design foundation of ANNs and its model is indicated in Fig.1.

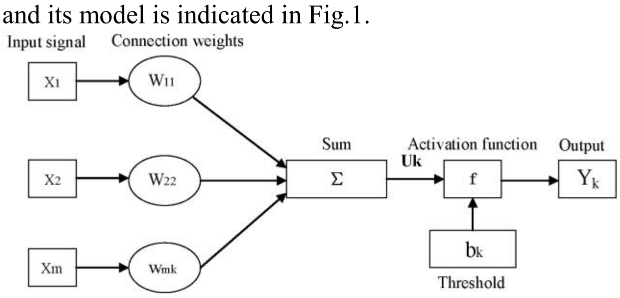

FIGURE 1 Artificial neuron model

An artificial neuron k can be indicated with the following formula.

$$u_k=\sum_{i=1}^{m}w_{ik}x_i \qquad y_k=f(u_k+b_k). \tag{5}$$

In this formula, m is the number of input signals; $x_i(i=1,\ldots,m)$ is the input signal; $w_{ij}(i=1,\ldots,m)$ is the synaptic weight of neuron k (the positive value is at the excited state and the negative value is at the depressed state); u_k is the output of linear combiner of the input signal; b_k is the bias (threshold) of the neuron unit; f is the activation function and y_k is the neuron output signal.

Organize numerous neurons with simple functions through certain topological structure and form the colony parallel processing computing structure. This is the artificial neural network.

According to different connection types, neural network can be divided into two types: layered and interconnected neural network. The layered neural network divides all the neurons into several layers, including input layer, hidden layer and output layer and every layer has some neurons. According to the feedback relationship between layers, the layered neural network can also be divided into the simple feed-forward network, the feedback feed-forward network and the inlayer interconnected feed-forward network. The topological structure of the layered neural network is demonstrated in Fig.2.

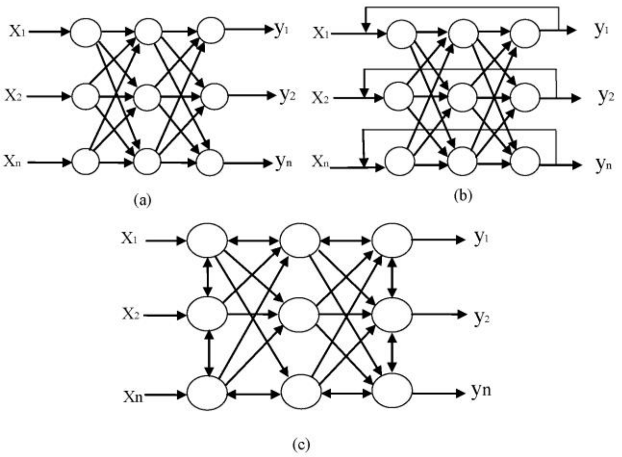

FIGURE 2 The topological structure of layered neural network

a) Simple Feed-Forward Network b)Feed-Forward Network with Feedback c) Inlayer Interconnected Feed-Forward Network

The difference between the interconnected neural network and the layered neural network is that any two units in the interconnected neural network is accessible. According to whether the output of every neuron is connected with other neurons, neural network can also be divided as: fully-connected network and locally-connected network. The topological structure of the interconnected neural network is shown in Fig.3.

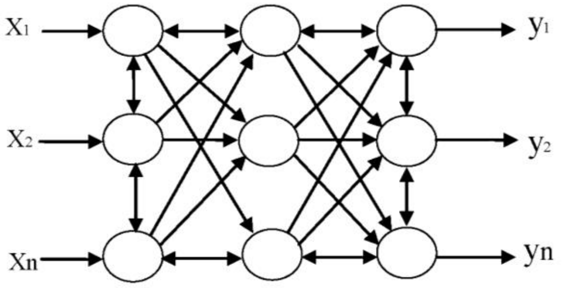

FIGURE 3 The topological structure of interconnected neural network

The basic attributes of neural network reflect the characteristics of neural network. The main characteristics of neural network include:

(1) It has strong robustness and fault tolerance because the information of neural network is distributed and saved in neurons with different network positions.

(2) It has strong information synthesization ability. Every neuron of neural network can process and save information; can handle quantitative and qualitative information and can better coordinate several input information relationships.

(3) It uses parallel processing method. Every neuron in the neural network is parallel in structure and similar processing can be conducted at the same time according to the information received to make faster computing rate.

(4) It has the advantages of self-study, self-organization and self-adaptation. The neurons of the neural network are connected in various ways and the connection strength changes by learning the training samples to make the neural network can handle uncertain systems.

(5) Neural network can fully get close to any complicated non-linear relationship.

3.2 THE NEURAL NETWORK LEARNING ALGORITHM BASED ON ANT COLONY OPTIMIZATION

The basic idea of ACA optimization neural network is to assume that there are N parameters in the neural network to be optimized, including all the weights and thresholds. Firstly, sort these parameters and form a collection I_{pi} of all the possible solutions for the parameter $p_i (1 \le i \le N)$. Then define that m ants go to find food from the ant nest. Every ant starts from the first collection and chooses one element from every collection I_{pi} according to the information status of every element and adjusts the corresponding pheromone to the selected element. When the ant has finished choosing the elements in all the collections, it has arrived at the food source and goes back to the nest in the previous path and adjusts the relevant pheromone. After continuous iteration of the ant, the optimal solution for the parameter can finally be found.

Because BP algorithm is easily trapped in local minimum, this paper has adopted the max-min ant system training neural network which can better solve problems and which has drawn most attention in the improved ant colony algorithm. The neural network training can be seen as an optimization problem, namely to find a group of optimal real number weight combination to minimize the error between the output result and the expected result in this weight. The main steps of the ant colony algorithm optimization neural network are classified as follows:

Step 1: Initialize the pheromone. Equally divide into spn the weight range $[W_{\min}, W_{\max}]$ and the point of every sub-region boundary is an alternative weight. In the initial moment, every point has the same pheromone τ_0. Set the pheromone volatilization coefficient as ρ ; the pheromone increment intensity as Q; the maximum iterations N_{ACO} of ant colony optimization algorithm as and the learning rate of BP algorithm as η.

Set the maximum iterations of BP algorithm as N_{BP}; the training error exit criteria as E_0 and the retaining number of the optimal solutions as σ. The number of ants is m and all of them are in the ant nest.

Step 2: Any ant $k (k = 1, \cdots, m)$ starts from the first collection and moves from one point to another according to the probability computed by the following formula.

$$prob(\tau_i^k) = \tau(i) / \sum_{j=1}^{m} \tau(j) . \qquad (6)$$

The ant records the marks in the passing points, namely to choose a numerical value for the weight and records them in $tabu_k$. When the ant has chosen values for all the weight parameters, it has completed one traversal and all the recorded values have become the parameters of this neural network. Input the training sample; get the corresponding output and compute the error E.

Step 3: Record the σ th group of weights with fewer errors and adjust the pheromone of every element according to the following formula. Assume that the above-mentioned ant has gone through n time units in its foraging.

$$\tau_j(t+n) = (1-\rho)\tau_j(t) + \Delta\tau_j(t)\,, \tag{7}$$

$$\Delta\tau_j(t) = \sum_{k=1}^{m} \Delta\tau_j^k(t)\,. \tag{8}$$

In the above formulas, $\Delta\tau_j^k(t)$ is the pheromone the k th ant has left on the j th element of this collection in this cycle and it can be computed according to the following formula.

$$\Delta\tau_j^k(t) = \begin{cases} \dfrac{Q}{e}, \text{Select jth element of the set} \\ \quad \text{if the ant k in this loop} \\ 0, \quad \text{Else} \end{cases} . \tag{9}$$

In this formula, Q is a constant and is the pheromone increment intensity and e is the maximum output error when the element chosen by all the ants is seen as the weight of the neural network, which can be defined as $e = \max_{n=1}^{s} |O_n - O_p|$. In addition, s is the number of samples and O_n and O_q are the actual output and the expected output of the neural network respectively. Therefore, it can be seen that the fewer the error, the more pheromone increment [8].

Step 4: Repeat the above steps until all the ants have converged to a path, namely having found the optimal solution of the parameters and the cycle ends. The flow chart for the training neural network of ant colony algorithm is shown as Fig.4.

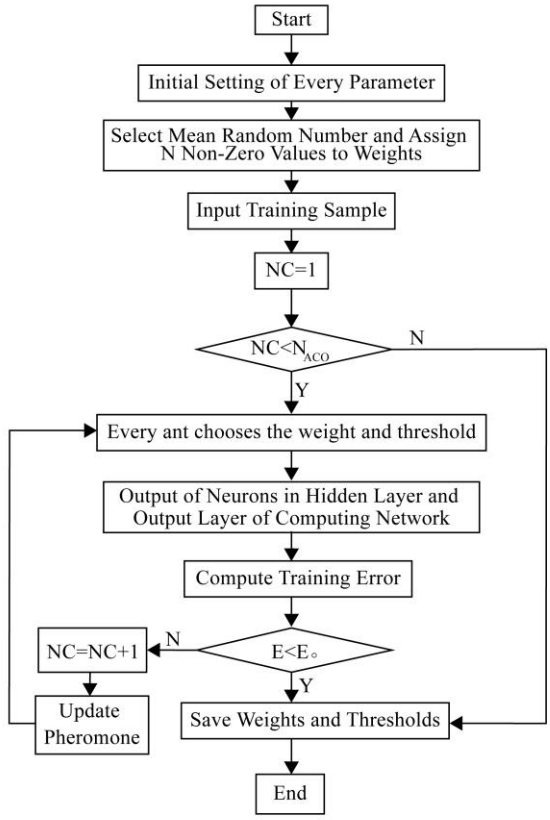

FIGURE 4 The flow chart for training neural network of ant colony algorithm

In this paper, we solve the classification problems with ant colony algorithm and the traditional BP algorithm separately and use the most commonly-used Iris data collection in the public database as the pattern classification data collection with UCI machine. Train the weights and thresholds of the neural networks with ant colony algorithm and the traditional BP algorithm and make comparisons of the results. The neural network in the simulation experiment is the three-layer neural network with a structure of 4-10-3. The algorithm parameters are set as follows: the number of ants S=50; the pheromone volatilization coefficient ρ=0.02 and the total information amount Q=30. The random value of the network weight parameter ranges from -5 to 5. Take the maximum iterations as the end condition of the algorithm. The test error, the classification accuracy and CPU running time in the training phase are displayed in the following table.

TABLE 1 Performance comparison of neural network training

	Test error	**Classification accuracy (%)**	**CPU running time (s)**
Ant colony optimization algorithm	1.353	96.25	20.28
Traditional BP algorithm	2.734	93.68	37.17

It can be seen from the table that ant colony optimization algorithm has fast convergence rate at the training phase and high error precision.

4 Conclusion

Although ant colony algorithm has made big breakthroughs in many application fields since its appearance, there are still some problems about ant colony algorithm to be settled in its practical applications. This paper explains the basic principles and typical characteristics of ant colony algorithm in detail; briefly introduces the model structure and learning algorithm of BP neural network; explores the establishment and optimization algorithm of the optimization neural networks of ant colony algorithm; finishes the network training and test data diagnosis with the extractive samples; makes comparative analysis with the training process and results of BP neural networks and proves the advantages of ant colony optimization neural network.

References

[1] Ghahreman Rezaei, Mohammad Hadi Afshar, etc. 2014 Layout optimization of looped networks by constrained ant colony optimisation algorithm Original Research Article *Advances in Engineering Software* **70**(4) 123-33

[2] Verdaguer M, Clara N, etc. 2013 Application of Ant-Colony-Optimization algorithm for improved management of first flush effects in urban wastewater systems *Science of The Total Environment* **485**(1) 143-52

[3] Zhaojun Zhang, Na Zhang, etc 2014 Multi-satellite control resource scheduling based on ant colony optimization *Expert Systems with Applications* **41**(6) 2816-23

[4] Peng-Yeng Yin, Ray-I Chang, etc. 2012 Niched ant colony optimization with colony guides for QoS multicast routing *Journal of Network and Computer Applications* **40**(4) 61-72

[5] Salabat Khan, Abdul Rauf Baig, etc. 2014 A novel ant colony optimization based single path hierarchical classification algorithm for predicting gene ontology *Applied Soft Computing* **16**(3) 34-49

[6] Zhang B, Qi H, Ren Y T, etc. 2013 Inverse transient radiation analysis in one-dimensional participating slab using improved Ant Colony Optimization algorithms *Journal of Quantitative Spectroscopy and Radiative Transfer* **133**(1) 351-63

[7] Michalis Mavrovouniotis, Shengxiang Yang 2014 Ant colony optimization with immigrants schemes for the dynamic travelling salesman problem with traffic factors *Applied Soft Computing* **13**(10) 4023-37

[8] Juan Rada-Vilela, Manuel Chica, etc. 2013 A comparative study of Multi-Objective Ant Colony Optimization algorithms for the Time and Space Assembly Line Balancing Problem *Applied Soft Computing* **13**(11) 4370-82

Authors

Kuangfeng Ning, born on November 16, 1980, China

Current position, grades: researcher at Hunan International Economics University, China.
University studies: B.S degree in computer science and technology from National University of Defense Technology, 2002, and received the M.S degree in computer application from Middle and Southern University, China in 2008.
Scientific interests: include Signal processing and multimedia technology.

Xiliang Zeng, born on February 7, 1979, China

Current position, grades: researcher at college of computer, Hunan University of International Economics, China.
University studies: B.S degree in computer science and technology from National University of Defense Technology, 2003, and received a M.S. degree in computer science and technology from Hunan University.
Scientific interests: image processing and information security.