International Journal of Hybrid Information Technology
Vol.9, No.4 (2016), pp. 173-182
http://dx.doi.org/10.14257/ijhit.2016.9.4.16

A Novel Hybrid Optimization Algorithm based on Improved ACO and FNN

Dao Jiang

School of Electronic and Information Engineering, Shunde Polytechnic, Shunde 528000 China
lunwen1209@163.com

Abstract

Due to the insufficient of the fuzzy neural network in solving complex problems, an improved ant colony optimization(ACO) algorithm is introduced into the fuzzy neural network in order to propose a novel hybrid optimization(APEACOFNN) algorithm in this paper. In the APEACOFNN algorithm, the self-adaptive pheromone evaporation factor strategy is used to dynamically adjust the pheromone evaporation factor on searching route in order to gradually lessen the amount of information between the optimal path and the worst path, and realize the full searching optimization for decision variable space. Then an improved ACO(APEACO) algorithm is obtained. Aiming at the parameters optimization problem of fuzzy neural network, the proposed APEACO algorithm is used to comprehensively optimize and select the parameters of fuzzy neural network in order to propose a novel hybrid optimization (APEACOFNN) algorithm. Finally, in order to test the effectiveness of the APEACOFNN algorithm, five UCI data sets are selected. The experimental results show that the proposed APEACOFNN algorithm takes the faster approximation objectives and higher solving accuracy.

Keywords*: Hybrid optimization, fuzzy neural network, ant colony optimization,self-adaptive, optimization, UCI data*

1. Introduction

Fuzzy neural network(FNN) is a technology based on combining fuzzy system and neural network. It draws the advantages of the neural network with the accurate fitting ability and learning ability and fuzzy logic with strong structural knowledge expression ability[1,2]. In general, the FNN mainly uses the neural network structure to achieve fuzzy logic reasoning. The topological structure and reasoning rules of FNN is deeply studied in order to determine the general structure model of FNN and the algorithm of FNN is realized. The physical meaning of FNN is clear. Each layer and each neuron have the relative physical meaning to fuzzy logic system. It is a new network expression of the neural network connection principle to the fuzzy logic system. In the course of studying the structure of FNN, the clustering method is used to select the number of fuzzy rules[3]. How to select the optimal number of clusters directly affects the partition of the whole network structure and fuzzy rules. And it is the key to successfully realize the FNN.

The learning of FNN mainly include two parts: structure identification and parameter estimation. Structure identification is to determine the number of rules of fuzzy system, the shape and number of membership function according to the certain performance requirements[4]. Traditional methods is to acquire by using the knowledge and experience of experts. In recent years, many scholars used fuzzy clustering method to obtain the initial fuzzy rule database in order to avoid the blindness and randomness of traditional methods. Parameter learning is further optimize the parameters of FNN after the initial structure was determined. A lot of methods were proposed to determine the structure of FNN and optimize the parameters of FNN. Park et al.[5] proposed GA hybrid scheme

ISSN: 1738-9968 IJHIT

to guarantee both global optimization and local convergence. An aggregate objective function (performance index) with a weighting factor is introduced to achieve a sound balance between approximation and generalization of the model. Tang et al.[6] proposed a hybrid system combining a fuzzy inference system and genetic algorithms - a genetic algorithms based Takagi-Sugeno-Kang fuzzy neural network (GA-TSKfnn) to tune the parameters in the Takagi-Sugeno-Kang fuzzy neural network. Abadeh et al.[7] proposed an evolutionary algorithm to induct fuzzy classification rules. The algorithm uses an ant colony optimization based local searcher to improve the quality of final fuzzy classification system. Tzeng[8] proposed an efficient method to design fuzzy wavelet neural network (FWNN) for function learning and identification by tuning fuzzy membership functions and wavelet neural networks. Dong et al.[9] proposed a PSO-FNNC scheme based on particle swarm optimization algorithm and fuzzy neural network. Yu et al.[10] proposed a new hybrid clustering algorithm that incorporates ACO (ant colony optimization)-based clustering into PCM, namely ACOPCM for noisy image segmentation. Hsu and Juang[11] proposed evolutionary wall-following control of a mobile robot using an interval type-2 fuzzy controller (IT2FC) with species-differential- evolution-activated continuous ant colony optimization (SDE-CACO). All of the free parameters in an online-generated IT2FC are optimized using SDE-CACO, in which an SDE mutation operation is incorporated within a continuous ACO to improve its explorative ability. Park et al.[12] used genetic algorithms with the dynamic variants to optimize the structure and the parameters of FNN. Yang et al.[13] proposed a GA-BP hybrid algorithm for designing polygonal fuzzy neural network. Liu and Guo[14] used ant colony optimization (ACO) algorithm to optimize the parameters of the RBF neural network. Wang and Wang[15] proposed an improved ant colony optimization fuzzy neural network control (ACO-FNC) algorithm for ABS, and the control object of ACO-FNC is slip rate. Hajar et al.[16] proposed a metaheuristic algorithm based on combining a fuzzy multi-objective approach and ant colony optimization (ACO) algorithm to solve the simultaneous reconfiguration and optimal allocation (size and location) of photovoltaic (PV) arrays.

The best merit of the ACO algorithm is the high efficiency for optimizing the FNN. Nevertheless, the ACO algorithm easily falls into the local extreme point and has poor optimization accuracy. So the self-adaptive pheromone evaporation factor strategy is used to improve the basic ACO algorithm. Then improved ACO algorithm is used to comprehensively optimize and select the parameters of fuzzy neural network in order to propose a novel hybrid optimization (APEACOFNN) algorithm.

2. Ant Colony Optimization(ACO) Algorithm and Fuzzy Neural Network

2.1. Ant Colony Optimization(ACO) Algorithm

Ant colony algorithm (ACO) was introduced by Marco Dorigo in the early 1991[17]. It is a metaheuristic inspired by the behavior of real ants in their search for the shortest path to food sources. The ACO algorithm consists of a number of cycles (iterations) of solution construction. During each iteration, a number of ants construct complete solutions by using heuristic information and the collected experiences of previous groups of ants. These collected experiences are represented by the pheromone trail which is deposited on the constituent elements of a solution. Small quantities are deposited during the construction phase while larger amounts are deposited at the end of each iteration in proportion to solution quality. Pheromone can be deposited on the components and/or the connections used in a solution depending on the problem. Each ant randomly starts at one city and visits the other cities according to the transition rule. After the ants complete their routes, the system evaluates the length of the routes. Then, the system uses the pheromone update rule to update the pheromone information. The learning procedure is to update the pheromone information repeatedly.

(1) Transition rule

In the route, the k^{th} ant starts from city r, the next city s is selected among the unvisited cities memorized in J_r^k according to the following formula:

$$s = \arg\max_{u \in J_r^k}[\tau_i(r,u)^{\alpha} \cdot \eta(r,u)^{\beta}] \ \ if\ q \le q_0 (Exploitation) \tag{1}$$

To visit the next city s with the probability $p_k(r,s)$,

$$p_k(r,s) = \begin{cases} \frac{\tau(r,s)^{\alpha} \cdot \eta(r,s)^{\beta}}{\sum_{u \in J_r^k} \tau(r,u)^{\alpha} \cdot \eta(r,u)^{\beta}} & if\ s \in J_r^k \\ 0 & otherwise \end{cases} \quad if\ q > q_0 (Bias\ Exploitation) \tag{2}$$

In two formula, $p_k(r,s)$ is the transition probability, $\tau(r,u)$ is the intensity of pheromone between city r and city u in the i^{th} group, $\eta(r,u)$ is the length of the path from city r to city u, J_r^k is the set of unvisited cities of the k^{th} ant in the i^{th} group, the parameter α and β are the control parameters, q is a uniform probability [0, 1].

(2) The pheromone update rule

In order to improve the solution, the pheromone trails must be updated. Trail updating includes local updating and global updating. The local trail updating formula is given by:

$$\tau(r,u) = (1-\rho)\tau(r,s) + \sum_{k=1}^{m} \Delta\tau_k(r,s) \tag{3}$$

In the formula (3), ρ ($0 < \rho < 1$) is the pheromone trial evaporating rate. $\Delta\tau_k(r,s)$ is the amount of pheromone trail added to the edge(r,s) by ant k between time t and $t+\Delta t$ in the tour. It is given by:

$$\Delta\tau_k(r,s) = \begin{cases} \frac{Q}{L_k} & (r,s) \in \pi_k \\ 0 & otherwise \end{cases} \tag{4}$$

where Q is a constant parameter, L_k is the distance of the sequence π_k toured by ant in Δt.

2.2. Fuzzy Neural Network(FNN)

Fuzzy theory and neural network are based on the mathematical model of dynamic characteristics. Fuzzy theory is used to describe the regular expert knowledge, experience or operation data. And the neural network is used to train sample data. Fuzzy neural network(FNN) is a neural network based on combining the advantages of the fuzzy theory and neural network. So the FNN has the characteristics of dealing with the non-linear and fuzziness and so on. In recent years, more and more experts and scholars contributed to studying the FNN model and proposed a lot of research results. The FNN is composed of five layers of multi-input and single-output. The general FNN structure is shown in Fig.1.

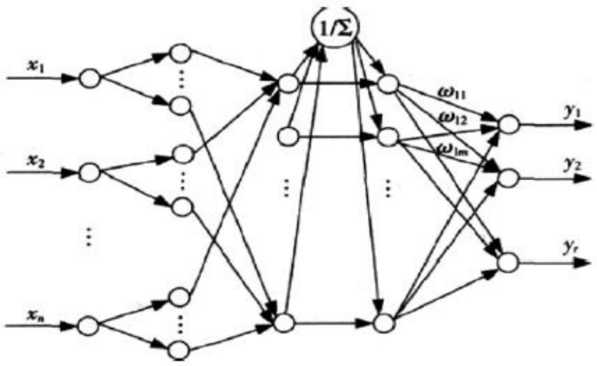


Figure 1. The Structure of the Fuzzy Neural Network

The function of each layer is described as follows:

(1) The first layer (Input layer)

In this layer, input vectors may be accurate numerical vector or fuzzy value. The 'N' is the number of input nodes. The number of output nodes equals to the number of input nodes. The expression is shown as:

$$O_i^1 = I_i = x_i, i = 1,2,3,\cdots,n \tag{5}$$

(2) The second layer (Fuzzification layer)

In this layer, the selected function is used to fuzzify the input variable. This layer can not only realize the membership function of input variable, but also match the fuzzy control rules. The number of nodes equals to all the possible fuzzy rules according to the input variables. The output nodes is corresponding to the product of each Gaussian function. Because the Gaussian function is provided with the good smoothness, the membership function adopts Gaussian function. The expression is shown as:

$$\mu_{ij} = \exp[-\frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}], i = 1,2,\cdots,r; j = 1,2,\cdots,u \tag{6}$$

Where, μ_{ij} is the i^{th} Gaussian function of the j^{th} neuron;

c_{ij} is the i^{th} Gaussian function center of the j^{th} neuron;

σ_{ij} is the i^{th} Gaussian function standard deviation of the j^{th} neuron;

r is the number of input vector (dimension), u is the number of neuron.

The output expression of the j^{th} neuron is shown as:

$$O_j^2 = \exp[\sum_{i=1}^{r} \frac{(x_i - c_{ij})^2}{\sigma_{ij}^2}], j = 1,2,\cdots,u \tag{7}$$

(3) The third layer (Fuzzy reasoning layer)

In this layer, each neuron is one fuzzy rule. The input of membership grade is the former rule, the reasoning uses fuzzy product reasoning to compute the incentive intensity of each fuzzy rule. The expression is shown as:

$$O_j^3 = w_{a1}O_1^2 \times w_{a2}O_2^2 \times \cdots \times w_{au}O_u^2 = \prod_{j=1}^{u} w_{aj}O_j^2 \tag{8}$$

(4) The fourth layer (Anti-fuzzification layer)

In this layer, the clear output of the fuzzy neural network is realized and the anti-fuzzification algorithm is provided with the global approximability. The general fuzzy method of the gravity solution is used in this paper. The expression is shown as:

$$O_j^4 = \frac{O_j^3}{\sum_{j=1}^{u} O_j^3}, \; j = 1,2,\cdots,u \tag{9}$$

(5) The output layer

The precision calculation is realized by the followed expression:

$$y = \sum_{j=1}^{u} w_{bj} O_j^4, \quad j = 1,2,\cdots,u \tag{10}$$

where, w_{bj} is the connection weight of the FNN model.

3. An Improved ACO(APEACO) Algorithm

It is found that the ACO algorithm is a algorithm based on combining heuristic algorithm and positive feedback mechanism of pheromone. In the process of the search optimal solution, the random selection strategy is widely applied, but the random strategy will cause the slower evolutionary speed. The positive feedback mechanism of pheromone is used to strengthen the ants for searching for the optimal solution, but it it's easy to appear the stagnation phenomenon. In order to overcome the these shortcomings, in the process of the search optimal solution, self-adaptive pheromone evaporation factor strategy is used to dynamically adjust the pheromone evaporation factor on searching route in order to gradually lessen the amount of information between the optimal path and the worst path, and realize the full searching optimization for decision variable space. The self-adaptive pheromone evaporation factor is introduced to improve the ACO algorithm. This method is to make the new ant colony optimization algorithm by using pheromone evaporation factor to improve the global convergence. The pheromone evaporation factor ρ in the expression(3) is replaced by self-adaptive $\rho(t)$ with the iteration, then the expression (3) is given:

$$\tau(r,u) = (1-\rho(t))\tau\ (r,s) + \sum_{k=1}^{m} \Delta\tau_k(r,s) \tag{11}$$

where $\rho(t)$ is pheromone evaporation factor, which varies with the iteration. $\rho(t)$ is adaptive to change according to the following expression:

$$\rho(t) = \begin{cases} 0.95\rho(t-1) & if\ 0.95\rho(t) \geq \rho_{\min} \\ \rho_{\min} & other \end{cases} \tag{12}$$

where $\rho_{\min}$ is the minimum value of ρ, the purpose is to prevent too small value of ρ to bring the slow convergence speed. In order to improve global search ability and search speed of the ACO algorithm, the optimal solution is found and saved in the end of each loop search in order to regard as judging the adaptive condition of ρ.

4. A Novel Hybrid Optimization (APEACOFNN) Algorithm

4.1. The Idea of APEACOFNN Algorithm

The proposed APEACO algorithm mainly includes two parts. The ACO algorithm is improved and the FNN is optimized. Aiming at the defects of low convergence speed and easy falling into local optimal solution of the basic ACO algorithm, an improved ACO algorithm based on self-adaptive pheromone evaporation factor strategy (APEACO) is proposed. The self-adaptive pheromone evaporation factor strategy is used to gradually lessen the amount of information between the optimal path and the worst path, realize the full searching optimization for decision variable space, and effectively avoid falling into local optimum for quickening the convergence speed. The improved ACO(APEACO) algorithm can improve the low convergence speed and avoid the easy falling into local optimal solution. Because the structure of fuzzy radial basis function neural networks (FRBFNN) includes the number of neurons in each layer and the connection number between neurons of two adjacent layers. However, the number of neurons in the FRBFNN is to only change in the third layer, and these neurons play an important role. The connection number of neurons is used to describe the number of neurons in the FRBFNN. The structure of the FRBFNN determines the number of nodes in the third layer, the connection number and connection weights between the second layer and the third layer and the fourth layer. So the improved ACO(APEACO) algorithm with higher convergence speed and better optimization ability is used to optimize the FRBFNN in order to propose a novel hybrid optimization(APEACOFNN) algorithm. A switching function is designed to transform the structure optimization and parameter learning problem of the FRBFNN into a simple function optimization problem, which is optimized by using APEACO algorithm in order to obtain the optimal structure and parameter of the FRBFNN for forming the novel hybrid optimization(APEACOFNN) algorithm.

4.2. The Steps of APEACOFNN Algorithm

According to the idea of the proposed APEACOFNN algorithm, the steps of APEACOFNN algorithm are described as follows:

Step1: Initialize the APEACOFNN algorithm

Generate N ants to form one population. $T_{\max}$ is the number of maximum iterations, α is the pheromone factor, β is heuristic factor, the pheromone evaporation factor is ρ, the amount of pheromone is Q. The initial pheromone($\tau(i)$) between two cities is given according to the following equation:

$$\tau(i) = k * a^{-f(x_i)} \qquad (13)$$

where k is a constant ($k > 0$), $0 < a < 1$, $f(x_i)$ is the objective function value. The values of the constants(k and a) are determined according to the actual problem.

Step 2. Each ant chooses one city as its started city.

Step 3. Compute the transition probability

The k^{th} ant constructs its traveling sequence by using the transition rule expression (2).

Step 4. Compute the path length for all ants. Select the next arriving node according to the expression (1) and (2).

Step 5. Update the amount of pheromone

The k^{th} ant allocates its traveling sequence by using the local pheromone update rule expression(3) according to its path length.

Step 6. When all ants completed a cycle, the value of ρ is updated according to the expression (12). Then the global pheromone is updated according to the expression (11).

Step 7. Output optimization solution

Compute whether a better solution is obtained in this step than the last step; if it is, then a global updating is performed on the solution and go into **step 8.** Otherwise repeat **step 2** to **step8**. Obtain the structure of the FRBFNN

The structure of the FRBFNN includes the number of the nodes and the related connection weights of the nodes in the hidden layer. Suppose the given learning sample $\{x(i), \overline{y}(i), i = 1,2,3,\cdots, N\}$, the error of one sampling $\{x, \overline{y}\}$ is defined as: $e = y - \overline{y}$.

Step 9. Optimize the parameters of the FRBFNN

The key of the parameters of the FRBFNN is to find out the optimal value. A unit step function δ is used in here. The optimization problem of the FRBFNN is converted to a pure parametric optimization problem. So the APEACO algorithm is used to optimize the structure and parameter optimization of the FRBFNN.

Step 10. Calculate the error

Suppose the total number of training samples is T, the mean value of the average error of the optimized FRBFNN is given:

$$\varepsilon_{AVG} = \frac{1}{K} \sum_{j \in n} e_j^2(k) \tag{14}$$

where $e_i(k) = d_i(k) - y_i(k)$, $e_i(k)$ is the error signal of the unit.

Step 11. Judge end condition

When the MSE meets the requirement, the APEACOFNN algorithm will end. Otherwise go into **Step 8.** to continue to train the FRBFNN until the maximum iteration is reached.

5. Experiment and Result Analysis

In order to test the effectiveness of the APEACOFNN algorithm, the APEACOFNN algorithm is used to classify the UCI data set. Iris, Glass, Wine, Credit_card, B_cancer_w from the UCI data set are selected to classify for verifying the performance of the APEACOFNN algorithm. The experiment environments are: Matlab 2012b, the Pentium CPU 2.40GHz, 2.0GB RAM with Windows XP operation system. The experimental parameters of the APEACOFNN algorithm are: ants N =50, the number of maximum iterations $T_{\max}$ =1000, the pheromone factor α =0.5, heuristic factor β =2, the amount of pheromone Q =100, the pheromone evaporation factor $\rho \in [0,0.1]$, stochastic selection threshold q_0 =0.90, the iteration algebraic counter $t = 0$. The number of nodes in the input layer equals to the number of dimensions of data set, and the number of nodes in the output layer equals to the number of classes of data set. The training samples are formed by randomly selecting 70% samples from data set. The testing samples are formed by surplus 30% samples from data set. The APEACOFNN algorithm is run independently 50 times. The average classification accuracy and mean square error(MSE) for all testing samples are shown in Table 1.

Table 1. Classification Results of APEACOFNN Algorithm

Index	Data set	Samples	Dimensions	Classes	Average accuracy(%)	MSE
1	**Iris**	150	4	3	97.78	1.536 E-01
2	**Glass**	214	9	6	89.28	5.046 E-02
3	**Wine**	178	13	3	98.65	6.739 E-03

4	**Credit_card**	690	14	2	92.03	7.213 E-03
5	**B_cancer_w**	683	2	2	95.07	5.124 E-01

In order to further prove the effectiveness of the APEACOFNN algorithm, the proposed APEACOFNN algorithm is compared with RBFNN and ACOFNN methods on five data sets from UCI. Each algorithm in the experiment selects the same training set(70%) and the testing set(30%), the average classification accuracy of each algorithm is shown in Table 2 and Fig.2.

Table 2. The Average Classification Accuracy of Each Algorithm

Index	Data set	Samples	RBFNN (%)	ACOFNN(%)	APEACOFNN(%)
1	**Iris**	150	92.14	95.46	97.78
2	**Glass**	214	83.06	86.79	89.28
3	**Wine**	178	92.75	96.91	98.65
4	**Credit_card**	690	88.36	90.67	92.03
5	**B_cancer_w**	683	91.47	94.23	95.07

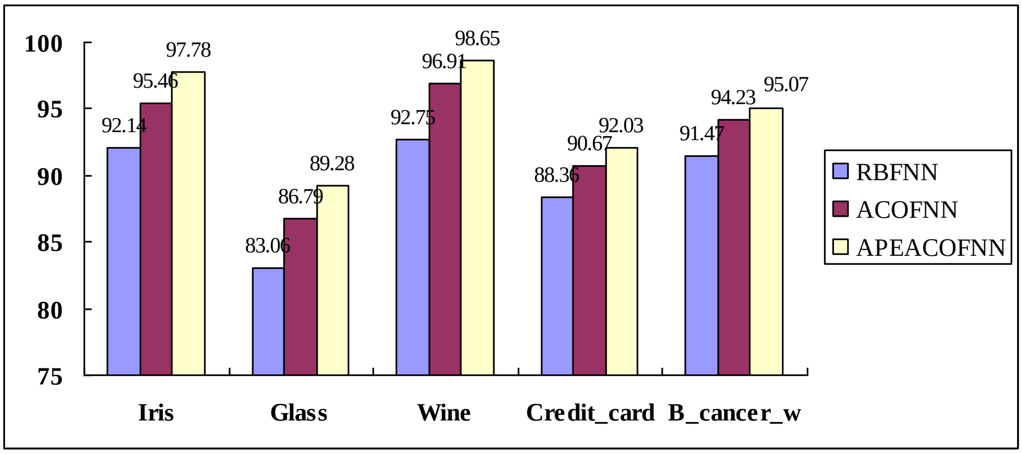


Figure 2. Comparison of Average Classification Accuracy under Testing Data Set

As can be seen from Table 1. and Fig.2, for five UCI data sets, the average classification accuracy of the APEACOFNN algorithm is 97.78%(Iris),89.28%(Glass), 98.65%(Wine), 92.03%(Credit_card) and 95.07%(B_cancer_w). So the average classification accuracy of the APEACOFNN algorithm is significantly better than the RBFNN and ACOFNN methods. The APEACOFNN algorithm can quickly find out the optimal parameters' combination of FNN and obtain the best classification results. And the APEACO algorithm can also ease the slow convergence and oscillation phenomenon of the basic ACO algorithm.

6. Conclusion

In this paper, the fuzzy theory, neural network and improved ant colony optimization algorithm are used to a novel hybrid optimization (APEACOFNN) algorithm. In the APEACOFNN algorithm, the self-adaptive pheromone evaporation factor strategy is used to dynamically adjust the pheromone evaporation factor on searching route in order to gradually lessen the amount of information between the optimal path and the worst path, and realize the full searching optimization for decision variable space. Then the proposed APEACO algorithm is used to comprehensively optimize and select the parameters of fuzzy neural network in order to propose a novel hybrid optimization (APEACOFNN) algorithm. Finally, the UCI data set are used to test the effectiveness of the APEACOFNN algorithm. The results show that the APEACOFNN algorithm can quickly find out the optimal parameters' combination of FNN and obtain the best classification results. And the APEACO algorithm can also ease the slow convergence and oscillation phenomenon of the basic ACO algorithm.

Acknowledgements

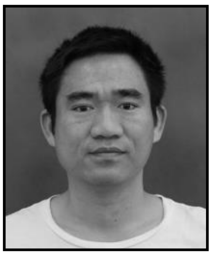
The authors would like to thank all the reviewers for their constructive comments. This research was supported by the Scientific and Technological Innovation Fund of Shunde District(No.20110302006), the National Natural Science Foundation of China (U1433124), the Open Project Program of the Traction Power State Key Laboratory of Southwest Jiaotong University (TPL1403), Open Project Program of Guangxi Key laboratory of hybrid computation and IC design analysis(HCIC201402),Open Project Program of Provincial Key Laboratory for Computer Information Processing Technology, Soochow University (KJS1326). The program for the initialization, training, and simulation of the proposed algorithm in this article was written with the tool-box of MATLAB 2012b produced by the Math-Works, Inc.

References

[1] H. Q. Zhao and J. S. Zhang, "Functional link neural network cascaded with Chebyshev orthogonal polynomial for nonlinear channel equalization", Signal Processing, vol. 88, no. 8, **(2008)**, pp. 1946-1957.
[2] H. X. Li and Z. Liu, "A probabilistic neural-fuzzy learning system for stochastic", IEEE Transaction on Fuzzy System, vol. 16, no. 4, **(2008)**, pp. 898-908.
[3] D. Arun and D. Charles, "Fuzzy neural network models for classification", Applied Intelligence, vol. 12, no. 3, **(2000)**, pp. 207-215.
[4] Y. Zhao, H. J. Gao and S. S. Mou, "Asymptotic stability analysis of neural networks with successive time delay components", Neurocomputing, vol. 71, no. 13-15, **(2007)**, pp. 2848-2856.
[5] H. S. Park and S. K. Oh, "Rule-based fuzzy-neural networks using the identification algorithm of the GA hybrid scheme", International Journal of Control, Automation and Systems, vol. 1, no. 1, **(2003)**, pp. 101-110.
[6] A. M. Tang, C. Quek and G. S. Ng, "GA-TSKfnn: Parameters tuning of fuzzy neural network using genetic algorithms", Expert Systems with Applications, vol. 29, no. 4, **(2005)**, pp. 769-781.
[7] M. S. Abadeh, J. Habibi and E. Soroush, "Induction of fuzzy classification systems via evolutionary ACO-based algorithms", International Journal of Simulation: Systems, Science and Technology, vol. 9, no. 3, **(2008)**, pp. 1-8.
[8] S. T. Tzeng, "Design of fuzzy wavelet neural networks using the GA approach for function approximation and system identification", Fuzzy Sets and Systems, vol. 161, no. 19, **(2010)**, pp. 2585-2596.
[9] X. C. Dong, Y. Y. Zhao, Y. Y. Xu, Z. Zhang and P. Shi, "Design of PSO fuzzy neural network control for ball and plate system", International Journal of Innovative Computing, Information and Control, vol. 7, no. 12, **(2011)**, pp. 7091-7103.
[10] J. M. Yu, S. H. Lee and M. G. Jeon, "An adaptive ACO-based fuzzy clustering algorithm for noisy image segmentation", International Journal of Innovative Computing, Information and Control, vol. 8, no. 6, **(2012)**, pp. 3907-3918.
[11] C. H. Hsu and C. F. Juang, "Evolutionary robot wall-following control using type-2 fuzzy controller with species-DE-activated continuous ACO", IEEE Transactions on Fuzzy Systems, vol. 21, no. 1, **(2013)**, pp. 100-112.

[12] B. J. Park, W. D. Kim, S. K. Oh and Sung-Kwun, "A design of dynamically simultaneous search GA-based fuzzy neural networks: Comparative analysis and interpretation", Journal of Electrical Engineering and Technology, vol. 8, no. 3, **(2013)**, pp. 621-632.
[13] Y. Q. Yang, G. J. Wang and Y. Yang, "Parameters optimization of polygonal fuzzy neural networks based on GA-BP hybrid algorithm", International Journal of Machine Learning and Cybernetics, vol. 5, no. 5, **(2014)**, pp. 815-822.
[14] J. Liu and Z. H. Guo, "Network traffic prediction using radial basis function neural network optimized by ant colony algorithm", Sensors and Transducers, vol. 172, no. 6, **(2014)**, pp. 224-228.
[15] C. P. Wang and L. Wang, "Research on the ant colony optimization fuzzy neural network control algorithm for ABS", Communications in Computer and Information Science, vol. 483, **(2014)**, pp. 130-139.
[16] B. T. Hajar M. H. Ali and M. Rizwan, "Simultaneous reconfiguration, optimal placement of DSTATCOM, and photovoltaic array in a distribution system based on fuzzy-ACO approach", IEEE Transactions on Sustainable Energy, vol. 6, no. 1, **(2015)**, pp. 210-218.

Author



Dao Jiang. He is a Lecturer, He received the Engineering Master degree in computer software from Jinan University in 2005, Guangzhou, China. The main research directions: Artificial Intelligence, Hybrid optimization algorithm.