



Weights Optimization Method of Differential Evolution Based on Artificial Bee Colony Algorithm

Ying Wu¹, Zibo Qi¹, Ling Jiang¹, Zifeng Dai², Chen Zhang², Changsheng Zhang²(✉), and Jian Xu²

¹ Shenyang Fire Science and Technology Research Institute of MEM, Shenyang 110034, People's Republic of China

² Software College of Northeastern University, Shenyang 110819, People's Republic of China
paper820@sohu.com, zhangchangsheng@mail.neu.edu.cn

Abstract. Differential evolution algorithm is a search and optimization strategy that simulates the process of biological evolution. In the initial stage of the algorithm, it is necessary to generate a series of deep neural networks with sufficient accuracy as the initial population of subsequent algorithms. In this article, an artificial bee colony search strategy is added to the cross-operation of the differential evolution algorithm to optimize the weight value. The artificial bee colony algorithm search operator is introduced to guide the search of the population to avoid individuals in the population from falling into a local optimal situation. The experiments in this article verify the validity of the method through the handwritten digit recognition data set. The final results show that in the process of obtaining the initial population, using the differential evolution weight optimization method of the artificial bee colony search strategy optimizes the process of the fitness calculation in the model. It significantly improves the accuracy of the first-generation population and speeds up the overall process of the algorithm.

Keywords: Artificial bee colony algorithm · Differential evolution · Neural network · Ensemble learning · Deep learning

1 Introduction

In recent years, the research of ensemble learning algorithm has developed vigorously. For almost any kind of machine learning algorithm, the idea of ensemble learning can be used to improve the accuracy and generalization of the algorithm. The ensemble learning model completes the learning task and improves the prediction accuracy of the final model by constructing and combining multiple learners. It is an excellent idea and method in the field of machine learning [1–4].

The differential evolution algorithm DE [5] is a stochastic model that simulates biological evolution. Through repeated iterations, those individuals who adapt to the environment are preserved. Because the algorithm has a simple structure, is easy to

implement, does not require gradient information, and has fewer parameters, it has attracted the attention and research of many scholars as soon as it was proposed. Similar to other intelligent algorithms, DE also has the problems of being easily trapped into a local optimum, slow convergence in the later stages of evolution, may not be able to search for optimal solutions for problems that are too complicated, and the calculation accuracy is not high.

The artificial bee colony algorithm [6] uses the communication, transformation and cooperation between bees of different roles to achieve swarm intelligence optimization by researching the behavior of bees during honey collection. Compared with the classic optimization methods, the artificial bee colony algorithm has simple operation, less control parameters, high search accuracy and strong robustness.

The artificial bee colony search strategy has strong exploration ability, and shows excellent optimization performance when optimizing complex multimodal problems. Therefore it is very suitable for combining it with differential evolution algorithms to improve the performance of the model. Lingling Huang et al. [7] proposed a differential evolution algorithm with artificial bee colony search strategy to solve the problems of premature phenomenon and slow convergence speed of differential evolution algorithm. The artificial bee colony search strategy was used to guide the population to help the algorithm jump out of the local best with its strong exploration ability. In addition, in order to improve the global convergence speed of the algorithm, an initialization method based on anti-learning is used. Through simulation experiments on 12 standard test functions and comparison with other algorithms, it shows that the proposed algorithm has a faster convergence speed and strong ability to jump out of local optimum. However, the accuracy of the generalization ability of this model is not high enough, the generalization ability is limited, and it only obtains good results by testing in some functions. In this article, an artificial bee colony search strategy is added to the cross-operation of the differential evolution algorithm based on the ensemble learning model to optimize the weight value. The artificial bee colony algorithm search operator is introduced to guide the search of the population to avoid individuals in the population falling into a local optimal situation and improve the generalization ability of the neural network model.

The other parts of this article are as follows: Section 2 introduces a specific differential evolution weight optimization method based on artificial bee colony algorithm; Section 3 uses the method and model described in this article to perform experiments on the data set, and the results show the accuracy of the final output is higher, which proves the effectiveness of the method in this paper; Section 4 summarizes the method in this paper and briefly proposes future research content.

2 Evolutionary Ensemble Learning Model Based on Artificial Bee Colony Algorithm

In the existing models built based on the differential evolution algorithm, the algorithm often falls into a local optimum at the later stage of the execution, which leads to the final convergence rate being too slow. And it is often unable to solve complex multimodal problems or the accuracy of the results is low. These disadvantages limit the scope of the differential evolution algorithm. This article proposes that the hyper parameters

that need to be determined in advance in the neural network are used as individuals of the evolutionary algorithm, the evolutionary algorithm is optimized by the artificial bee colony algorithm, and then the neural network with higher accuracy is obtained through the optimized evolutionary algorithm. The specific process of optimizing the evolutionary algorithm is shown in Fig. 1.

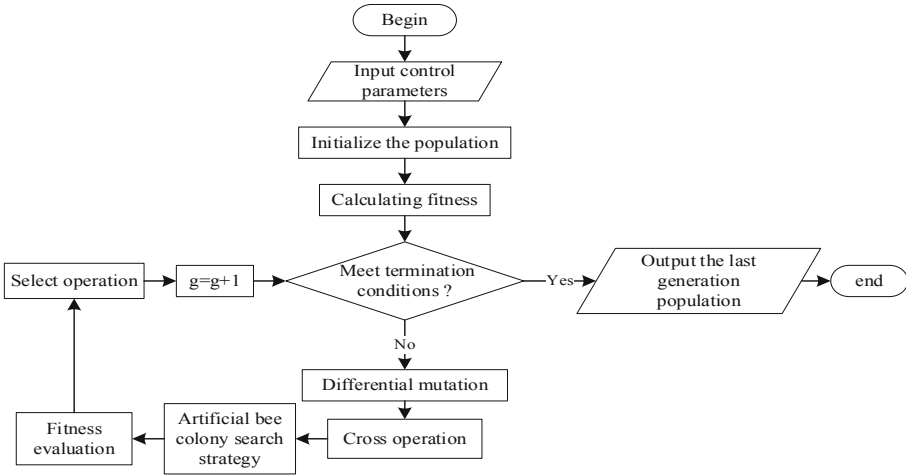


Fig. 1. Flow chart differential evolution algorithm based on artificial bee colony search strategy.

First, enter the preset control parameters, perform population initialization, calculate the initial fitness and use it as the evaluation standard for subsequent fitness. Then when the existing population size is smaller than the control value of the offspring, new offspring are generated cyclically. The cyclic process first differentiates and then crosses to get the new individuals, and then uses the artificial bee colony search strategy to search for more excellent individuals around the new individuals. Finally, the optimal individual is selected as a new individual to generate a new offspring individual. When the population size reaches the threshold, it exits the cycle and outputs the final generation population of the process, which is used as the initial population to input into the subsequent algorithm process.

2.1 Fitness Calculation Method Based on Artificial Bee Colony Search Strategy

Artificial bee colony (ABC) was proposed by Turkish scholar Karaboga in 2005. Its basic idea is to inspire the bee colony to cooperate with each other to complete the honey collection task through individual division of labor and information exchange. Although a single bee’s own ability is limited, without a unified command, the entire bee colony can always find high-quality nectar sources more easily. Artificial bee colony algorithm is a new type of intelligent optimization algorithm by simulating the honey collecting process of bees. It consists of three parts: food source, hired bee and non-employed bee. The core of the algorithm includes 3 parts: leading bees searching for

honey source; leading bees share honey source information and follow the bees to select a honey source to search with a certain probability; the scout bees randomly search in the search space.

In this paper, an artificial bee colony algorithm is added between the cross operation and the selection operation of the differential evolution algorithm to optimize the combined weight value in the process of fitness calculation. With the evolution of the population of the differential evolution algorithm, when the accuracy of the corresponding network output results of all individuals in the population is high enough, the last generation of the population at this time is regarded as the initial type of the subsequent algorithm Group, so as to ensure the accuracy of each network in the follow-up algorithm can be high enough.

Firstly, the initialization operation is carried out, and m individuals are randomly and uniformly generated in the corresponding decision space (each individual is the real number code of the super parameter to be optimized). Each individual is composed of n -dimensional vector (assuming that there are n parameters to be optimized), see formula (2.1).

$$X_i(0) = (x_{i,1}(0), x_{i,2}(0), \dots, x_{i,n}(0)), i = 1, 2, 3 \dots, M \tag{2.1}$$

The initialization method of the j -th dimension vector of the i -th individual is shown in formula (2.2).

$$X_{i,j}(0) = L_{j_min} + rand(0, 1)(L_{j_max} - L_{j_min}) \tag{2.2}$$

$$i = 1, 2, 3 \dots, M, j = 1, 2, 3, \dots, n$$

L_{j_min} and L_{j_max} represent the upper and lower boundaries of the value of the j -th dimension vector and X represents every individual.

Then the mutation operation is started. The classic differential strategy of differential evolution algorithm is used to implement individual mutation. That is to say, two different individuals in the population are randomly selected, and their vector differences are scaled to synthesize vectors with the individuals to be mutated, as shown in formula (2.3).

$$X'_i(g) = X_{r1}(g) + F \cdot (X_{r2}(g) - X_{r3}(g)) F \in [0, 2], i \neq r1 \neq r2 \neq r3 \tag{2.3}$$

At the same time, we need to strictly control the boundary of each element of the new individual. If it goes beyond its corresponding range, it is different from the initial random generation method. In this paper, we map it to the appropriate range by some operations. Then the crossover operation is started. According to the crossover probability $cr \in [0,1]$, the elements of each dimension are selected from the variation individual and the original individual, and the crossover individual is obtained. See formula (2.4).

$$V_{i,j} = \begin{cases} X'_{i,j}(g + 1), & rand(0, 1) \leq cr \\ X_{i,j}(g), & else \end{cases} \tag{2.4}$$

At this time, according to the search rules of artificial bee colony, search for the individual with the lowest adaptive value around the crossed individual, and select as a

new individual. See (2.5) for the formula.

$$z_{i,j} = x_{i,j} + \phi_{i,j}(x_{i,j} + x_{k,j}) \tag{2.5}$$

In the next selection operation, according to the greedy rule, cross individuals and original individuals are selected based on the fitness function value to build a new generation of population, as shown in formula (2.6).

$$X_i(g + 1) = \begin{cases} V_i(g), f(V_i(g)) < f(X_i(g)) \\ X_i(g), \textit{else} \end{cases} \tag{2.6}$$

$$k \in \{1, 2, \dots, M\}, j \in \{1, 2, \dots, D\}, k \neq i, \phi_{i,j} \in [-1, 1]$$

The fitness function here is an objective function constructed by using the gap between the output result of the network corresponding to the specific parameter vector and the real result, which is used to evaluate the merits and demerits of individuals. See formula (2.7) for the calculation method.

$$\textit{minimize} : f(x) = \frac{1}{N} \sum_{i=1}^N (NET_x^i - REAL^i) \tag{2.7}$$

$NET_{(x)}^i$ represents the output of the depth neural network constructed by the parameter vector x in the i th training sample. $REAL^i$ represents the real result of the i th training sample.

Finally, through the above steps, the operations of crossover, mutation and selection are carried out repeatedly, and the population is continuously updated. When certain conditions are met, the single objective optimization part is completed. The condition here is set as: when the accuracy of network output results corresponding to all individuals in a generation population meets a threshold, the cycle will be terminated.

2.2 Weight Optimization Method Based on Artificial Bee Colony Algorithm

In the artificial bee colony algorithm, the artificial bee colony is divided into three categories: leading bee, following bee, and investigating bee. During each search process, the leading bee and following bee are mining food sources in succession, that is for finding the optimal solution, and the investigating bee observes whether it is trapped in a local optimum, if it is trapped in a local optimum, it randomly searches for other possible food sources. Each food source represents a possible solution to the problem, and the amount of nectar from the food source corresponds to the quality of the corresponding solution (fitness value fit_i). The detailed steps of weight optimization method based on artificial bee colony algorithm proposed in this paper are as follows.

Step 1: Initialize the population: Initialize each parameter, the total number of bee colonies S_n , the number of the food source is collected(the maximum number of iterations MCN), and the control parameter limit. Each solution x_i is a D-dimensional vector. D is the dimension of the problem. Determine the problem search range, and an initial solution $x_i(i = 1, 2, \dots, S_n)$ is randomly generated within the search range;

Step 2: Calculate and evaluate the fitness of each initial solution;

Step 3: Set cycling conditions and start cycling;

Step 4: Leading bee to perform a neighborhood search on the solution x_i according to formula (2.5) to generate a new solution (food source) v_i , and calculate its fitness value fit_i ;

Step 5: Perform greedy selection according to formula (2.8): if the fitness value of v_i is better than x_i , replace x_i with v_i , otherwise leave x_i unchanged;

$$v_i = \begin{cases} v_i (fit_{v_i} > fit_{x_i}) \\ x_i (fit_{x_i} \leq fit_{v_i}) \end{cases} \quad (2.8)$$

Step 6: Calculate the probability p_i of the food source according to formula (2.9);

$$p_i = \frac{fit_i}{\sum_{k=1}^{S_n} fit_k} \quad (2.9)$$

Step 7: The following bee selects a solution or a food source according to the probability p_i , searches for a new solution (food source) v_i according to formula (2.5), and calculates its fitness;

Step 8: Perform greedy selection according to formula (2.8): if the fitness value of v_i is better than x_i , replace x_i with v_i , otherwise leave x_i unchanged;

Step 9: judge whether there is a solution to give up. If there is, the detection bee will randomly generate a new solution to replace it according to formula (2.10);

$$x_{i,j} = x_{min,j} + rand(0, 1) \times (x_{max,j} - x_{min,j}) \quad j \in \{1, 2, \dots, D\} \quad (2.10)$$

Step 10: record the optimal solution so far;

Step 11: judge whether the cycle termination condition is met. If it is, the cycle ends and the optimal solution is output. Otherwise, return to step 4 to continue searching.

The pseudo code for the above procedure is shown below.

Algorithm 2.2. The process of artificial bee colony.

Algorithm Single-objective differential evolution algorithm

Input

Cross: the crossed individual
 Iter_max: population number
 nPop: the number of honey source
 nLooker: the number of max search times

Output

The best value of fitness and the best individual

Begin

Initializing honey source location and calculating initial fitness according to eq.(2.7);

for iter **in** Iter_max:

for i **in** nPop:

 Find next honey source according to eq.(2.5);

 Greedily choose the individual according to eq.(2.8);

End for;

 Calculating the selection probability matrix;

For k **in** nLooker:

 Find next honey source according to eq.(2.5);

 Greedily choose the individual according to eq.(2.8);

End for;

For k **in** nPop:

If the times of source reaching the nLooker:

 Find next honey source according to eq.(2.5);

End if;

End for;

End for

Record the best individual and the best value of fitness;

End

In this process, the main advantage of the artificial bee colony algorithm is that it has a strong randomness when selecting. That is to say, in the process of exploring the best fitness of chromosomes, the probability of selecting individuals with good fitness and those with poor fitness is the same. It can be seen from the above pseudo code that for each generation of individuals, the time complexity of the artificial bee colony search strategy is $T_n = O(m * n)$, where m represents the number of populations and n represents the number of initial honey sources.

3 Experiment

In the experimental part of this paper, firstly, we use the differential evolution method based on the artificial bee colony algorithm to get the model with high accuracy. Then we use the multi-objective differential evolution algorithm to weigh the accuracy and difference of the network, and construct many networks with both accuracy and difference. Finally, we use the idea of integrated learning to combine these networks into the model. The experiment is carried out on MNIST dataset, and the results of the model with artificial bee colony search strategy and the model without it are compared and analyzed. Experimental results show that the proposed method can improve the accuracy of the model.

In order to reduce the interference of other factors, in the comparative test, other parameters are set to the same value. The specific parameter settings of the network collection stage obtained in the experiment are shown in Table 1.

Table 1. Parameter settings for the part of generating candidate deep networks.

Algorithm name	Parameter name	Specific value
Single-objective differential evolution algorithm	Population size	10
	Variation control parameters	0.5
	Cross control parameters	0.8
Multi-objective differential evolution algorithm	Population size	20
	Neighborhood number	6
	Evolutionary algebra	10

In the model, for the part of differential evolution weight optimization method based on artificial swarm search strategy proposed in this paper, the initial parameters of artificial swarm set in the experiment are shown in Table 2.

Table 2. Parameter setting of artificial bee colony search strategy.

Algorithm name	Parameter name	Specific value
Artificial bee colony search strategy	Population number	50
	Initial honey sources number	100
	Maximum number of iterations	100

Under the premise that the overall algorithm remains unchanged, the experiment compares and analyzes the experimental results with or without the artificial bee colony search strategy model. In order to avoid the influence of random chance on the experimental results, repeat each experiment ten times, take the average value of the results as the final result, and the specific data is shown in Table 3.

Table 3. The influence of differential evolution algorithm based on artificial swarm search strategy on experimental results.

Artificial bee colony search strategy	Training set accuracy (%)	Validation set accuracy (%)	Test set accuracy (%)
No	99.6485	99.4870	99.2430
Yes	99.7589	99.6140	99.4740

It can be seen from the above table that the accuracy of the model has been improved by combining the artificial bee colony search strategy, which shows that the method proposed in this paper can improve the performance of the model and achieve the purpose of optimizing the model.

As many existing models have achieved good results [8] on the MNIST data set, the model obtained in this paper is compared with the existing model. The results are shown in Table 4.

Table 4. The performance of other models on MNIST dataset.

Classifier	Test error rate (%)
K-Nearest neighbors	
K-NN with non-linear deformation (IDM)	0.54
K-NN with non-linear deformation (P2DHMDM)	0.52
K-NN, shape context matching	0.63
Convolutional nets	
Committee of 7 conv. Net, 1-20-P-40-P-150-10 [elastic distortions]	0.27 ± 0.02
Committee of 35 conv. Net, 1-20-P-40-P-150-10 [elastic distortions]	0.23
Large/deep conv. Net, 1-20-40-60-80-100-120-120-10 [elastic distortions]	0.35

As shown in Table 4, considering that the convolution neural network used in this paper has a small number of layers, the results show that it basically achieves the performance of some networks with deep layers. Therefore, the model proposed in this paper can exceed some networks with many layers, but it can not catch up with those models with deep convolution layers as a whole.

4 Summary

In the process of building the deep integration network model, this paper proposes a differential evolutionary weight optimization method based on the artificial bee colony algorithm. In general, the general differential evolution algorithm has insufficient search ability for the best fitness individuals, which greatly restricts the performance of the model, while the artificial bee colony algorithm has a strong search ability, which can be used in the process of cross selection to explore the best fitness individuals and improve the overall performance of the model. In the experimental part of this paper, an integrated learning model is constructed by using this method, which can effectively improve the generalization ability and make up for the shortcomings of the existing differential evolution algorithm. The integrated learning model and the single neural network model are tested repeatedly on the handwritten digit recognition data set. By comparing the experimental results on the training set, the verification set and the test set, the method has achieved good results.

Combined with the research content of this paper, there is still room for improvement in the following aspects in the future: (1) How to set the initial parameters of the artificial bee colony algorithm can get the required individuals more accurately and quickly. (2) In terms of network differences, more indicators can be considered to judge the differences between networks in multiple dimensions, such as the number of network layers and other factors. (3) This method will be applied to the identification scenes of fire-prone devices to improve identification efficiency.

References

1. Zhou, Z.H., Wu, J., Tang, W.: Ensembling neural networks: many could be better than all. *Artif. Intell.* **137**(1–2), 239–263 (2002)
2. Partalas, I., Tsoumakas, G., Hatzikos, E.V., et al.: Greedy regression ensemble selection: theory and an application to water quality prediction. *Inf. Sci.* **178**(20), 3867–3879 (2008)
3. Martín, I., de Diego, Á., Serrano, C.C., Cabello, E.: Face verification with a kernel fusion method. *Pattern Recogn. Lett.* **31**(9), 837–844 (2010). <https://doi.org/10.1016/j.patrec.2009.12.030>
4. Takemura, A., Shimizu, A., Hamamoto, K.: Discrimination of breast tumors in ultrasonic images using an ensemble classifier based on the AdaBoost algorithm with feature selection. *IEEE Trans. Med. Imaging* **29**(3), 598–609 (2010)
5. Storn, R., Price, K.: Differential evolution – a simple and efficient heuristic for global optimization over continuous spaces. *J. Glob. Optim.* **11**(4), 341–359 (1997). <https://doi.org/10.1023/A:1008202821328>
6. Karaboga, D., Basturk, B.: A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *J. Glob. Optim.* **39**, 459–471 (2007)
7. Huang, L., Liu, S., Gao, W.: Differential evolution with the search strategy of an artificial bee colony algorithm. *Control Decis.* **27**(11), 1644–1648 (2012)
8. MNIST Homepage. <https://yann.lecun.com/exdb/mnist/>. Accessed 12 Feb 2020