International Journal of Control and Automation
Vol. 8, No. 2 (2015), pp. 311-322
http://dx.doi.org/10.14257/ijca.2015.8.2.30

Investigation of Modified Bee Colony Algorithm with Particle and Chaos Theory

Guo Cheng

Shangluo College, Zhangye, Gansu, China
chengguo3805@163.com

Abstract

Foraging behavior of animal widely concerns researchers. Some swarm intelligence algorithms, such as ant colony optimization algorithm, particle swarm optimization algorithm, artificial fish swarm algorithm, and so on, have been developed. Artificial bee colony algorithm (ABC), which is based on self-organization model, has been proposed. Its application is mainly used in the field of numerical optimization. Researchers verify the outstanding performance in function optimization domain according to the comparison with other algorithms with various improvements. Artificial bee colony algorithm itself has better performance in solving high dimension function. It needs not large population size and can guarantee the global convergence. In the paper, from the view of improving the convergence rate of the algorithm, search operators have been studied and a faster algorithm has been proposed. At the same time, the search region has been optimized. According to the example verification, the new algorithm is effective and the algorithm can be used in the optimization field.

Keywords: *modified algorithm, artificial bee colony, particle, chaos*

1. Introduction

In recent years, research on the foraging behavior of animal is widely concerned by researchers. Some swarm intelligence algorithms, such as ant colony optimization algorithm, particle swarm optimization algorithm, artificial fish swarm algorithm, *et al.*, have been developed. In 2005, artificial bee colony algorithm (Artificial Bee Colony, ABC) [1-3], which is based on self-organization model [4-5], has been proposed. Its application is mainly used in the field of numerical optimization [6-13]. Thus, domestic and foreign researchers verify the outstanding performance in function optimization domain according to the comparison to other algorithms with a variety improvement of the algorithm. Present research on ABC algorithm is still in the exploring stage [14-16].

As with other intelligent algorithms, many improvements have been proposed for the algorithm. For the initialization, orthogonal experimental design [17-19] and chaotic search [20-21] initialization has been introduced to make the initial population distribution better; For the nectar source selection by the following bee, strategies, such as Boltzmann mechanism [22], championship [23], ordering strategy [24], splitting strategy [25], anti-roulette strategy, pheromone sensitivity strategy [26], *et al.*, has been developed; For improving the search operators, there are sharing factor with dynamic change method, speed updating formula derived from particle swarm algorithm (PSO), quantum coordinate model, and addition of other search modes: crawling process in monkey algorithm, differential evolution, and so on; In the aspect of swarm variety, particle swarm algorithms are often used to establish the dual population model, and some interactions are often designed. Due to the overall good performance of artificial bee colony algorithm for individual updating and difficult in falling into local optimum, it often introduced as an searching operator into other algorithms, such as the ant colony

ISSN: 2005-4297 IJCA

algorithm, particle swarm, firefly algorithm, and so on, to improve the convergence speed and solution accuracy.

Keep balance between global exploration and local exploitation is the key to keep the better performance of swarm intelligence algorithm. In the standard artificial bee colony algorithm (ABC algorithm), randomly selected neighbor strategy is adopted in the searching nectar source position for hired bee and follow bee. This would lead to weak local development ability although good global exploratory ability. Particle swarm optimization is used to make the hire bee get the global optimum guidance in exploring new source location. This can improve the performance of the algorithm and decrease the amount of calculation.

Artificial bee colony algorithm itself has better results in solving high dimension function. It needs not large population size and can guarantee the global convergence. However, solving speed is quite slow, and the number of the generations would be larger when the dimensions are higher.

In this paper, from the view of improving the convergence rate of the algorithm, search operators have been studied and a faster algorithm has been proposed. At the same time, the search region has been optimized. The main contribution is the proposition of the new algorithm to increase the convergence speed. The remainder of the paper is shown as the following: Standard ABC algorithm is listed in section 2. Modified ABC algorithm is shown in section 3. Adaptive search space introduction is shown in section 4. The verification is shown in section 5 and the conclusion is described in section 6.

2. Standard ABC Algorithm

In standard ABC algorithm, the bee swarm is composed of leading bee, follow bee and scout bee. The number of food sources is the same as leading bees and scout bees. The basic processes are as follows:

Step1: Initialize the leading bees random, and one leading bee is set for a food source, and calculates the concentration of the food according to the objective function. Then, optimal location and the optimal fitness would be recorded.

Step2: Each lead bee will be preceded as follows: randomly choose to a neighbor of leading bee, and randomly select one dimension. Location is updated according to the formula (1). where $k \in \{1,2,...,Num\}$, but $k \neq 1\}$, and Num is the number of leading bees; $J \in \{1,2,...,\mathrm{dim}\}$, and Dim is the dimension space; $Rand$ is a random number and $Rand \in [0,1]$; If fitness of the new position is better, the new location would be updated as the current location. Or else, the non-renewable counting number would plus 1.

$$newx_{ij} = x_{ij} + (2 * rand - 1) * (x_{ij} - x_{kj}) \tag{1}$$

Step3: Probability of every lead being selected would be calculated according to formula 2. Fit is the fitness of leading bee i.

$$P_i = \frac{fitness_i}{\sum_{j=1}^{SN/2} fitness_j} \tag{2}$$

Step4: Each follow bee would be proceeding as follows: a leading bee would select in accordance with the roulette wheel strategy and the position would be updated according to equation (1). If the new position is better, the selected bee would be updated in the current position, otherwise, the number $Basi$ would plus 1. A leading bee can be selected by many follow bees repeatedly, which means that leading bee with greater fitness degree would be selected with bigger probability.

Step5: The location and concentration of optimal food sources of this generation would be recorded.

Step6: Leading bee with the maximum number *Bas* would be selected, and the leading bee would be as a scout bee if the number is bigger than *Limit* . Then, position, fitness and *Bas* would be initialized. The parameter Limit plays a role of reborn of lead bee with the long-term with no updating.

Step7: If the generation number is smaller than the maximum number, it should be go to step 2 to the next generation, otherwise, output the results.

The flow chart of ABC is shown in Figure 1.

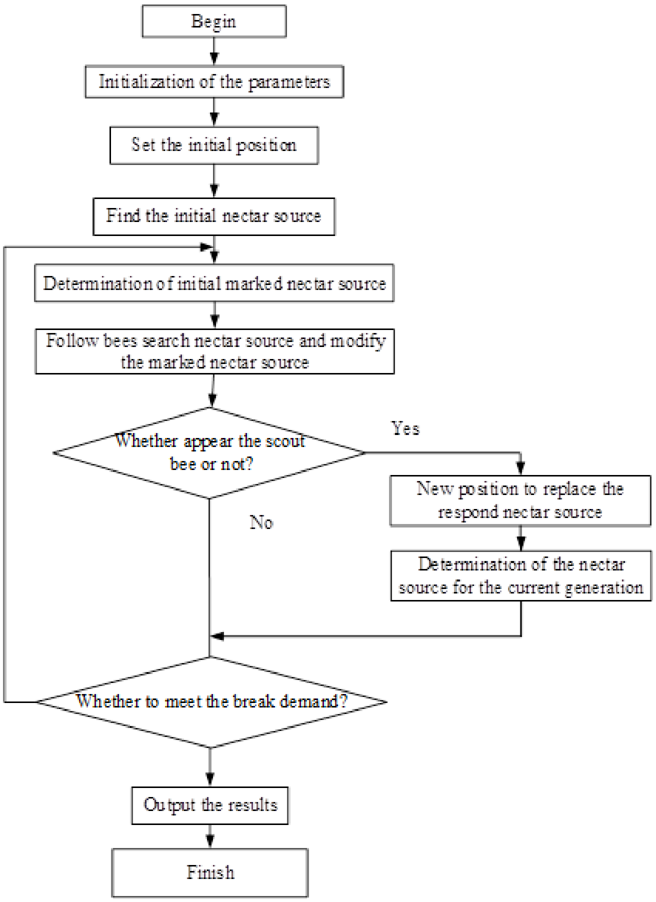
Begin
Initialization of the parameters
Set the initial position
Find the initial nectar source
Determination of initial marked nectar source
Follow bees search nectar source and modify the marked nectar source
Whether appear the scout bee or not?
Yes
No
New position to replace the respond nectar source
Determination of the nectar source for the current generation
Whether to meet the break demand?
Output the results
Finish

Figure 1. Flow Chart of ABC

In the bee colony foraging behavior, population is divided into three categories of hire bee, follow bee, and scout bee, respectively. They have the primary exploration, re-development and avoid stagnation effect. From the algorithm structure, bee colony algorithm and other intelligent optimization algorithm are quite different, and there is more room for improvement. For example, in genetic algorithm and particle swarm algorithm, the group has no division of labor. That is to say that in each generation all individuals are selected according to the probability for a certain operation. The probability may change with the iteration. The bees are divided into two main groups (hire bee and follow bee) and form a two stage operations. Appropriate adjustments are used to maintain the population diversity with scout bees.

The hire bee and the follow bee would update the location according to the formula 1. Nectar source selected by follow bee is according to the probability calculated by Equation 2, and then roulette wheel is adopted to choose. The nectar source with higher fitness is chosen with bigger probability, which has been developed for a second time. The search is still in accordance with equation 1. Bee colony algorithm with single dimension search and greedy choice may lead to strong global exploration ability and weak local exploitation ability. Then it should be improved.

3. Modified ABC Algorithm

In the particle swarm optimization (PSO), the particle updates the position according to the following equation:

$$v_{ij}(t+1) = \omega v_{ij}(t) + c_1 r_1(t)(p_{ij}(T) - x_{ij}(t)) + c_2 r_2(t)(p_{gj}(t) - x_{ij}(t)) \tag{3}$$

$$x_{ij}(t+1) = x_{ij}(t) + v_{ij}(t+1) \tag{4}$$

Where, $x_i = (x_{i1}, x_{i2}, \ldots, x_{iD})^T$ is the position of particle i ;

$V_i = (v_{i1}, v_{i2}, \ldots, vid, \ldots, x_{iD})^T$ is the velocity;

$p_i = (p_{i1}, p_{i2}, \ldots, p_{iD})^T$ is the individual extreme value;

$p_g = (p_{g1}, p_{g2}, \ldots, p_{gD})^T$ is the global extreme value;

t is the current generation number; r_1, r_2 is the random number belong to $[0,1]$;

ω is the inertia weight, and c_1, c_2 are the acceleration coefficient.

PSO algorithm makes good use of the prior knowledge, and it has higher performance in local searching. In the standard ABC algorithm, leading bee and follow bee would take strategy of randomly selected neighbor in updating position. So, if introduce the global optimal solution and set one bulletin board to show the global optimal location with only visible to the leading bee. The leading be would update the position according to the formula (5), and other variables will update the position according to equation (1).

$$newx_{ij} - (1 - rand) * x_{ij} + rand * GBest_{1j} \tag{5}$$

Where, $GBest$ is the optimal global position.

For the standard ABC, existing study adds the cross reaction with optimum global position based on the equation (1), and studies the coefficient of the global components. Making full use of the PSO algorithm, individual extremum and asynchronous learning factor have been defined in the algorithm, and drop the parameter. This method is more difficult to resolve and decreases the solution efficiency. So, in the paper, we proposed another method: selecting the current food source instead of selecting a random neighbor.

4. Adaptive Search Space Introduction

4.1 Dynamic Adjustments of the Search Space

Set the spatial solution is N , and each solution is a vector with D dimension.

Initial state, we will generate the solution according to equation (6):

$$v_{ij} = x_{ij}(x_{ij} - x_{kj}) \tag{6}$$

And the solution is:

$$\begin{aligned} X_1 &= (x_{11}, x_{12}, \ldots x_{1d}) \\ X_2 &= (x_{21}, x_{22}, \ldots x_{2d}) \\ &\ldots\ldots \\ X_n &= (x_{n1}, x_{n2}, \ldots x_{nd}) \end{aligned} \tag{7}$$

Then,

$$Y = (y_1, y_2, ..., y_i, ...x_d) = (max(|x_{11}|, |x_{21}|, \ldots, |x_{n1}|), max(|x_{12}|, |x_{22}|, \ldots, |x_{n2}|), \ldots, max(|x_{13}|, |x_{23}|, \ldots, |x_{ni}|), \ldots, max(|x_{1d}|, |x_{2d}|, \ldots, |x_{nd}|)) \quad (8)$$

Where, component y_i means the distribution of populations in the dimension of i. The bigger y_i means greater distribution of population in dimension coordinates of i. When the search reach a certain number of iterations, maximum possible change interval of Y would be used to generate the population, and then calculate the fitness value. The fitness values would be evaluated and then continue to search after the evaluation. After the iteration, the interval of generating the initial population would be gradually shrunk, and the iterative process will be speed up to improve the efficiency of the whole algorithm.

In accordance with the method described before, the search space is contractible, then two problems would be appeared: a) optimal solution may be excluded from the reduction of the search space, and the optimal solution of such problems cannot be searched; b) the motion range of the individual position is greatly reduced, and the local optimal breaking ability of the algorithm is decreased. If most of the individuals are in motion near the local extremum of the same algorithm, the solving process would be prone to temporary stagnation. Then, breaking the limitation of the local extremum may need for a long period of time, and may still not be able to break through this limitation and falling into local optimum. Therefore, we should solve the problems according so some method. In the paper, we will mainly uses two methods in the solving process: a) the population will be divided into two parts, one part is used for the dynamic adjustment of the search area to accelerate the convergence speed, while another part is still in the original space. The solution at the space edge will not be ignored, simulation results show the feasibility of the method; b) when the search space is adjusted, next compression would not be processed immediately. But in each time after compression, iteration would be processed for several times to make the group to have a process of adaptation to a new environment. Then, the next time compression would be given.

4.2 Chaotic Search

Chaos is a kind of nonlinear phenomena and it is widely exist in nature. It seems confusing but with exquisite structure, and has the characteristics of randomness, ergodicity and regularity. It can traversal all the state with no repeatability in a certain range according to their own laws. The general chaos refers to the state of motion which is obtained by the deterministic randomness equations. Variables with chaotic state are called chaotic variables. For example, the logistic map is a typical chaotic system with the following equation:

$$z_{n+1} = \mu z_n + (1 - z_n); \qquad n = 0,1,2,\ldots \quad (9)$$

Where, μ is the control parameters, and the equation (5) can be regarded as a dynamic system. When μ is determined by the initial value, any value of $z_0 \in [0,1]$ can be calculated for a fixed period of time sequences $z_0, z_1, z_2, \cdots$, when $\mu = 4$, the system is in complete chaos.

Due to the ergodicity of chaos, the optimization algorithm with chaos is easy to jump out of local optimal solution. As a good search mechanism, many studies, which combined the chaos and swarm intelligence algorithms, have been published. For example, chaotic genetic algorithm can be obtained with the chaos operator used in the genetic algorithm (GA) and chaotic particle swarm optimization algorithm can be obtained with the combination of chaotic theory and particle swarm optimization (PSO). Both the methods have better search performance.

In the ABC algorithm, if a solution is still not improved through *limit* cycles, it means that the solution is in the local optimum, and then a new solution will randomly generate to replace it. In the paper, we will make use of the chaotic search to assign the solution to jump out of the local optimum.

The main idea is the use of the ergodicity of chaos to generate chaotic sequence based on current search stagnation solution. The optimal position in chaotic sequences generated would be used to replace the original position. Stagnation solutions by this treatment will make the search evolve continuously to improve the convergence speed and accuracy. This paper assumes that search stagnation solution is:

$$X_k = (x_{k1}, x_{k2}, \ldots, x_{kd}), \qquad x_{ki} \in [a_i, b_i] \tag{10}$$

And it will be processed with chaos operator. According to equation (9). The main steps are described as the following:

(1) Mapping X_k to Logistic equation domain [0, 1]:

$$Z_{ki}^{0} = \frac{x_{ki} - a_i}{b_i - a_i}; \quad k = 1,2,\ldots,n; \qquad i = 1,2,\ldots,d \tag{11}$$

(2) Logistic equation was used to generate chaotic variable iteration sequence:

$$Z_k^m(m = 1,2,\ldots,C_{\max})$$

Where, $C_{\max}$ is the maximum number of iterations in chaos search.

(3) The chaotic sequence $Z_{ki}^m(m = 1,2,\ldots,C_{\max})$ will be processed by inverse mapping $x_{ki} = a_i + (b_i - a_i)z_{ki}^m$ to the original solution space. That is to say that we acquire the $X_k^{'} = (x_{k1}^{'}, x_{k2}^{'}, \ldots, x_{kd}^{'})$. The fitness value $f' = f(X_k')$ calculated would be compared with the solution of the original, and the best solution would be retained.

(4) If maximum iterative algebraic has been reached, the optimization process would be finished, or else, return to the step (2).

Then, selection strategies should be determined. In the algorithm, tournament method would be used in the selection strategy for bee to search food source. Because the tournament selection only adapt relative value as the selection standard and has no requirement to positive and negative fitness, the algorithm can avoid premature convergence and stagnation phenomenon to a certain extent.

The flow chart of the algorithm is shown in the following:

(1) Initialize the swarm solution $x_i(i = 1,\ldots,n)$;

(2) Calculate the fitness value of each solution x_i ;

(3) Determination of whether to adjust the search space. If meet the adjusting demands, the leading bee would generate the new solution space with equation $v_i = 2y_i \times rand(0,1) - y_i$, or else it should generate the new solution with $v_{ij} = x_{ij} + r_{ij}(x_{ij} - x_{kj})$. Fitness values of the new solution should be calculated.

(4) If the fitness value of v_i is better than x_i , v_i would be used to replace the x_i , or else, obtain the x_i .

(5) Probability P_i for x_i should be calculated with equation $P_i = fit_i / \sum_{i=1}^{SN} fit_i$ and the tournament method. Where, is the fitness value of solution and is the number of solution.

(6) The follow bee would select food source (solution) according to P_i , and generate new solution v_i , then calculate the fitness value.

(7) If the fitness value is better than, will be replaced by or keep constant.

(8) Determine whether to give up one solution. If it exists, a new solution would be generated by chaos search to replace it.

(9) Record the current optimal solution.

(10) If it meets the termination condition, the optimal solution would be output, or else go to step (3).

5. Verification

T In order to test the new algorithm, 9 standard test functions, all of which are widely used in the field of function optimization, have been tested. Definition and test function space are given and their theoretical optimal values are 0. All the results calculated by the new algorithm are compared with ABC algorithm and the improved ABCP algorithm.

Specific algorithm parameters are set with the follows: All of numbers of leading bees follow bees and food sources are 50; test functions are with dimensions number of 50; the limit is 10; each function will iterate 1000 times. In order to test the algorithm performance, for each test function, the algorithms will run 50 times. Parameters of optimal solution, the worst solution, average value, variance, mean running time is selected to examine the algorithm performance.

(1) Continuous type function with single mode

Function $f_1(x) = \sum_{i=1}^{D} x_i^2 \qquad [-100,100]^D$

Table 1. Comparison of the Algorithms

Algorithm	Mean value	Optimal value	Worst value	Variance	Operation time (equivalent value)
ABC	5.051×10-5	7.89×10-6	1.66×10-4	2.34×10-4	1
ABCP	1×10-173	2.72×10-180	3.15×10-170	0	3.8
New algorithm	0	0	0	0	0.025

Function $f_2(x) = \sum_{i=1}^{D}\left(\sum_{i=1}^{D} x_j\right)^2 \qquad [-100,100]^D$

Table 2. Comparison of the Algorithms

Algorithm	Mean value	Optimal value	Worst value	Variance	Operation time (equivalent value)
ABC	7.95×10^{-004}	1.83×10^{-004}	0.002 1	4.33×10^{-019}	1
ABCP	1.00×10^{-172}	1.08×10^{-175}	1.29×10^{-168}	0	0.5
New algorithm	0	0	0	0	0.03

Function $f_3(x) = \sum_{i=1}^{D} \left|x_i\right| + \left(\prod_{j=1}^{D} \left|x_j\right|\right)$ $[-100,100]^D$

Table 3. Comparison of the Algorithms

Algorithm	Mean value	Optimal value	Worst value	Variance	Operation time (equivalent value)
ABC	4.72×10^{-218}	1.36×10^{-240}	1.33×10^{-216}	0	1
ABCP	2.74×10^{-094}	4.05×10^{-099}	1.91×10^{-089}	0	3.02
New algorithm	4.23×10^{-217}	1.26×10^{-236}	3.85×10^{-277}	3.71×10^{-217}	0.05

Function $f_4(x) = \max_i \{x_i\}$ $[-100,100]^D$

Table 4. Comparison of the Algorithms

Algorithm	Mean value	Optimal value	Worst value	Variance	Operation time (equivalent value)
ABC	0.4	0	0	76.91	1
ABCP	7.73×10^{-071}	5.03×10-72	2.70×10^{-068}	0	0.8
New algorithm	6.25×10^{-197}	1.51×10^{-217}	1.34×10^{-197}	0	0.013

(2) Non-continuous type function with single mode
This function is mainly used to test the search precision and execution performance.

The function is:

$$f_5(x) = \sum_{i=1}^{D} \left[x_i + 0.5\right]^2 \qquad [-100,100]^D$$

Table 5. Comparison of the Algorithms

Algorithm	Mean value	Optimal value	Worst value	Variance	Operation time (equivalent value)
ABC	0.4	0	2	4.69	1
ABCP	0	0	0	0	10.23
New algorithm	0	0	0	0	0.1

(3) The noise function

$$f_6(x) = \sum_{i=1}^{D} i x_i^4 rand[0,1) \, [x_i + 0.5] \qquad [-128,128]^D$$

Table 6. Comparison of the Algorithms

Algorithm	Mean value	Optimal value	Worst value	Variance	Operation time (equivalent value)
ABC	0.1626	0.1010	0.2274	0.1454	1
ABCP	1.83×10^{-016}	9.26×10^{-016}	1.62×10^{-12}	0	15.02
New algorithm	0	0	0	0	1

(4) Multi modal function

This type of function has more than one local extreme value, and their global extreme values are often difficult to search. These functions can be used to test global search performance and premature avoid convergence of the algorithm.

Function $f_7(x) = \sum_{i=1}^{D} x_i^2 - 10\cos(2\pi x_i) + 10]$ $[-5.12,5.12]^D$

Table 7. Comparison of the Algorithms

Algorithm	Mean value	Optimal value	Worst value	Variance	Operation time (equivalent value)
ABC	0.0950	0.0564	0.2034	0.182	1
ABCP	0	0	0	0	2.5
New algorithm	0	0	0	0	0.0104

Function $f_8(x) = 0.5 + \dfrac{\sin(\sqrt{\sum_{i=1}^{D} x^{i}}^{2} - 0.5)}{(1 + 0.001\sum_{i=1}^{D} x_i)^2}$ $[-5.12,5.12]^D$

Table 8. Comparison of the Algorithms

Algorithm	Mean value	Optimal value	Worst value	Variance	Operation time (equivalent value)
ABC	0.2182	0.1526	0.2947	0.2526	1
ABCP	0.009 715 9	0. 009 715 9	0.009 715 9	0	38.72
New algorithm	0	0	0	0	0.03

Function $f_9(x) = \dfrac{1}{4000}\sum_{i=1}^{D} x_i^2 + \prod_{i=1}^{D}\cos\left(\dfrac{x_i}{\sqrt{i}}\right) + 1$ $[-600,600]^D$

Table 9. Comparison of the Algorithms

Algorithm	Mean value	Optimal value	Worst value	Variance	Operation time (equivalent value)
ABC	0.0015	4.77×10^{-27}	0.0123	0.0236	1
ABCP	0.679 6	0.437 12	0.88277	0.1903	60.5
New algorithm	0.213	4.77×10^{-12}	0.2376	0.2945	0.5

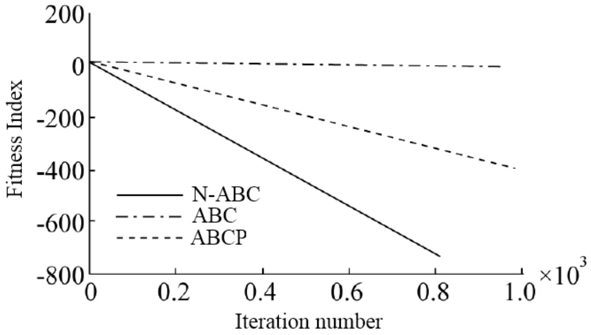

Figure 2. Comparison for Function 1

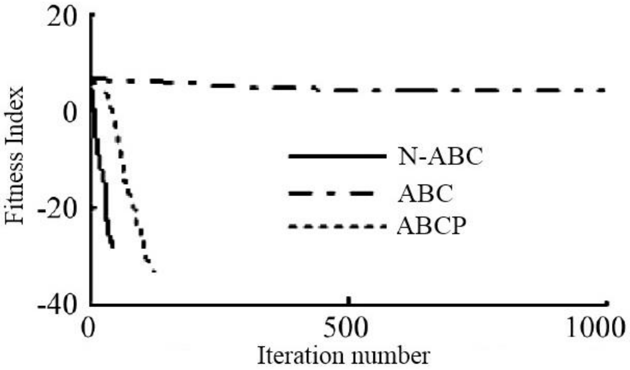

Figure 2. Comparison for Function 3

From the Table 1-7 and Figure 2 and Figure 3, we can see the new algorithm has a better performance. It has high accuracy of the optimal solution and less operation times.

6. Conclusion

Swarm intelligence algorithms, such as ant colony optimization algorithm, particle swarm optimization algorithm, artificial fish swarm algorithm, have been developed. Artificial bee colony algorithm (ABC), which is based on self-organization model, has been proposed. It has better performance in solving high dimension function and need not large population size to get better convergence performance.

In the paper, from the view of improving the convergence rate of the algorithm, search operators have been studied and a faster algorithm has been proposed. At the same time, the search region has been optimized. In the verification, the accuracy of the optimal solution found by the new algorithm is obviously improved and with less iteration times. At the same time, the operation time is shorter. The new algorithm improved the search ability and reduces the search time. It can be used in the optimization of function.

Acknowledgment

This paper supported by the project of the Scientific research project in shaanxi province department of education(No.2013Jk0597)" Image sparse representation method and application research ",and is supported by Scientific research foundation of Shangluo University(No.12SKY010)" Feature extraction method based on the maximum margin criterion ".

References

[1]. T. D. Seeley, "The wisdom of the hive the social physiology of honey bee colonies", Cambridge, Massachusetts: Harvard University Press, **(1995).**

[2]. X. Zhengguang, X. Jun and W. Yanfei, "Representative artificial bee colony algorithms", A survey LISS 2012 - Proceedings of 2nd International Conference on Logistics, Informatics and Service Science, **(2013),** pp. 1419-1424.

[3]. R. Amr, A. A. Youssif and S. Eldin, "Introducing Adaptive Artificial Bee Colony algorithm and using it in solving traveling salesman problem", Proceedings of 2013 Science and Information Conference, SAI, **(2013),** pp. 502-506.

[4]. S. Sumedha, C. Abhishek and C. Manoj, "Self-organization architecture and model for wireless sensor networks", Proceedings International Conference on Electronic Systems, Signal Processing, and Computing Technologies, ICESC, **(2014),** pp. 204-208.

[5]. I. Takeshi, "Self-organization model for the energy cluster formation with distributed energy network", IEEE Symposium on Computational Intelligence Applications in Smart Grid, CIASG, **(2013),** pp. 161-166.

[6]. W. Yuyong, Y. Jianqiao and Y. Yongdou, "Parameter optimization of support vector machine based on artificial bee colony algorithm", Journal of Computational Information Systems, vol. 10, no. 1, **(2014),** pp. 395-401.

[7]. H. Shayeghi and A. Ghasemi, "A modified artificial bee colony based on chaos theory for solving non-convex emission/economic dispatch", Energy Conversion and Management, vol. 79, **(2014),** pp. 344-354.

[8]. Y. Zhen, Z. Ya, Z. W. Lan and Z. Li, "Extensive particle swarm artificial bee colony algorithm for function optimization", Applied Mechanics and Materials, vol. 496-500, **(2014),** pp. 1808-1811.

[9]. L. Jun-Qing, P. Quan-Ke and T. M. Fatih, "A discrete artificial bee colony algorithm for the multi-objective flexible job-shop scheduling problem with maintenance activities", Applied Mathematical Modelling, vol. 38, no. 3, **(2014),** pp. 1111-1132.

[10]. W. Zhaowei, L. Xiaoxiang and Z. Jiajie, "Performance evaluation in color-based image retrieval using artificial bee colony algorithm", Journal of Information and Computational Science, vol. 11, no. 4, **(2014),** pp. 1077-1086.

[11]. G. Wei-Feng, L. San-Yang and H. Ling-Ling, "Enhancing artificial bee colony algorithm using more information-based search equations", Information Sciences, vol. 270, **(2014),** pp. 112-133.

[12]. L. Jian-Sha, W. Yao-Wei, L. Xiu-Lin, T. Hong-Tao and D. Qiao-Ying, "Application of hybrid artificial bee colony algorithm in mixed assembly lines sequencing", Computer Integrated Manufacturing Systems, CIMS, vol. 20, no. 1, **(2014),** pp. 121-127.

[13]. Z. Jianzhong, L. Xiang, O. Shuo, Z. Rui and Z. Yongchuan, "Multi-objective artificial bee colony algorithm for short-term scheduling of hydrothermal system", International Journal of Electrical Power and Energy Systems, vol. 55, **(2014),** pp. 542-553.

[14]. A. M. Shafiul, U. K. M. Wasi and I. M. Monirul, "Self-adaptation of mutation step size in artificial bee colony algorithm for continuous function optimization", Proceedings of 2010 13th International Conference on Computer and Information Technology, ICCIT, **(2010),** pp. 69-74.

[15]. Z. Yanyu, Z. Peng, W. Yang, Z. Baohui and K. Fangjun, "Linear weighted gbest-guided artificial bee colony algorithm", Proceedings 2012 5th International Symposium on Computational Intelligence and Design, ISCID, vol. 2, **(2012),** pp. 155-159.

[16]. T. Milan, B. Nebojsa and S. Nadezda, "Adjusted artificial bee colony (ABC) algorithm for engineering problems", WSEAS Transactions on Computers, vol. 11, no. 4, **(2012),** pp. 111-120.

[17]. S. Qiang, "Effects of apparatus parameters on MFL signals using orthogonal experimental design", Applied Mechanics and Materials, vol. 44-47, **(2011),** pp. 3524-3528.

[18]. J. Lijun, S. Yunfeng, L. Hongfei, S. Xiaoli, Z. Wei and Z. Aiping, "Application of orthogonal experimental design in synthesis of mesoporous bioactive glass", Microporous and Mesoporous Materials, vol. 184, **(2014),** pp. 122-126.

[19]. W. Guangming, M. Xiandong, H. Tianjiang and Z. Daibing, “Experimental and analytical study on factors influencing biomimetic undulating fin propulsion performance based on orthogonal experimental design”, Advanced Robotics, vol. 27, no. 8, **(2013),** pp. 597-609.

[20]. L. Lin, H. Jianming and S. Boan, “A new hybrid method of genetic algorithm, Tabu and Chaotic search for CVRPTW”, Proceedings 2010 IEEE International Conference on Intelligent Computing and Intelligent Systems, ICIS, vol. 2, **(2010),** pp. 336-340.

[21]. Z. Ping, W. Ping, Y. Hong-Yang and F. Chun, “Biogeography-based optimization algorithm by using chaotic search”, Journal of the University of Electronic Science and Technology of China, vol. 41, no. 1, **(2012),** pp. 65-69.

[22]. Y. Kentaro, Y. Takayuki, Y. Masato, M. Toshiro, I. Kazuhiro and N. Shinji, “A level set-based topology optimization using the lattice-boltzmann method”, Nihon Kikai Gakkai Ronbunshu, C Hen/Transactions of the Japan Society of Mechanical Engineers, Part C, vol. 79, no. 802, **(2013),** pp. 2152-2163.

[23]. K. A. Husseinzadeh and K. Behrooz, “A new algorithm for constrained optimization inspired by the sport league championships”, 2010 IEEE World Congress on Computational Intelligence, WCCI 2010 - 2010 IEEE Congress on Evolutionary Computation, CEC, **(2010).**

[24]. Q. Lian and S. Lingjia, “Research on two-stage supply chain ordering strategy optimization based on system dynamics”, LISS 2012 - Proceedings of 2nd International Conference on Logistics, Informatics and Service Science, **(2013),** pp. 345-354.

[25]. L. G. Ming, S. Wei, Q. X. Dong, Z. Y. Hao and Z. Q. Jie, “The optimal partitioning strategy for online verification based on GN splitting algorithm”, Applied Mechanics and Materials, vol. 373-375, **(2013),** pp. 1503-1508.

[26]. F. Stefka, A. Krassimir and M. Pencho, “Intuitionistic fuzzy estimation of the ant colony optimization starting points”, Lecture Notes in Computer Science, 7116 LNCS, **(2012),** pp. 222-229.

Authors

Guo Cheng, 1983-4, He born in Zhangye, Gansu. He is a Master of Science. Now, he is a lecturer in Shangluo University, and his research directions are r computational intelligence and pattern recognition.