Journal of Information Hiding and Multimedia Signal Processing ©2016 ISSN 2073-4212
Ubiquitous International Volume **7**, Number **3**, May **2016**

Artificial Bee Colony Based on Special Central and Adapt Number of Dimensions Learning

Jia Zhao, Li Lv, Hui Wang, De-Kun Tan, Jun Ye, Hui Sun and Yin-Tai Hu
School of Information Engineering
Nanchang Institute of Technology
Nanchang 330099, China
zhaojia925@163.com; lvli623@163.com; huiwang@whu.edu.cn;
tdk2001219331@21cn.com; 1208561815@qq.com; sunhui@nit.edu.cn; 269016084@qq.com

Received August, 2015; revised February, 2016

Abstract. *There is lack of information sharing ability among nectar sources in standard Artificial Bee Colony (ABC), to improve it, we define the central location as the special central location, build special central strategy and propose a new approach called Artificial Bee Colony based on special central and adapt number of dimensions learning. In the processing of evolution, the optimal bee greedily chooses the optimal location as the new optimal location on the initial and special central location; then the employer bee and the onlooker bee choose some dimension of one bee to learn adaptively according the fitness value, which can accelerate the optimization efficiency, and new approach adopts a new searching strategy by learning from global optimal nectar source and nectar source in a neighborhood, it can enlarge the optimization scope. Experiments are conducted on 9 famous benchmark functions by our method and other ABC variants; the results demonstrate that our method has better performance than other ABC variants on precision, convergence velocity and stability.*
Keywords: Artificial bee colony, Special central, Adaptive, Learning, Number of dimensions.

1. **Introduction.** Artificial bee colony (ABC), originally developed by Dervis Karaboga in 2005 [1] is a swarm intelligence optimization algorithm which is proposed to simulate the behavior of honey bee. Because ABC has many advantages of a few parameters easy implementation and strong global search optimum, many researchers pay more attention to ABC and apply it into lots of science projects successfully, such as Neural networks training [2], clustering analysis [3], constrained optimization [4], spectrum sensing [5], wireless sensor networks [6], passive continuous authentication system [7], cloud computing [8] and mechanical design [9] and so on. ABC is similar to other swarm intelligence algorithms; it is also easy to fall into local optimum, to be premature, and to be slow convergence. Recently, there are many ABC variants. Pei-Wei Tsai [10] proposed Enhanced Artificial Bee Colony Optimization (EABC), the fast non-dominated sorting and population selection strategy are applied to measure the quality of the solution and select the better ones. The elite-guided solution generation strategy is designed to exploit the neighborhood of the existing solutions based on the guidance of the elite, Ying Huo [11] presented Elite-guided multi-objective artificial bee colony algorithm; Duc-Hoc Tran [12]proposed a new hybrid multiple objective evolutionary algorithm that is based on hybridization of artificial bee colony and differential evolution, the proposed algorithm integrates crossover operations from differential evolution (DE) with the original artificial bee colony (ABC) in order

to balance the exploration and exploitation phases of the optimization process; Lianbo Ma [13]proposed hybrid artificial bee colony optimizer based on life-cycle is a cooperative and varying-population model where the bee can switch its state periodically according to the local environmental landscape.

2. **ABC algorithm.** There are three kinds of bee, the employed bee, the onlooker bee and the scout. The number of employed bee is equal to the number of the onlooker bee. The employed bee conducts a global search, then all employed bees finish the search, they return the area of information exchange to share information with other bees by swinging dance. The more abundant nectar source, the greater probability to be selected, the more the onlooker bees. Then the onlooker bees search in the areas like the employed bee. The employed bee and onlooker bee choose the better nectar source position as the next generate solution according to the greedy rule.

Assume that the dimension of problem is D, the position of nectar source corresponds to the points of solution space, the ith (i=$1, 2, \cdots, NP$). nectar sources quality is regarded as the fitness of solution fit_i, the number of solution (NP, i.e. the nectar sources number), is total of the number of employed bees and onlooker bees. $\mathbf{X}_i = \{x_{i1}, x_{i2}, \cdots, x_{iD}\}$ represents the location of the ith nectar source, the location $\mathbf{X}_i$ is updated randomly in the d dimensions as follows(d=$1, 2, \cdots, D$).

$$x_{id} = L_d + rand(0, 1) \times (U_d - L_d) \tag{1}$$

where x_{id} is the location of the ith bee in the dth dimension; U_d and L_d stand for the Lower and upper bounds of search space respectively.

At the beginning of the search, the employed bee generates a new nectar source around the nectar source according to the formula (2).

$$v_{id} = x_{id} + rand(-1, 1) \times (x_{id} - x_{jd}) \tag{2}$$

where d is a random integer which denotes that the employed bee searches in random dimension; $j \in \{1, 2, \cdots, NP\}(j \neq i)$ represents that it chooses a nectar source which is different from the nectar source in the NP nectar source; is a random number with uniform distribution between 0 and 1.

When the fitness value of new nectar source $\mathbf{V}_i = \{v_{i1}, v_{i2}, \cdots, v_{id}\}$ is better than $\mathbf{X}_i$, the $\mathbf{V}_i$ replaces $\mathbf{X}_i$, otherwise, the ABC algorithm keeps $\mathbf{X}_i$, and update the employed bees according to the formula (2). After updating, the employed bee feed information back to the onlooker bee. The onlooker bee chooses the employed bee to follow in roulette way according to the probability p_i, and determined the retention of nectar according to the same greedy method of the employed bee. p_i is computed as follows.

$$p_i = \frac{fit_i}{\sum_{j=1}^{SN} fit_j} \tag{3}$$

In the processing of search, if $\mathbf{X}_i$ does not find the better the nectar source after reaching the threshold and times iteration, $\mathbf{X}_i$ would be given up and the employed bee would be changed into the scout. The scout generates randomly a new nectar source which replaces $\mathbf{X}_i$ in the search space. The above-mentioned process can be described as follows.

$$v_i^{t+1} = \begin{cases} L_d + rand(-1, 1) \times (U_d - L_d) & trial \geq limit \\ v_i^t & trial < limit \end{cases} \tag{4}$$

Generally, ABC algorithm takes the minimum optimization problem as an example; the fitness value of solution is estimated by formula (5).

$$fit_i = \begin{cases} 1/(1+f_i) & f_i \geq 0 \\ 1+abs(f_i) & otherwise \end{cases} \tag{5}$$

where the f_i is the function value of solution.

3. **Artificial Bee Colony based on Special Central and Adapt Number of Dimensions Learning (SALABC).**

3.1. **Special Central Strategy.** With the change of populations activity range, the center may be closer to the optimal value [14], so the paper introduces the special central theory to accelerate the convergence velocity.

Definition 3.1. *Special Central Bee (SCB)— in the processing of evolution, x_{ij} (j=1, 2, $\cdots$, D) stands for the th bee, the SCB is remembered as P^{SCB},which is defined as follows.*

$$P_j^{SCB} = \frac{1}{NP}(\sum_{i=1}^{NP} x_{i,j}) \tag{6}$$

It can jump out of the boundary to become non-feasible solution, so reset it by (7).

$$P_d^{SCB} = \begin{cases} U_d & P_d^{SCB} > U_d \\ L_d & P_d^{SCB} < L_d \end{cases} \tag{7}$$

Then we calculate and estimate the fitness value of special central, then regard the optimal solution as the individual , comparing with the current populations optimum. G is expressed as follows.

$$G = \begin{cases} P_d^{SCB} & f(P_d^{SCB}) < f(G) \\ G & f(G) < f(P_d^{SCB}) \end{cases} \tag{8}$$

3.2. **Adapt number of dimensions learning.** The dimensions is simple in standard ABC, the search method of employed bee and onlooker bee greatly influences the convergence velocity and the quality of solution, and the employed bee and onlooker bee only learn randomly from one dimension, the learning ability and sharing ability of the above two bees are limited to some extent. In order to improve the above shortcomings, the paper presents adapt number of dimensions learning strategy. We define a global integer array $Adapt[NP]$, and initialize the employed bee or onlooker bee X_i and a random integer $Adapt[i-1] (0 \leq i \leq NP-1)$.

Because $Adapt[i-1]$ may jump out of boundary [1,D], we reset it as followings.

$$Adapt[i-1] = \begin{cases} D & Adapt[i-1] > D \\ 1 & Adapt[i-1] \leq 0 \end{cases} \tag{9}$$

In the stage of updating, the employed bee or onlooker bee X_i attains a new nectar source V_i in the number of $Adapt[i-1]$ dimensions according to the value of Adapt[i-1], updating Adapt[i-1] by (10).

$$Adapt[i-1] = \begin{cases} Adapt[i-1]-1 & f(X_i) < f(V_i) \\ Adapt[i-1]+1 & f(X_i) \geq f(V_i) \end{cases} \tag{10}$$

3.3. **Improved updating rules of the scout.** Search updating rules only learn randomly from one nectar source in ABC, the learning ability and sharing ability are limited to some extent. So we update the rules of the employed bee and onlooker bee as follows.

$$V_{id} = G_d + \varphi \times (G_d - X_{id}) + \omega \times (X_{id} - X_{jd}) \tag{11}$$

where d is a random number between 1 and D, d represents that the employed bee chooses one dimension to search; $j \in \{1, 2, \cdots, NP\}$ stands for that the rules choose randomly a nectar source not equal to i from NP nectar sources; G_d is the location of global optimal source in the d dimensions; ψ is uniform random number between -0.5 and 0.5, and ω is random number between -1.0 and 1.0; ψ and ω determine the perturbation amplitude.

In formula (11), new nectar source not only learns from source around it and global optimal source, but also inherits global optimal nectars advantage and enhances the global searching ability.

3.4. **The steps of our method. Step 1:**Initialization of parameters and nectar source, including the number of employed bee and onlooker bee NP,$limit$, the maximum estimation times MAX_{FES} and iterations times $maxCycle$;

Step 2: Initializing the location of NP nectar sources, updating the dimensions $Adapt[NP]$, and calculating the fitness value and degree;

Step 3: Ensuring the P^{SCB} according to formula (6),choosing the optimal nectar source by (7) and (8).

Step 4: Updating that $X_i(i = 1, 2, \cdots, NP)$ attains the new source V_i in $Adapt[i-1]$ dimensions by (11), and updating $Adapt[i-1]$ according to (10).

Step 5: Judging $trial \geq limit$, if it is satisfied, initializing the ith nectar source.

Step 6: Updating global optimal value and location, and recording the best nectar source. if the algorithm satisfied the ending condition, output the optimum, otherwise , please turn to step 3 and continue.

4. Experiments and Results Analysis.

4.1. **Test functions.** 9 well-known test functions are shown in the table 1, $f_1 \sim f_6$ are unimodal functions, functions $f_7 \sim f_9$ are multimodal functions. Most optimization algorithms fall into local optimal solution in the process of searching the optimum.

Table 1. Benchmark function set

Function No	Function Name	Range	The optimal value
f_1	Sphere	[-100,100]	0
f_2	Schwefel 2.22	[-10,10]	0
f_3	Schwefel 1.2	[-100,100]	0
f_4	Schwefel 2.21	[-100,100]	0
f_5	Step	[-100,100]	0
f_6	Quartic with noise	[-1.28,1.28]	0
f_7	Rastrigin	[-5.12,5.12]	0
f_8	Ackley	[-32,32]	0
f_9	Schwefelenalized	[-50,50]	0

4.2. **Experiments.** In order to verify our methods advantage of convergence velocity and optimization, we compare it with standard ABC [1], Gbest-guided Artificial Bee Colony Algorithm [14](GABC), Modified Artificial Bee Colony Algorithm [15] (MABC) and Multi-strategy Ensemble Artificial Bee Colony Algorithm [16](MEABC), in the experiments, $MAX_{FES} = 200000$, $limit = 100$.

Table 2 presents results of 5 kinds of algorithms over 30 dimensions, where Mean stands for the mean optimal fitness value and Std.Dev represents the standard deviation. In order to eliminate the influence of the random of the algorithm, the algorithm runs 30 times independently, and the final average value is the final result of the algorithm. The best values are shown in bold.

Table 2. the comparison results of 5 algorithms on famous benchmark functions

Function Name		ABC	GABC	MABC	MEABC	SALABC
Sphere	Mean	9.58e-16	1.09e-32	4.02e-40	4.85e-40	**1.61e-61**
	std.Dev	1.07e-15	4.01e-32	1.58e-39	2.31e-40	**4.71e-60**
Schwefel 2.22	Mean	1.21e-10	6.65e-18	1.67e-21	1.25e-21	**1.25e-36**
	std.Dev	2.67e-10	1.03e-17	6.37e-18	3.56e-21	**1.63e-35**
Schwefel 1.2	Mean	7.66e3	7.88e3	1.03e4	9.81e3	**7.14e-5**
	std.Dev	8.84e3	1.40e4	1.20e4	2.49e3	**1.19e-3**
Schwefel 2.21	Mean	2.25e1	1.76e1	4.22e0	4.89e0	**1.16e-103**
	std.Dev	2.25e1	1.32e4	2.62e0	1.37e0	**1.05e-102**
Step	Mean	**0**	**0**	**0**	**0**	**0**
	std.Dev	**0**	**0**	**0**	**0**	**0**
Quartic with noise	Mean	1.73e-1	8.11e-2	3.18e-2	2.29e-2	**1.36e-3**
	std.Dev	2.12e-1	9.80e-2	3.34e-2	1.38e-2	**3.47e-3**
Rastrign	Mean	1.95e-14	**0**	**0**	**0**	**0**
	std.Dev	1.65e-13	**0**	**0**	**0**	**0**
Ackley	Mean	1.17e-9	3.49e-14	2.88e-14	2.90e-14	**9.71e-15**
	std.Dev	2.66e-9	1.69e-14	1.37e-14	1.32e-14	**1.09e-14**
Penalized	Mean	7.44e-16	**1.57e-32**	**1.57e-32**	3.02e-17	**1.57e-32**
	std.Dev	6.26e-16	**1.50e-47**	**1.50e-47**	0	**1.50e-47**

We can see from the table 2 that the mean and standard deviation of SALABC is better than other 4 algorithms on Sphere, Schwefel 2.22,Schwefel 1.2,Schwefel 2.21, Quartic with noise and Ackley functions, our method attains the same optimization effects as 4 algorithms on Step function. SALABC, GABC, MABC and MEABC can achieve the same results on Rastrigin function, but SALABC outperforms ABC on Rastrigin. In the Penalized function, for the comparison of SALABC, GABC and MABC, they attain the same results on Penalized function, our method is better than ABC and MEABC. In order to compare the performance of 5 algorithms, Friedman test is employed to analyze results. Table 3 presents the average ranking of SALABC, MEABC, MABC, GAB and ABC on 9 test functions. The smaller is value of ranking, the better the performance, the higher the rank. The best values are shown in bold. From the results in the table 3, the SALABC has the best performance among 5 algorithms.

4.3. **Convergence curve analysis.** To illustrate the convergence velocity of 5 ABC variants in the evolution processing, we give the convergence performance curve of SALABC, MEABC, MABC,GABC and standard ABC algorithm in 30 dimension on 9 test functions. Abscissa means function evaluations, ordinate shows logarithm of fitness value.

Table 3. results of 5 algorithms using Friedman test

Algorithm	Ranking
SALABC	**2.00**
MEABC	2.89
MABC	3.00
GABC	3.11
ABC	4.00

We can see from Fig. 1, the whole performance of SALABC is better than other ABC variants on optimization and convergence velocity.

5. **Conclusions.** The paper defined special central bee on the basis of standard ABC, and presents artificial bee colony based on special central and adapt number of dimensions learning. We build special central bee and choose greedily better nectar source as new global optimal location from current global optimal bee and special centre bee; then change the evolution rule of standard ABC, it can increase the ability of population learning from special central bee, enhance the exploration ability, improve the learning ability of population, accelerate the convergence velocity and improve the performance of new method. Finally, we compare SALABC with other famous ABC variants, for example, ABC, GABC, MABC and MEABC on 9 test functions, the results show that SALABC outperforms 4 ABC variants on optimum, precision and convergence velocity. So possible future work [17, 18]is how to adapt artificial bee colony based on special ventral and adapt number of dimensions learning to different optimization problems.

Acknowledgment. This work is supported by the National Natural Science Foundation of China under Grant (Nos. 61261039, 61263029,6 1563036), Science Foundation of Jiangxi Province under Grant (Nos. 20132BAB211031, 20151BAB207067), Science and Technology Support Program of Jiangxi Province under Grant(Nos. 20142BBG70034, 20142BBE50040), Jiangxi Province Department of Education Science and Technology Project under Grant (Nos. KJLD13096, GJJ151133), Science and Technology Planning Project of Nanchang City under Grant (No.2014HZZC008), National Undergraduate Training Programs for Innovation and Entrepreneurship (Nos. 201311319004, 201311319015) and college students innovation and entrepreneurship training program of Jiangxi Province(No. 201411319028).

REFERENCES

[1] D. Karaboga, An idea based on honey bee swarm for numerical optimization, *Kayseri: Erciyes University*, 2005.

[2] D. Karaboga, C. Ozturk, Neural networks training by artificial bee colony algorithm on pattern classification, *Neural Network World*, vol. 19, no. 3, pp. 279-292, 2009.

[3] D. Karaboga, C. Ozturk, A novel clustering approach: artificial bee colony algorithm, *Applied Soft Computing*, vol. 11, no. 1, pp. 652-657, 2011.

[4] D. Karaboga, B. Akay, A modified artificial bee colony algorithm for constrained optimization,*Applied Soft Computing*, vol. 11, no. 3, pp. 3021-3031, 2011.

[5] S. H. Cai, W. Long, J. J. Jiao, Hybridizing artificial bee colony with biogeography-based optimization for constrained mechanical design problems, *Journal of Central South University*, vol. 22, no. 6, pp. 2250-2259, 2015.

[6] T. K. Dao, T. S. Pan, T. T. Nguyen, S. C. Chu, A Compact Articial Bee Colony Optimization for Topology Control Scheme in Wireless Sensor Networks, *Journal of Information Hiding and Multimedia Signal Processing*, vol. 6, no. 2, pp. 297-310, 2015.

[7] P. W. Tsai, M. K. Khan, J. S. Pan, B. Y. Liao, Interactive Artificial Bee Colony Supported Passive Continuous Authentication System, *IEEE Systems Journal*, vol. 8, no. 2, pp. 395-405, 2014.

[8] J. S. Pan, H. B. Wang, H. N. Zhao, L. L. Tang, Interaction Artificial Bee Colony Based Load Balance Method in Cloud Computing, *The 8th International Conference on Genetic and Evolutionary Computing*, pp. 49-57, 2014.

[9] X. B. Li, L. L. Liu, G. Q. Li, X. Q. Guan, Cooperative spectrum sensing based on an efficient adaptive artificial bee colony algorithm, *Soft Computing*, vol. 19, no. 3, pp.597-607, 2015.

[10] P. W. Tsai, J. S. Pan, B. Y. Liao, S. C. Chu, Enhanced Artificial Bee Colony Optimization,*International Journal of Innovative Computing, Information and Control*, vol. 5, no. 12, pp. 5081-5092, 2009.

[11] Y. Huo, Y. Zhuang, J. J. Gu, S. Ni, Elite-guided multi-objective artificial bee colony algorithm, *Applied Soft Computing*, vol. 32, no. 7, pp. 199-210, 2015.

[12] D. H. Tran, M. Y. Cheng, M. T. Cao, Hybrid multiple objective artificial bee colony with differential evolution for the time cost quality tradeoff problem, *Knowledge-Based Systems*, vol. 74, no. 1, pp. 176-186, 2015.

[13] L. B. Ma, K. Y. Hu, Y. L. Zhu, H. N. Chen, A hybrid artificial bee colony optimizer by combining with life-cycle, Powells search and crossover, *Applied Mathematics and Computation*, vol. 252, no. 1, pp. 133-154, 2015.

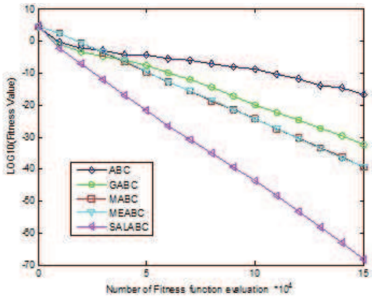
[14] G. P. Zhu, S. Kwong, Gbest-guided artificial bee colony algorithm for numerical function optimization, *Applied Mathematics and Computation*, vol. 217, no. 7, pp. 3166-3173, 2010.

[15] W. F. Gao, Sanyang Liu, A modified artificial bee colony algorithm, *Computers* & *Operations Research*, vol. 39, no. 3, pp. 687-697, 2012.

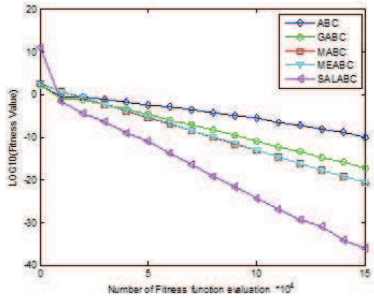
[16] H. Wang, Z. J. Wu, S. Rahnamayan, H. Sun, Y. Liu, J. S. Pan, Multi-strategy ensemble artificial bee colony algorithm, *Information Sciences*, vol. 279, no. 20, pp. 587-603, 2014.

[17] L. H. Xu, X. F. Li, S. X. Yang, Intelligent Information Processing and System Optimization, *Intelligent Automation and Soft Computing*, vol. 17, no. 7, pp. 829-833, 2011.

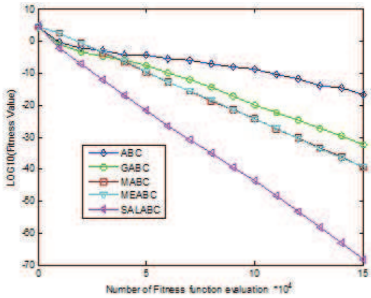
[18] L. Z. Xu, X. L. Feng, C. L. Wen, Sequential Fusion Filtering for Networked Multi-Sensor Systems Based on Noise Estimation, *ACTA ELECTRONICA SINIC*, vol. 42, no. 1, pp. 160-168, 2014.



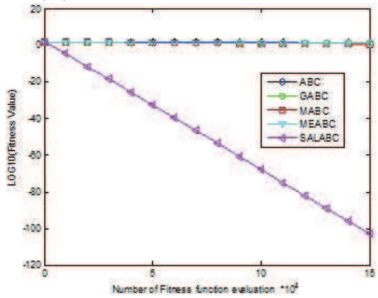
(1)Sphere function



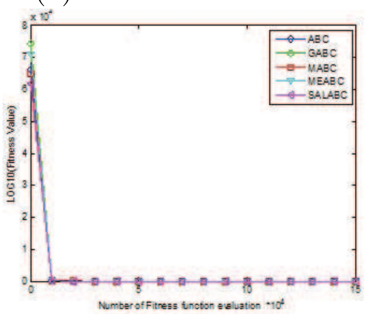
(2) Schwefel2.22 function



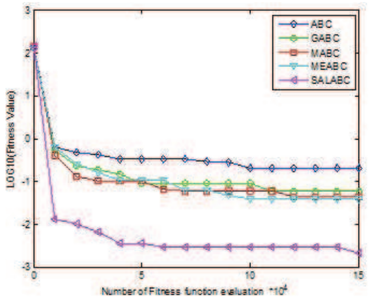
(3) Schwefel 1.2 function



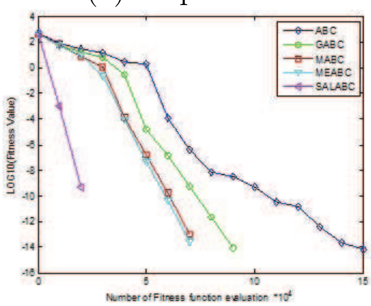
(4) Schwefel 2.21 function



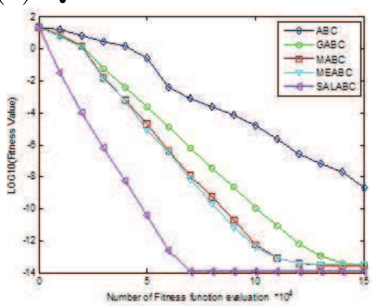
(5) Step function



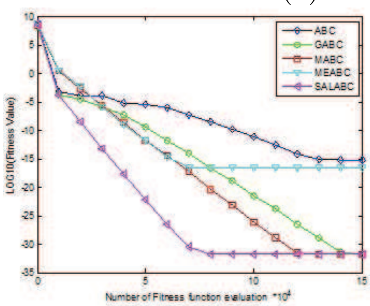
(6) Quartic with noise function



(7) Rastrigin function



(8) Ackley function



(9) Penalized function

Figure 1. Convergence curve of 5 algorithms.