Improved Artificial Bee Colony Algorithm Based on Harris Hawks Optimization

Liyi Zhang[1], Zuochen Ren[2], Ting Liu[1*], Jinyan Tang[2]

[1] Information Engineering College, Tianjin University of Commerce, China
[2] College of Science, Tianjin University of Commerce, China
zhangliyi@tjcu.edu.cn, zc.ren.98@gmail.com, liuting@tjcu.edu.cn, misstangi@sina.com

Abstract

Artificial bee colony algorithm, as a kind of bio-like intelligent algorithm, used by various optimization problems because of its few parameters and simple structure. However, there are also shortcomings such as low convergence accuracy, slow convergence speed, and not easy to jump out of the local optimum. Aiming at this shortcoming, this paper proposes an evolutionary algorithm of improved artificial bee colony algorithm based on reverse learning Harris Hawk (HABC). The basic inspiration of HABC comes from the good convergence of Harris Hawk algorithm in the process of finding the best point of the function. First, introduce the Harris Hawks optimization progressive rapid dives stage in the onlooker bee phase to speed up the algorithm convergence; Secondly, Cauchy reverse learning is added in the scout phase to make the algorithm development more promising areas in order to find a better solution; Finally, 13 standard test functions and CEC-C06 2019 benchmark test results are used to test the proposed HABC algorithm and compare with ABC, Markov Chain based artificial bee colony algorithm (MABC), dragonfly algorithm (DA), particle swarm optimization (PSO), learner performance based behavior algorithm (LPB), and fitness dependent optimizer (FDO). Compared with other algorithms, the convergence speed, optimization accuracy and algorithm success rate of the HABC algorithm are relatively excellent.

Keywords: Artificial bee colony algorithm, Harris hawks optimization, Progressive rapid dives, Cauchy reverse learning

1 Introduction

In recent years, numerical optimization problems have been a hot issue in the fields of mathematics and physics and engineering. As a kind of bionic intelligent algorithm, bee colony algorithm is also used to solve optimization problems in most cases. The algorithm was proposed by Karaboga [1] in 2005. Like the ant colony algorithm [2] and the fish swarm algorithm [3], the bee colony algorithm is inspired by the behavior of bees looking for the source of bees, and optimizes the search through the intelligent guidance of the colony generated by the bees in the colony looking for good nectar sources. The ABC has fewer parameters, and its principle is simple [4], and has been applied to various real-world optimization problems.

With the development of science and the continuous deepening of research on bee colony algorithm, the bee colony algorithm has a very broad application prospect. For example, deep learning plays a role in the medical field by recognizing X-ray images [5-6], and the ABC algorithm is also applied to the medical field by processing images [7]. In other aspects, the algorithm also solves the problem of flow shop scheduling [8], the problem of weather forecasting [9] and so on. Since the bee colony algorithm was proposed, various improvements to the bee colony algorithm and algorithm research continue to develop. Today, scholars have proposed many improvements and algorithm fusion schemes, and these improvements have achieved good results and solved different actual problems of the situation.

In 2019 Harris hawks optimization (HHO) was first proposed by Iranian scholar Heidari [10]. HHO's inspiration is mainly derived from the cooperative behavior of eagles in nature, and it is inspired by the three stages of eagle hunting behavior search, search and development conversion, and development. HHO has the characteristics of Simple logic and fewer parameters, so it has been applied in many aspects in recent years, such as photovoltaic model parameter estimation [11], dynamic optimization problem [12], image segmentation problem [13-14].

Aiming at the shortcomings of slow convergence speed, low convergence accuracy, and easy to fall into local optimality, an evolutionary algorithm based on improved ABC algorithm based on inverse learning Harris Hawk (HABC) is proposed. Utilizing the fast convergence characteristics of Harris Eagle algorithm, introducing parameter escape energy E, searching for the optimal value according to two position update strategies, and searching for the direction of the optimal value, which improves the convergence speed and accuracy of the algorithm. At the same time, the Cauchy reverse learning strategy is introduced into the reconnaissance bee stage to generate the Cauchy reverse solution, and the final solution needs to be selected from the two known terms through "greedy selection", so that the algorithm can jump out of the local optimal while improving the global search ability. Through simulation experiments, the HABC algorithm is compared with the original algorithm, and the p-value is calculated to verify its different degrees, and then it is compared with another improved algorithm. The convergence speed, convergence accuracy and the ability to jump out of the local optimum have been greatly improved.

*Corresponding Author: Ting Liu; E-mail: liuting@tjcu.edu.cn
DOI: 10.53106/160792642022032302016

The work of this article is as follows:

(1) The gradual rapid diving phase of Harris Eagle algorithm is introduced into the bee colony algorithm, combined with the local optimization ability of Harris Eagle algorithm to accelerate the convergence speed of the artificial bee colony algorithm, when the escape energy is large, the current optimal value is larger. The range attachment is searched. When the escape energy is small, the search is performed near the current optimal value.

(2) In the reconnaissance bee stage, the Cauchy reverse learning strategy is introduced to generate the Cauchy reverse solution, and in order to select the final solution, the method of "greedy selection" is used. This method can better improve the ability of the algorithm to jump out of the local optimum.

This article is divided into the following sections. The first part serves as a literature review. The second part introduces the basic principles of the ABC algorithm. The third part proposes ABC algorithm based on reverse learning Harris Hawk (HABC). The four parts are comparative experiments, and the last part draws some conclusions and prospects.

2 Literature Review

2.1 Artificial Bee Colony Algorithm

The improvement of ABC can generally be divided into three categories: the improvement of the search mechanism, the improvement of other links, and the combination with other algorithms. Xu Chen et al. [15] by studying the principle of the firework explosion algorithm, the explosion search mechanism is tried to be added to the bee colony algorithm reconnaissance stage. Through the explosion search, the algorithm can develop a better food source; Bandi Rambabu et al. [16] proposed to use the mutant butterfly adjustment operator in the monarch butterfly optimization algorithm to replace the employment phase in ABC algorithm, in order to maintain the balance between development and exploration, and prevent premature fall into local optimization and delay convergence; Huaglory Tianfield et al. [17] integrated a variety of differential search strategies and adaptive mechanisms into the ABC algorithm. Through the differential search strategy, in the loop process, many variables are generated each time, and the items are combined through the mutation and crossover within the algorithm, so that the algorithm is more easy to solve inseparable problems; Nirmala Sharma et al. [18] were inspired by the phenomenon of beer foam decay and proposed a new location update, which was applied to the bystander level to keep the algorithm in balance between exploration and development and solve the problem of workshop scheduling. Shaohuai Dai et al. [19] proposed a chaotic multi-dimensional discrete artificial bee colony algorithm based on optimal difference orientation by introducing optimal difference matrix to update food source, determining update dimension by multi-dimensional search and greedy selection, initializing population by Tent chaotic sequence and so on; Zhiqiang Zhang [20] A. by adding the optimal value guide to the algorithm, the algorithm can converge faster, and it also strengthens the search ability in the algorithm field. And through the current optimal solution, the field search is carried out randomly in the employed bee stage, thereby improving the global search ability of the algorithm.

Through the improvement of the ABC algorithm by many scholars, it is found that the ABC algorithm is relatively weak in terms of convergence speed and jumping out of the local optimum. This article uses a combination of algorithms to make up for the shortcomings of the ABC algorithm at these two points.

2.2 Harris Hawks Optimization

Andi Tang et al. [21] according to the shortcomings of the algorithm, such as the convergence speed is too slow, the convergence accuracy is not high, and the algorithm is easy to fall into the local optimum during operation, the elite hierarchy strategy is introduced to make full use of the dominant population of Harris eagle species, so as enhance the population diversity and improve the convergence speed and accuracy of the algorithm. Secondly, the key parameters of the algorithm are adjusted by tent chaotic mapping, and then a nonlinear energy factor adjustment strategy is used to balance the algorithm. Finally, the Gaussian random walk strategy is used to disturb the optimal individual, and when the algorithm is stagnant, the random walk strategy is used to effectively jump out of the local optimization; Yiming Ma et al. [22] improved the Harris Eagle algorithm, changed the fitness of the algorithm through Gaussian distribution, and obtained an initial solution of the Chan algorithm [23] with low computational complexity, and randomly selected an individual to replace it with the initial solution to get the initial population, so as to reduce the unnecessary global search of the algorithm; Mohamed Abd Elaziz et al. [24] combined the strong exploration ability of Harris Eagle algorithm and the mining ability of moth optimization algorithm, proposed a Harris Eagle-Moth hybrid optimization algorithm, and proved that it is better than other algorithms through 36 mathematical problems and 4 constrained design problems.

In summary, as a newly proposed algorithm in recent years, HHO algorithm has not been studied much, but HHO has a good optimization ability. This article analyzes HHO and selects its optimization part to improve the ABC algorithm.

3 Artificial Bee Colony (ABC)

ABC algorithm is inspired by the cooperative search feature of bee colony. The algorithm uses three different bees (employed bee, onlook bee, scout bee) in the colony to cooperate with each other to find food sources. Each candidate solution represents the food source of a bee colony, and the fitness value of the candidate solution represents the quality of each food source. The sum of the hired bees and the bystanders formed the initial bee colony. These two parts each account for 50% of the total. In the algorithm, each employed bee corresponds to a food source, and the responsibility of the employed bee is to pass the food information to the onlooker bee, and onlooker bee uses this information to filter out what they need Information to follow. When the location of the food source corresponding to employed bee has not changed for many times, scout bees come by employed changes in bees, and at the same time, scout bees will make new changes in food source information

The specific implementation process of the ABC algorithm is as follows.

3.1 Initialise

In the initialization phase, a uniformly distributed initial population containing food sources is randomly generated. The dimension of each food source is D, and the i food source $x_i (i = 1, 2, ..., SN)$ is initialized by formula (1):

$$x_{id} = x_{id,min} + rand(0,1)\left(x_{id,max} - x_{id,min}\right)? \quad d = 1, 2, \ldots, D \tag{1}$$

Where, $x_{id,\max}$ as the upper and $x_{id,\min}$ as the lower limits of the search domain, and $rand(0,1)$ is a random number on $(0,1)$.

Calculate food concentration by formula (2), which is fitness:

$$fit(\mathbf{x}_i) = \begin{cases} 1 + f(\mathbf{x}_i) & f(x_i) \geq 0 \\ \dfrac{1}{1 + \left|f(\mathbf{x}_i)\right|} & f(x_i) < 0 \end{cases} \tag{2}$$

In the formula, $f(x_i)$ is a function of the objective and $fit(x_i)$ is the food concentration of the algorithm, the i food source, respectively.

3.2 Employed Phase

Employed bee to search for food sources and produce new food sources $\mathbf{v}_i = (v_{i1}, v_{i2}, \ldots, v_{iD})$

$$v_{id} = \begin{cases} x_{id} + r_{id}\left(x_{id} - x_{qd}\right) & if\ d = d_{rand} \\ x_{id} & other \end{cases} \tag{3}$$

Where, q is randomly generated between $[1, SN]$, and $q \neq i$; d_{rand} is randomly generated between $[1, D]$; $r_{id} \in [-1,1]$, it controls the scope of the search. ◦

When the domain search is over, a new food source will appear, and you need to compare two food sources and choose one of them. The method of "greedy selection" is used to select the preferred rule according to the food concentration.

3.3 Onlook Phase

When the hire bee completes the first stage of search, the employed bee will pass the information to the onlooker bee. Before transmitting the information, roulette will be used to select the food concentration. The food concentration is higher, the greater the probability of being selected. Similarly, if the food concentration is too low, it will not be selected. The calculation formula for food concentration is as follows (4) Shown:

$$p(x_i) = \frac{fit(\mathbf{x}_i)}{\sum_{i=1}^{SN} fit(\mathbf{x}_i)} \tag{4}$$

In the onlooker bee stage, the algorithm will use equation (3) to perform a domain search to generate a new food source. When the concentration of the newly generated food source is greater than that of the original food source, the original food source will be replaced, otherwise, the original food source will remain unchanged.

3.4 Scout Phase

When the ABC algorithm falls into the local optimum, the food source concentration will stop updating. When the number of stopped cycles has not changed many times, the employed bee will become scout bee. Formula (1) represents the generation of a new food source. The bee will change back to the original employed bee and begin to continue to send signals to the onlooker bee.

During the cycle, it will continuously record, compare, and update the current optimal solution. When the number of cycles does not reach the preset number, it will return to the employed bee and start a new round of optimization. When the number of cycles reaches the preset number, the cycle ends.

4 HABC Algorithm

4.1 Algorithm Introduction

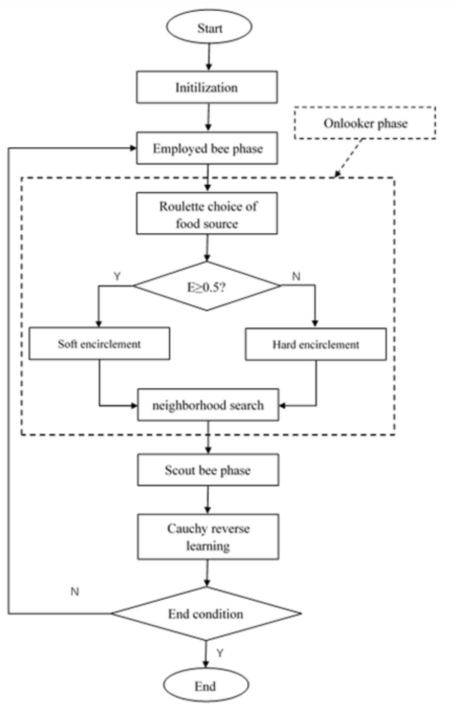

Figure 1. HABC framework

ABC is good at exploration, but not good at development. An efficient search process requires good exploration capabilities. In HHO, Harris Hawk search can generate new individuals around several well-distributed individuals. This operator can effectively use useful information and find better solutions. This paper introduces the Harris Eagle search stage into ABC, and proposes a new hybrid HABC framework.

The HABC framework generates a new food source in the initialization phase, and then enters five search phases: (1) Hire bee phase (2) Harris Hawk search phase (3) Bystander bee phase (4) Scout bee phase (5) Cauchy anti To the stage. (1)(3)(4) stage comes from ABC, (2)(5) stage comes from HHO. The HABC framework is shown in Figure 1.

4.1.1 Introducing Harris Hawk

The ABC algorithm is widely used in continuous optimization problems because of its fewer parameters and strong global search capabilities. However, the critics of the ABC algorithm include slow convergence speed and low convergence accuracy. The fundamental reason is: the main function of onlooker bees is to speed up the algorithm convergence and enhance the local mining ability. However, in the traditional algorithm, the field of onlooker bees search is random and lacks the guidance of the optimal value, which makes the convergence speed slow. As a new meta-heuristic algorithm, Harris hawks optimization has fast convergence speed and strong local search ability [25], which is mainly reflected in the progressive rapid dives stage. At this stage, there is the guidance of the optimal value , and it is easier to approach the global optimal direction during the update process. In order to be able to find more high-quality nectar sources, the Harris Hawk's progressive rapid dives was introduced onlooker bee phase, and an advance update was performed so that it could find the optimal value of the function faster. The specific implementation is as follows:

First, introduce the parameter escape energy E:

$$E = 2E_0(1-\frac{t}{T}) \tag{5}$$

Where E_0 is the initial energy, and its value varies randomly within $(-1,1)$; t is the current number of cycles, T is the total number of cycles. When $|E|<1$, the food source is easy to detect; When $|E|>1$, the food source is not easy to detect. After the introduction of the bee colony, E represents the size of the chance that the bees can detect the optimal food source when searching for the food source. It will decrease as the number of iterations increases, and the reconnaissance opportunity will increase. When $|E|\geq 1$, onlooker bee using the traditional (3) domain search; when $|E|<1$, the algorithm introduces the progressive rapid dives in the Harris hawks optimization to perform a pre-search before the traditional domain search, which optimize the convergence and speeds up the algorithm optimization accuracy.

(1) When $|E|\geq 0.5, r<0.5$ is in the soft besiege, the exact position of the food source is not easy to be discovered by the bees. At this time, the bee colony will use two strategies to carry out soft encirclement more intelligently like Harris Hawk.

Strategy One:

$$\mathrm{Y} = \mathrm{X}_{best} - E\left|J\mathrm{X}_{best} - \mathrm{X}\right| \tag{6}$$

where $J = 2(1-r)$ Is a random number $(-2,0)$.

When strategy one is invalid, implement strategy two.

Strategy Two:

$$Z = Y + S \times LF(D) \tag{7}$$

Where S is a D dimensional random vector, and LF is the Levy function.

$$LF(x) = 0.01 \times \frac{\mu+\sigma}{|\upsilon|^{\frac{1}{\beta}}} \tag{8}$$

$$\sigma = \left(\frac{\Gamma(1+\beta)\times\sin(\frac{\pi\beta}{2})}{\Gamma(\frac{1+\beta}{2})\times\beta\times 2^{(\frac{\beta-1}{2})}} \right)^{\frac{1}{\beta}}$$

Where $\beta = 1.5$, μ, υ is a number between 0 and 1 , Γ Is the gamma function. Therefore, the location update strategy of the bees at this stage is executed according to the following equation:

$$X = \begin{cases} Y & if\ F(Y) < F(X) \\ Z & elseif\ F(Z) < F(X) \\ X & otherwise \end{cases} \tag{9}$$

(2) When $|E|<0.5, r<0.5$ is in the hard besiege, the food source does not have enough energy to cover itself, but it may not be discovered by the bees. Therefore, the bees will reduce the distance from the food source before collecting nectar, forming a hard envelopment, and take two strategies for hunting.

Strategy One:

$$\mathrm{Y} = \mathrm{X}_{best} - E\left|J\mathrm{X}_{best} - \mathrm{X}_m\right| \tag{10}$$

Where X_m represents the average position of the current bee colony.

When strategy one is invalid, use strategy two. Strategy two:

$$Z = Y + S \times LF(D) \tag{11}$$

Position update formula at this stage:

$$X = \begin{cases} Y & if\ F(Y) < F(X) \\ Z & elseif\ F(Z) < F(X) \\ X & otherwise \end{cases} \tag{12}$$

4.1.2 Introducing Cauchy Reverse Learning

The exploration ability of the ABC algorithm is very good, but it is easy to fall into the local optimum [26]. The scout bee helps the algorithm jump out of the local optimum, but in the traditional algorithm, because the randomly generated scout bee has not been estimated, the convergence speed is unpredictable.

When there are points generated by Cauchy reverse learning and original points, the two points will always be closer to the best point. And if you choose this closest point, it will make the algorithm more convenient to find the global optimization. Cauchy reverse learning is applied to the scout phase, and Cauchy reverse learning is performed on the newly generated food source, which is intended to make the search behavior of the scout bee easier to jump out of the local optimum through reverse learning.

H. R. Tizhoosh [27] first proposed a new concept in 2005, Opposition-Based Learning (OBL). Some scholars combined this method with particle swarms [28] in 2015. From the simulation results, the effect is positive.

In reverse learning, the reverse solution $\bar{\boldsymbol{x}}_i(t) = (\bar{x}_1, \bar{x}_2, \ldots, \bar{x}_D)$ of individual x_i is calculated by Eq.(13):

$$\bar{\boldsymbol{x}}_i = x_{id,\max} + x_{id,\min} - \boldsymbol{x}_i \tag{13}$$

The Cauchy reverse point is generated by generating a random number x_i^f between the intermediate point and the reverse point. This random number is called the Cauchy reverse point. which is calculated using Eq. (14)

$$\boldsymbol{x}_i^f = rand\left(\frac{x_{id,\max} + x_{id,\min}}{2}, \bar{\boldsymbol{x}}_i\right) \tag{14}$$

In the scout phase, it is judged whether there is a solution that needs to be abandoned. If there is a solution that needs to be abandoned, a new solution is randomly generated according to Eq.(1), and then the Cauchy reverse solution is generated through Eq.(13) and Eq.(14). The final selected solution needs to be selected through the "greedy selection" in the algorithm. Through the above steps, the algorithm's search in the global scope can be improved, and it is easier to jump out of the local optimum during calculation.

4.2 Algorithm Flow

Step 1. initialization. Set the relevant parameters of ABC algorithm, and initialize the population by Eq. (1);
Step 2. Employed phase. Employed bee to search through the field through Eq. (3) to obtain new food sources;
Step 3. Onlooker phase. Onlooker bee to get the leading bee information, calculate the probability of the food source by Eq. (4) and select it by roulette;
Step 4. The selected food source is pre-updated through Eq. (5)-Eq. (12) in HHO to speed up the accuracy of the algorithm's convergence;
Step 5. Scout phase. The food source has not been updated many times, employed bee to become a scout bee, use formula (1) to generate new food sources, and then use Eq. (13) and Eq. (14) to perform Cauchy reverse learning on the newly generated food source to filter out more Valuable food sources continue to iterate;
Step 6. Determine whether the number of cycles reaches the maximum: No, return to Step2；Yes, end the algorithm and output the optimal value.

5 Simulation

To test the performance and efficiency of the Harris Hawks based on ABC Algorithm (HABC), The experiments in this article use MATLAB R2014b to simulate the HABC algorithm. Running a PC equipped with Inter Core i5-6300HQ (2.3 GHz) and 16 GB RAM.

This chapter includes 4 experiments. Experiment 1 compares the results of ABC algorithm, MABC algorithm [29] and HABC algorithm. Experiment two, the reliability research of HABC algorithm. In the third experiment, HABC passed 12 basic test functions to compare horizontally with LPB [30] and FDO [31]. The fourth experiment is to compare CEC2019 with PSD and DA. Select 13 classic test functions for experiment, different function setting parameters are the same, as shown in Table 1.

Table 1. Text function

Function	Function name	Range	$f_{\min}$
$f1 = \sum_{i=1}^{n} x_i^2$	sphere	$[-100,100]$	0
$f2 = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos\left(\frac{x_i}{\sqrt{i}}\right) + 1$	Griewank	$[-600,600]$	0
$f3 = \sum_{i=1}^{n}\left(x_i^2 - 10\cos(2\pi x_i) + 10\right)$	Rastrigin	$[-5.12,5.12]$	0
$f4 = \sum_{i=1}^{n-1}\left[100\left(x_{i+1} - x_i^2\right)^2 + \left(x_i - 1\right)^2\right]$	Rosenbrock	$[-30,30]$	0
$f5 = -20\exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}\right) - \exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i)\right) + 20 + e$	Ackley	$[-32,32]$	0
$f6 = \sum_{i=1}^{n}\lfloor x_i + 0.5\rfloor^2$	Step	$[-100,100]$	0
$f7(x) = \sum_{i=1}^{n}\lvert x_i\rvert + \prod_{i=1}^{n}\lvert x_i\rvert$	schwefel 2.22	$[-10,10]$	0

$f8=\sum_{i=1}^{n} ix_i^2$	SumSquares	$[-10,10]$	0
$f9=\sum_{i=1}^{n} ix_i^4+random[0,1]$	quartic	$[-1.28,1.28]$	0
$f10=\sum_{i=1}^{n}(\sum_{j=1}^{i} x_j)^2$	schwefel 1.2	$[-100,100]$	0
$f11=\frac{\max\{\lvert x\rvert,1\le i\le n\}}{i}$	schwefel 2.21	$[-100,100]$	0
$f12=\frac{\pi}{n}\{10\sin(\pi y_1)+\sum_{i=1}^{n-1}(y_i-1)^2[1+10\sin^2(\pi y_{i+1})]$ $+(y_n-1)^2\}+\sum_{i}^{n}u(x_i,10,100,4)$ $y_i=1+\frac{x+1}{4}$ $\mathrm{u}(x_i,\mathrm{a,k,m})=\begin{cases}k(x_i-a)^m\ x_i>a\\ 0-a<x_i<a\\ k(-x_i-a)^m\ x_i<-a\end{cases}$	penalized	$[-50,50]$	0
$f13=0.1\left\{\sin^2(3\pi x_1)+\sum_{i=1}^{n}(x_i-1)^2[1+\sin^2(3\pi x_i+1)]+(x_n-1)^2[1+\sin^2(2\pi x_n)]\right\}$ $+\sum_{i=1}^{n}u(x_i,5,100,4)$	penalized2	$[-50,50]$	0

5.1 Algorithm Optimization Comparison

This article uses the parameters in the literature [29]: D=30, SN=20, Limit=100, MaxCycle=500. Perform 30 independent experiments and record the average, standard deviation, optimal value and worst value of each algorithm. The results are shown in Table 2.

Table 2 shows that in the process of algorithm simulation, the optimization ability of the improved algorithm HABC is stronger than the original ABC and the improved algorithm MABC.

The convergence curve of the average solution of the ABC algorithm and the HABC algorithm is shown in Figure 2.

Table 2. Comparison of optimization accuracy of different algorithms

Function	Algorithm	Best	Worse	Mean	Std
*f*1	ABC	6.17E-05	2.69E-03	6.43E-04	7.22E-04
	MABC	1.11E-12	3.36E-05	1.37E-06	6.02E-06
	HABC	**5.78E-113**	**3.40E-41**	**1.13E-42**	**6.20E-42**
*f*2	ABC	1.24E-03	1.45E-01	5.13E-02	3.47E-02
	MABC	1.80E-14	**3.00E-06**	**1.94E-07**	**6.28E-07**
	HABC	**0.00E+00**	1.62E-01	3.73E-02	7.16E-02
*f*3	ABC	9.56E+00	1.00E+02	9.21E+00	2.46E+00
	MABC	4.28E-08	**4.41E-01**	**1.50E-02**	**7.91E-02**
	HABC	**0.00E+00**	2.98E+02	7.78E+01	1.07E+02
*f*4	ABC	7.38E+00	1.49E+03	1.52E+02	2.96E+02
	MABC	**4.86E+00**	1.03E+02	4.46E+01	3.35E+01
	HABC	2.87E+01	**2.88E+01**	**2.87E+01**	**1.42E-02**
*f*5	ABC	4.52E-02	1.83E+00	6.59E-01	4.57E-01
	MABC	9.59E-08	2.27E-03	2.83E-04	5.49E-04
	HABC	**8.88E-16**	**4.44E-15**	**1.95E-15**	**1.66E-15**
*f*6	ABC	0.00E+00	1.00E+00	3.33E-02	1.83E-01
	MABC	0.00E+00	1.00E+00	1.33E-01	3.46E-01
	HABC	**0.00E+00**	**0.00E+00**	**0.00E+00**	**0.00E+00**
*f*7	ABC	4.08E-03	8.07E-02	1.33E-02	1.37E-02
	MABC	1.77E-03	2.02E-02	8.61E-03	3.90E-03
	HABC	**3.80E-56**	**9.35E-37**	**3.11E-34**	**1.70E-33**

*f*8	ABC	2.99E-06	1.64E-03	1.86E-04	3.18E-04
	MABC	4.80E-12	2.57E-04	2.14E-05	6.22E-05
	HABC	**8.46E-120**	**4.80E-46**	**1.60E-47**	**8.76E-47**

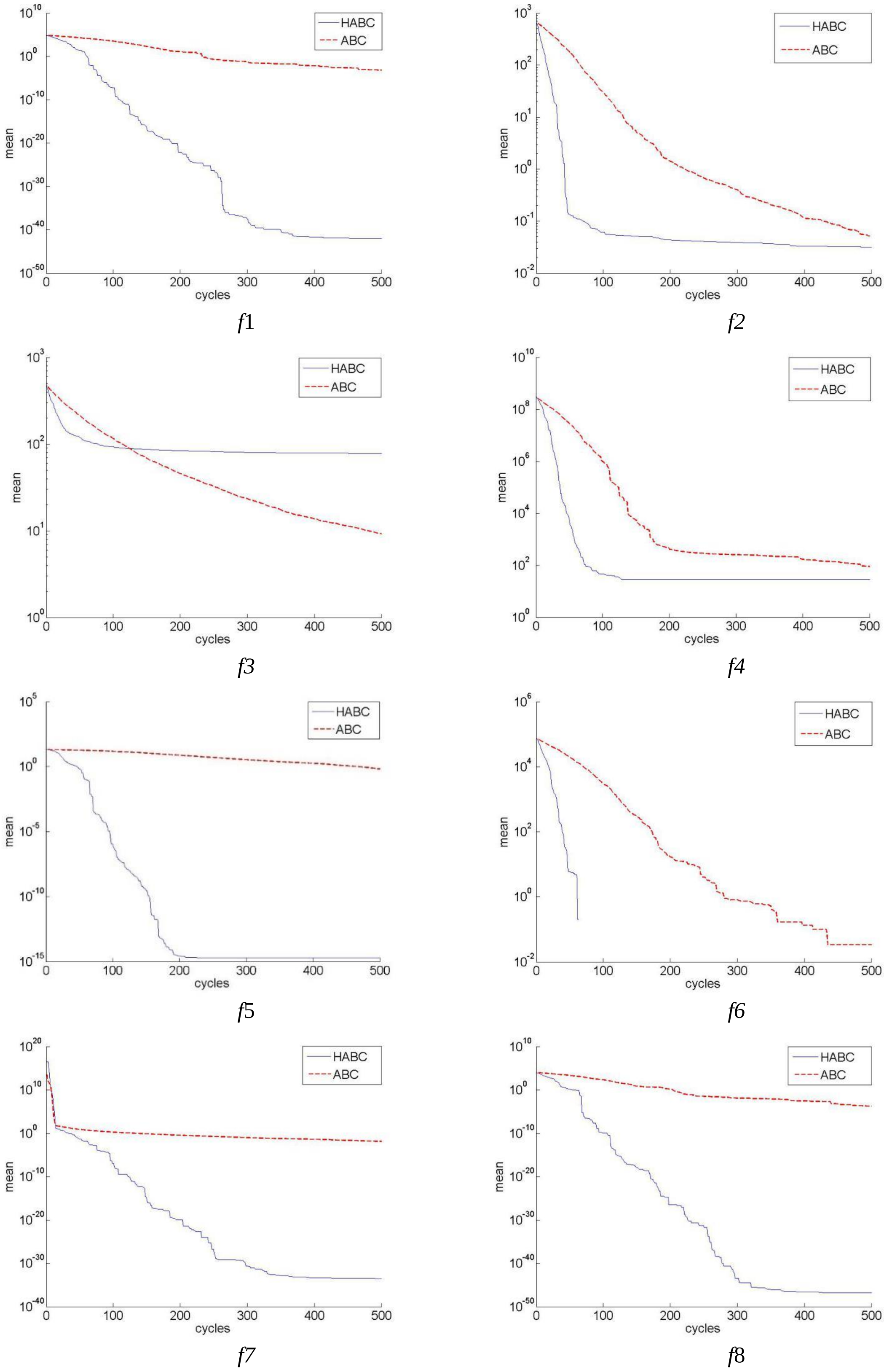

Figure 2. Convergence graphs of ABC and HABC algorithms

The trend graph in Figure 2 already clearly shows that under the HABC algorithm, the convergence speed and accuracy have been improved. *f*1, *f*5, *f*6, *f*7, *f*8 have obvious improvements in function accuracy and convergence speed. *f*2 and F4 have a certain convergence speed after the improvement. When *f*3 is simulated, the probability of this algorithm falling into the optimal value trap is very small, but the optimum value is much better than the original algorithm. The contrast between the improved algorithm and the original algorithm is introduced in section 5.2.

5.2 Statistical Text

In order to show that Table 2 and Figure 2 are statistically significant, the p-value of the Wilcoxon test is found for the 8 test functions used in Section 5.1, and the results in Table 3 are analyzed. As shown in Table 3, the Wilcoxon rank sum test function performs a statistical test on the importance of the results. The experiment uses SPSS for operation, which is inconvenient for scientific notation. Here, it is uniformly accurate to three decimal places and does not affect the result analysis. The p-values of the eight classic test functions in Table 3 prove that f3 is not significant, and f6 is not significant because the ABC algorithm can also be optimized to 0 in most cases. Therefore, for most test functions, HABC shows significantly better results than ABC, that is, the p-value is less than 0.05, which proves that the HABC results are statistically significant compared with the original ABC algorithm.

In order to show that Table 2 and Figure 2 are statistically significant, the p-value of the Wilcoxon test is found for the 8 test functions used in Section 5.1, and the results in Table 3 are analyzed. As shown in Table 3, the Wilcoxon rank sum test function performs a statistical test on the importance of the results. The experiment uses SPSS for operation, which is inconvenient for scientific notation. Here, it is uniformly accurate to three decimal places and does not affect the result analysis. The p-values of the eight classic test functions in Table 3 prove that *f*3 is not significant, and f6 is not significant because the ABC algorithm can also be optimized to 0 in most cases. Therefore, for most test functions, HABC shows significantly better results than ABC, that is, the p-value is less than 0.05, which proves that the HABC results are statistically significant compared with the original ABC algorithm.

5.3 Compare with Other New Algorithms

HABC was compared with LPB [30], FDO [31]. HABC is carried out under the experimental conditions of SN=20 and Limit=100. Other conditions are consistent with the literature [30-31].

In Table 4, the better results are marked in bold. As shown in Table 4, for *f*3, *f*10 and *f*11, the HABC algorithm is weaker than other algorithms, LPB performs better in *f*3, and FDO performs better in *f*10 and *f*11. This proves that the global optimization of the algorithm is not better than the comparison algorithm. However, comparing the unimodal test functions *f*1, *f*4, *f*6, and *f*7 of the HABC algorithm with several other algorithms shows that the HABC algorithm is found to have better development capabilities. The results of test functions *f*2, *f*3, *f*5, *f*9, *f*12, *f*13 prove that HABC has superior exploratory and completeness, avoiding local optima, and compared with LPB, FDO, it has more advantages between exploration and development stages. Good performance. balance. It can be concluded from Table 4 that the HABC algorithm passes 12 test functions and ranks first among these algorithms because the effect of 9 test functions is that the HABC algorithm performs better.

Table 3. The Wilcoxon rank-sum text overall runs for classical benchmark test function

Function	HABC Vs. ABC
*f*1	**0.000**
*f*2	**0.014**
*f*3	0.206
*f*4	**0.000**
*f*5	**0.000**
*f*6	0.317
*f*7	**0.000**
*f*8	**0.000**

Table 4. Comparison of results of the classical benchmark function between HABC, LPB, FDO

Fun		*f*1	*f*2	*f*3	*f*4
HABC	Ave	**7.06E-43**	**2.36E-03**	2.10E+01	**1.03 E+01**
	Std	**2.71E-42**	7.08E-02	1.51E+01	**1.27E+01**
LPB	Ave	1.88E-03	1.79E-02	**0.001567**	1.68E+01
	Std	2.09E-03	**1.35E-02**	**0.001842**	2.22E+01
FDO	Ave	7.47E-21	5.69E-01	1.46E+01	2.35E+01
	Std	7.26E-19	1.04E-01	5.20E+00	5.98E+01
Fun		***f5***	***f6***	***f7***	***f9***
HABC	Ave	**2.78e-15**	**0.00E+00**	**1.24E-44**	**1.92E-03**
	Std	2.03e-15	**0.00E+00**	**4.87E-44**	**2.95E-03**
LPB	Ave	1.79E-02	2.03E-03	5.24E-03	4.98E-03
	Std	1.35E-03	2.78E-03	3.65E-03	2.97E-03
FDO	Ave	3.99E-15	1.42E-18	9.39E-06	5.44E-01
	Std	**6.38E-16**	4.75E-18	6.91E-06	3.15E-01
Fun		***f10***	***f11***	***f12***	***f13***
HABC	Ave	1.11E-01	1.49E-01	**2.44E-13**	**1.60e-13**
	Std	5.05E-02	1.04E-01	**3.10E-13**	**4.69e-13**

LPB	Ave	36.47488	3.94E-01	3.09E-04	3.09E-04
	Std	29.22415	1.36E-01	5.12E-04	5.12E-04
FDO	Ave	**8.55E-07**	**6.69E-04**	1.03E+01	1.03E+01
	Std	**4.40E-06**	**2.49E-03**	7.42E+00	7.42E+00

5.4 CEC-C06 2019 Benchmark Test Functions

In this stage, the CEC06-2019 benchmark test function is used to evaluate HABC. These 10 test functions are shown in Table 5, which are improved by Professor Suganthan on the single-objective optimization problem. These test functions are called "hundreds of challenges". In the first three test sets, CEC01 to CEC03 are moved and rotated, while CEC04 to CEC10 are not moved and rotated, and CEC04 to CEC10 are converted into a minimization problem with a dimension of 10 in the interval [-100,100]. All CEC functions are variable, and the global optimum of these test sets is set to 1. HABC is compared with two optimization algorithms: DA, PSO. The reasons are as follows: 1) They are all open source with simple parameter settings and a large amount of literature for testing. 2) The three algorithms are all developed based on particle swarm optimization. The data in the following table are all from the literature [31].

Each algorithm is carried out under the conditions of 30 cycles and 500 iterations. As shown in Table 6, HABC is mostly superior to other algorithms. PSO is better than HABC under the conditions of CEC04, CEC05, CEC07, and CEC09, but the difference is not very different. HABC algorithm still has room for improvement.

Table 5. CEC-C06 2019 benchmark test functions

Function	Text Functions	Dim	Range	$f_{\min}$
CEC 01	Storn's Chebyshev Polynomial Fitting Problem	9	$[-8192, 8192]$	1
CEC 02	Inverse Hilbert Matrix Problem	16	$[-16384, 16384]$	1
CEC 03	Lennard-Jones Minimum Energy Cluster	18	$[-4, 4]$	1
CEC 04	Rastrigin	10	$[-100, 100]$	1
CEC 05	Grienwank	10	$[-100, 100]$	1
CEC 06	Weiersrass	10	$[-100, 100]$	1
CEC 07	Modified Schwefel	10	$[-100, 100]$	1
CEC 08	Expanded Schaffer	10	$[-100, 100]$	1
CEC 09	Happy Cat	10	$[-100, 100]$	1
CEC 10	Ackley	10	$[-100, 100]$	1

Table 6. CEC-C06 2019 benchmark test results

Function	HABC		DA		PSO	
	Ave	Std	Ave	Std	Ave	Std
CEC 01	**5.12E+10**	**6.24E+10**	5.43E+10	6.69E+10	1.47E+12	1.32E+12
CEC 02	**1.73E+01**	**1.55E-07**	7.80E+01	8.78E+01	1.52E+01	3.73E+03
CEC 03	**1.27E+01**	1.56E-06	1.37E+01	7.00E-04	**1.27E+01**	**9.03E-15**
CEC 04	7.62E+01	2.95E+01	3.44E+02	4.14E+02	**1.68E+01**	**8.20E+00**
CEC 05	2.01E+00	5.97E-01	2.56E+00	**3.25E-01**	**1.14E+00**	8.94E-02
CEC 06	**4.24E+00**	1.36E+00	9.90E+00	1.64E+00	9.31E+00	1.69E+00
CEC 07	2.65E+02	1.21E+02	5.79E+02	3.29E+02	**1.61E+02**	**1.04E+02**
CEC 08	**5.13E+00**	5.08E-01	6.87E+00	**5.01E-01**	5.22E+00	7.87E-01
CEC 09	2.40E+00	2.84E-02	6.05E+00	2.87E+00	**2.37E+00**	**1.84E-02**
CEC 10	**2.00E+01**	**1.46E-02**	2.12E+01	1.71E-01	2.03E+01	1.29E-01

5.5 Limitation

Some problems in the HABC algorithm need to be solved in the future

(i) In order to speed up the convergence speed, it is necessary to find out the redundant code and make corresponding improvements.

(ii) More verification is needed for HABC. The article is only tested in the continuity function. HABC can also be applied to logistics location and route optimization problems.

(iii) In the process of testing functions, some functions are still easy to fall into the local optimum, and it is necessary to find a better way to make HABC jump out of the local optimum.

6 Conclusion

This article aims to improve the shortcomings of the ABC algorithm, it is easy to be transferred into the trap of local optimization, and the optimization time is too long. Using the fast convergence characteristics of Harris Hawks optimization and Cauchy reverse learning to enhance the ability of global exploration, the artificial bee colony algorithm based on Harris Hawk (HABC) is proposed. HABC uses the three stages of the ABC algorithm to search for the solution space, introduces the HHO convergence formula before the onlooker bee phase, and explores potential food sources near the local optimum, so that the potential solutions can be fully utilized, and the accuracy of the algorithm convergence is accelerated. After the scout bee phase add to Cauchy reverse learning to generate a new food source to achieve the local optimum.

From the theoretical analysis, it is concluded that the improved algorithm improves the performance of the original ABC algorithm, and through 13 basic test functions and 10 test functions of CEC-C06 2019, 4 sets of experiments have been verified by simulation that the HABC algorithm has a higher convergence effect. The statistical results show that HABC has obvious advantages in solving quality.

At the time of the new crown epidemic, some scholars have talked about the application of deep learning to the detection of pneumonia [32-33]. In the future, the bee colony is also very promising in this field. At the same time, the HABC algorithm will be used in more practical optimization problems, such as route optimization and logistics location problems

References

[1] D. Karaboga, *An idea based on honey bee swarm for numerical optimization*, technical report-TR06, October, 2005.

[2] M. Dorigo, L. M. Gambardella, Ant colony system: a cooperative learning approach to the traveling salesman problem, *IEEE Transactions on Evolutionary Computation*, Vol. 1, No. 1, pp. 53-66, April, 1997.

[3] X. Li, *A new intelligent optimization method -artificial fish school algorithm*, Ph. D. Thesis, Zhejiang University, Zhejiang, China, 2003.

[4] D. Karaboga, B. Gorkemli, C. Ozturk, N. Karaboga. A comprehensive survey: artificial bee colony (ABC) algorithm and applications, *Artificial Intelligence Review*, Vol. 42, No. 1, pp. 21-57, June, 2014.

[5] S. Bharati, P. Podder, M. R. H. Mondal, Hybrid deep learning for detecting lung diseases from X-ray images, *Informatics in Medicine Unlocked*, Vol. 20, Article No. 100391, 2020.

[6] A. Khamparia, S. Bharati, P. Podder, D. Gupta, A. Khanna, T. K. Phung, D. N. H. Thanh, Diagnosis of breast cancer based on modern mammography using hybrid transfer learning, *Multidimensional Systems and Signal Processing*, Vol. 32, No. 2, pp. 747-765, April, 2021.

[7] Ş. Öztürk, R. Ahmad, N. Akhtar, Variants of Artificial Bee Colony algorithm and its applications in medical image processing, *Applied Soft Computing*, Vol. 97, Part A, Article No. 106799, December, 2020.

[8] H. Li, X. Li, L. Gao, A discrete artificial bee colony algorithm for the distributed heterogeneous no-wait flowshop scheduling problem, *Applied Soft Computing*, Vol. 100, Article No. 106946, March, 2021.

[9] Y. Wang, J. Liu, R. Li, X. Suo, E. Lu, Precipitation forecast of the Wujiang River Basin based on artificial bee colony algorithm and backpropagation neural network, *Alexandria Engineering Journal*, Vol. 59, No. 3, pp. 1473-1483, June, 2020.

[10] A. A. Heidari, S. Mirjalili, H. Faris, I. Aljarah, M. Mafarja, H. Chen, Harris hawks optimization: Algorithm and applications, *Future Generation Computer Systems*, Vol. 97, pp. 849-872, August, 2019.

[11] S. Jiao, G. Chong, C. Huang, H. Hu, M. Wang, A. A. Heidari, H. Chen, X. Zhao, Orthogonally adapted Harris hawks optimization for parameter estimation of photovoltaic models, *Energy*, Vol. 203, Article No. 117804, July, 2020.

[12] İ. Gölcük, F. B. Ozsoydan, Quantum particles-enhanced multiple Harris Hawks swarms for dynamic optimization problems, *Expert Systems with Applications*, Vol. 167, Article No. 114202, April, 2021.

[13] M. Abd Elaziz, A. A. Heidari, H. Fujita, H. Moayedi, A competitive chain-based Harris Hawks Optimizer for global optimization and multi-level image thresholding problems, *Applied Soft Computing*, Vol 95, Article No. 106347, October, 2020.

[14] E. Rodríguez-Esparza, L. A. Zanella-Calzada, D. Oliva, A. A. Heidari, D. Zaldivar, M. Pérez-Cisneros , L. K. Foong, An efficient Harris hawks-inspired image segmentation method, *Expert Systems with Applications*, Vol. 155, Article No. 113428, October, 2020.

[15] X. Chen, X. Wei, G. Yang, W. Du, Fireworks explosion based artificial bee colony for numerical optimization, *Knowledge-Based Systems*, Vol. 188, Article No. 105002, January, 2020.

[16] B. Rambabu, A. V. Reddy, S. Janakiraman, Hybrid Artificial Bee Colony and Monarchy Butterfly Optimization Algorithm (HABC-MBOA)-based cluster head selection for WSNs, *Journal of King Saud University-Computer and Information Sciences*, Available online, December, 2019.

[17] X. Chen, H. Tianfield, K. Li, Self-adaptive differential artificial bee colony algorithm for global optimization problems, *Swarm and Evolutionary Computation*, Vol. 45, pp. 70-91, March, 2019.

[18] N. Sharma, H. Sharma, A. Sharm, Beer froth artificial bee colony algorithm for job-shop scheduling problem, *Applied Soft Computing*, Vol. 68, pp. 507-524, July, 2018.

[19] S. Dai, G. Yang, M. Li, Z. Zhao, K. Yu, An Improved DABC Algorithm for Interference Waveform Decision-Making, *Electronics optics and control*, Vol. 28, No.7, pp. 53-56, July, 2021.

[20] Z. Zhang, X. Lu, Q. Sun, K. Wang, Improved artificial bee colony algorithm with enhanced exploitation ability, *Journal of Computer Applications*, Vol. 39, No. 4, pp. 949-955, April, 2019.

[21] A. Tang, T. Han, D. Xu, L. Xie, Chaotic elite Harris hawks optimization algorithm, *Journal of Computer Applications*, Vol. 41, No. 8, pp. 2265-2272, August, 2021.

[22] Y. Ma, Z. Shi, K. Zhao, C. Gong, L. Shan, TDOA localization based on improved Harris hawk optimization algorithm, *Computer Engineering*, Vol. 46, No. 12, pp. 179-184, December, 2020.

[23] T. Zhao, J. Fu, Y. Wang, B. Zhang, An underwater measurement and control network centralized data fusion localization algorithm based on Chan-algorithm method, *The Journal of the Acoustical Society of America*, Vol. 142, No. 4, Article No. 2732, October, 2017.

[24] M. Abd Elaziz, D. Yousri, S. Mirjalil, A hybrid Harris hawks-moth-flame optimization algorithm including fractional-order chaos maps and evolutionary

population dynamics, *Advances in Engineering Software*, Vol. 154, Article No. 102973, April, 2021.

[25] A. A. Ewees, M. Abd Elaziz, Performance analysis of Chaotic Multi-Verse Harris Hawks Optimization: A case study on solving engineering problems, *Engineering Applications of Artificial Intelligence*, Vol. 88, Article No. 103370, February, 2020.

[26] X. Chen, X. Wei, G. Yang, W. Du, Fireworks explosion based artificial bee colony for numerical optimization, *Knowledge-Based Systems*, Vol 188, 105002, January 2020.

[27] H. R. Tizhoosh, Opposition-based learning: A new scheme for machine intelligence, *International conference on computational intelligence for modelling, control and automation and international conference on intelligent agents, web technologies and internet commerce (CIMCA-IAWTIC'06)*, Vienna, Austria, 2005, pp. 695-701.

[28] L. Kang, W. Dong, J. Tian, Opposition-based Particle Swarm Optimization with Adaptive Cauchy Mutation, *Computer Science*, Vol. 42, No. 10, pp. 226-231, October, 2015.

[29] J. Guo, C. Ma, M. Miao, S. Zhang, Markov chain based artificial bee colony algorithm, *Journal of Beijing University of Posts and Telecommunications*, Vol. 43, No. 1, pp. 54-60, February, 2020.

[30] C. M. Rahman, T. A. Rashid, A new evolutionary algorithm: Learner performance based behavior algorithm, *Egyptian Informatics Journal*, Vol. 22, No. 2, pp. 213-223, July, 2021.

[31] J. M. Abdullah, T. Ahmed, Fitness Dependent Optimizer: Inspired by the Bee Swarming Reproductive Process, *IEEE Access*, Vol. 7, pp. 43473-43486, March, 2019.

[32] S. Bharati, P. Podder, M. R. H. Mondal, V. B. S. Prasath, CO-ResNet: Optimized ResNet model for COVID-19 diagnosis from X-ray images, *International Journal of Hybrid Intelligent Systems*, Vol. 17, No. 1-2, pp. 71-85, July, 2021.

[33] S. Bharati, P. Podder, M. R. H. Mondal, V. B. S. Prasath, Medical Imaging with Deep Learning for COVID- 19 Diagnosis: A Comprehensive Review, *International Journal of Computer Information Systems and Industrial Management Applications*, Vol 13, pp. 91-112, July, 2021.

Biographies

Liyi Zhang is a professor and PhD candidate supervisor in Tianjin University of Commerce, China. His research interests include Signal detection and information processing.

Zuochen Ren received the B.S. degree from the Xi’an University of Finance and Economics, Xi'an, China, in 2020. He is currently pursuing the M.S. degree in Tianjin university of commerce, Tianjin, China. He current research interests include swarm intelligence and evolutionary computation.

Ting Liu received the doctor from the Tianjin University, Tianjin, China, in 2013. She is an associate professor with the Tianjin University of Commerce, Information Engineering College. Her main research directions interests include intelligent signal and information processing.

Jinyan Tang received the B.S. degree from the Tianjin University of Commerce, Tianjin, China, in 2020. She is currently pursuing the M.S. degree in Tianjin University of Commerce, Tianjin, China. Her current research interests include swarm intelligence and evolutionary computation.