Journal of Information Hiding and Multimedia Signal Processing ©2018 ISSN 2073-4212
Ubiquitous International Volume **9**, Number **4**, July **2018**

Hybrid Algorithm for Optimization Problems based on EBA and ABC

Si-Jing Cai[1,2,*], John F. Roddick[3] and Shu-Chuan Chu[3]

[1]School of Information Science and Engineering
Fujian University of Technology
No. 3, Xueyuan Road, University Town, Minhou, Fuzhou, 350118, China

[2]Research center for Microelectronics Technology in Fujian University of Technology

[3]College of Science and Engineering
Flinders University, Tonsley 5042, South Australia

[*]Corresponding author: caisijing@163.com

Received April, 2018; revised May, 2018

Abstract. *In this paper, we concentrate on the modification of the evolved bat algorithm (EBA), designed for solving numerical optimization by utilizing the scheming idea of Artificial Bee Colony algorithm (ABC) .Three roles of bat colony and six successive processes are realized to accelerate the convergence characteristic of the modificatory algorithm, namely evolved bat colony algorithm (EBC). In the initialization, two roles (employed bats and onlooker bats) are set with equal probability, and the movement law of EBA is applied for obtaining candidate solutions by the greedy selection strategy. The last modified phase is movement phase of scout bats, the employed bats become scouts with randomly search in this phase, when one solution cannot be further improved any more. In the experiments, five well-know benchmark functions are utilized to evaluate the performance of EBC algorithm. The obtained results show that the EBC algorithm is superior to EBA according to solution quality and robustness.*

Keywords: EBA, ABC, Swarm intelligence, Greedy selection strategy, Numerical optimization

1. **Introduction.** Swarm intelligence algorithm is a set of evolutionary steps, inspired by the intelligent behaviors in biome, to solve optimization problems in fields of engineering process and mathematical programming[1-3]. Artificial Bee Colony algorithm (ABC), Bat algorithm (BA), QUATRE[4-6] and Monkey King Evolution[7-9] are the typical representative.

BA is a bio-inspired heuristic algorithm proposed in 2010 by Yang [10-11]. The algorithm is flexible and efficient, but the good results was obtained just on the lower-dimensional problems[12]. Evolved Bat Algorithm (EBA) has been developed by Tsai et al in 2012, to improve the performances for higher-dimensional[12]. However, there are still some defects in EBA, for example, the process of global optimum is similar blind[11]. On the other hand, The ABC algorithm is a effective method of the numerical optimization problems [13]. The algorithm was excited by the seeking behavior of honeybee colony by Karaboga et al in 2005. It shows superior performance in solving the unconstrained problems[14]. The efficient organizing skills and unique highly developed foraging ability of bee swarms is useful to overcome the defects of blind guess process in EBA. So, we proposed the improved EBA algorithm based on the scheming idea of ABC algorithm using the movement law of EBA, which can search the solution space more efficiently.

The rest of the paper is organized as follows: the principle of EBA and ABC algorithms are briefly reviewed in section 2. In section 3, the proposed improved method is described. The experiments and the experimental results are given in section 4. Finally, the conclusions are introduced.

2. **Background and Knowledge.** The Bio-inspired algorithms are simulated the animals' behaviors in biome, like ants, bees, bats, fishes and so on. These algorithms are extremely useful in the aspect of solving optimization problems, such as EBA and ABC algorithm, and they have been successfully applied in practical optimization problems. For these reasons, an introduction about EBA and ABC algorithm is presented in the section.

2.1. **Evolved bat algorithm.** BA is one kind of problem-independent algorithms, motivated by the echolocation characteristic of bats. It was proposed by simulating the behaviors of micro-bat. Inspired by the process's characteristics, Tsai et al. proposed the EBA in 2012, by redesigning the process for the conventional BA. The new solution is executed by Eq(1-3)[12].

$$x_i^t = x_i^{t-1} + D \tag{1}$$

$$D = \frac{V \cdot \Delta T}{2} = 0.17 \cdot \Delta T (km/s) \tag{2}$$

$$x_i^{t_R} = \beta \cdot (x_{best} - x_i^t) \tag{3}$$

Where the parameter x_i^t indicates a solution obtained by the i^{th} artificial agent.

D is the displacement during the time interval.

V represent the sound speed.

$\Delta T \in [-1, 1]$, denotes the time interval between sending the sound wave and receiving the echo.

$x_i^{t_R}$ stands for the solution of the bat after the random walk process.

β is a random constant within the interval [0, 1].

x_{best} means the near best solution, calculating in the preceding iterations.

The displacement during the time interval (D) is immediately calculated by Eq(2), and uses the range of ΔT to represent the time and orientation. The Eq(3) is a random walk process to avoid the local optimum. This step is carried out stochastic[3]. The mathematical model with fewer parameter settings, has a simple structure, and it is easy to realize, so the EBA were attracted much attention in recent years.

2.2. **Artificial bee colony algorithm.** The ABC algorithm is simulated the foraging food behavior of honeybee colony. the bee colony consists of three groups, employed bees, onlookers and scouts[14]. Employed bees exploits nectar around a specific food source area in their memories, and share the nectar information with onlookers, such as the distance, direction, and profitability[15]. The onlookers select a food source depended on the information which is provided by the employed bees, usually is nectar amount of the food source, it corresponds to the quality (fitness) of the associated solution[15]. That is, the more amount of a food source, the higher probability that the food source would be selected by onlookers. After arriving at the selected food source, a further searches around the selected food source area is given by onlooker bee. At last, the new food source is searched by scouts randomly.

In ABC algorithm, the positions of the food source signifies the potential solutions of the optimization problem. The number of the solutions is set to be equal to the number of employed or onlooker bees, which means one food source is extracted by one employed bee [16-17]. The process can be described as following:

Step 1. Initialization: Generate the initial population (SN) in the D-dimensional solution space $X_i = (x_{i1}, x_{i2} ... x_{iD})$ randomly according to Eq(4). Then calculate the fitness of each food sources according to the fitness functions. Assign the food sources to the employed bees.

$$x_{ij} = \theta_{ij\min} + \gamma(\theta_{ij\max} - \theta_{ij\min}) \tag{4}$$

where x_{ij} is the j^{th} dimension of i^{th} food source which will be assigned to i^{th} employed bees.

$\theta_{ij\min}$ is lower bound of the j^{th} dimension.

$\theta_{ij\max}$ is upper bound of the j^{th} dimension.

$\gamma \in [0, 1]$, is a uniform random value.

Step 2. Move the employed bees: Produce new position of each employed bee in their respective areas. The employed bees further search around their current food sources to get a new positions be replaced with the Eq(5). Then calculate the fitness values of each employed bee's new solution, if the fitness of the new position is better than the old one, memorize the new position, otherwise retained.

$$X_{newj} = X_{ij} + (X_{ij} - X_{kj}) \times \phi_i \tag{5}$$

where $\phi_i \in [-1, 1]$, is a random value.

Step 3. Move the onlookers: Calculate the probability of food source received from employed bees in accordance with Eq(6), each onlooker bee choose a food source to search by roulette wheel selection mechanism using the Eq(5). Then calculate fitness of each onlooker's new solution, if the fitness of the new solution is better than the old one, memorize the new position, otherwise retained.

$$P = \frac{Fit_i}{\sum_{j=1}^{Ne} Fit_j} \tag{6}$$

where P_i is probability of the i^{th} employed bee by an onlooker bee.

Fit_j denotes the fitness value of the i^{th} solution.

Ne is the number of employed bees.

Step 4. Move the scouts: If one food source cannot be further improved through a predefined number of cycles, the employed bee would abandoned the food source and become a scout generated new food source position by Eq(4)[15], also calculate the fitness of scout's new position and update.

Step 5. Update the best food source solution by fitness value at this iteration.

Step 6. Check the termination condition: Check the termination condition to decide the next step, go back to the step 2 or finish.

3. **Improved Method: Evolved Bat Colony Algorithm.**

With the easy realization, simple structure and strong robustness of ABC[14], a improved Bat algorithm based on EBA and ABC is proposed, named evolved bat colony algorithm (EBC algorithm). The bat colony can be partitioned into three roles: employed bats, onlookers and scouts.

The quantity of employed bats which is equivalent to that of onlookers is half of the colony. An employed bat is related to one food target, and share the information of the target with onlookers. Each onlooker selects a food target area to further search dependent on the quality (fitness) of the target. A food target will be abandoned if it is not updated, and the corresponding employed bat will become a scout to search for a new food target randomly. The steps of EBC algorithm as follows:

Step 1. Initialization phase: Generate an initial population randomly by Eq(4), the test case sets including all bats. Calculate fitness of each bat by using defined fitness function.

Step 2. Arrange roles: according to the value of the fitness function, the first 50% of the bat colony constitutes employed bats, and the rest of bats consists of the onlookers.

Step 3. Move the employed bats phase: Calculate the next generation search around each employed bat's current positions to forage new solutions. The motion law of employed bats satisfies the echo characteristics as the Eq(1-2). Then decide whether to update the new solution in accordance with the greedy selection strategy, the process is: calculate the fitness value of new solution, if it better than the old one, memorize the new position, otherwise retain.

Step 4. Move onlooker bats phase: evaluate the probability value P_i of each employed bats utilizing Eq(6). Onlooker bats select a solution to further search on the basis of P_i. The moving law of onlooker bats satisfies Eq(1-2). Then determine with to update by the greedy selection strategy.

Step 5. Move the scout bats phase: if one food target cannot be updated more, the employed bat become scout bat, and generates a new solution randomly replace the old one using Eq(3). Calculate the fitness value of new solution to determine with to update by the greedy selection strategy.

Step 6. Update phase: Memorize the best fitness value and the best food source solution by fitness values of all bats. Check the termination condition to decide the next step, go back to step2 or finish.

4. **Experiment Design and Experimental Results.**

The experiment is executed by Matlab 2010a to analyze the different between the EBC algorithm and EBA. In this section, the performance of the two algorithms is compared by optimizing benchmark functions selected from literatures.

4.1. **Benchmark functions.**

The five well-known benchmark functions utilized in the experiment are presented in Eq(7)-(11).

$$f1(x) = -20\exp(-0.2\sqrt{\frac{1}{d}\sum\nolimits_{i=1}^{d} x_i^2}) - \exp(\frac{1}{n}\sum\nolimits_{i=1}^{n}\cos 2\pi x_i) + 20 + e \tag{7}$$

$$f2(x) = -[\sum\nolimits_{i=1}^{d}\sin x_i + \sum\nolimits_{i=1}^{M}\sin(\frac{2}{3}x_i)] \tag{8}$$

$$f3(x) = -\sum\nolimits_{n=1}^{d}\sin x_i \sin^{2m}(\frac{ix_i^2}{\pi}) \tag{9}$$

$$f4(x) = \sum\nolimits_{n=1}^{d}(\lfloor x_i + 0.5\rfloor^2) \tag{10}$$

$$\begin{gathered} f5(x) = a_0 + \sum_{i=1}^{D}(\sum\nolimits_{k=0}^{k\max}[a^k cos(2\pi b^k(x_i+0.5))]) - D\sum\nolimits_{k=0}^{k\max}[a^k cos(2\pi b^k 0.5)] \\ a = 0.5, b = 3, k\max = 20 \end{gathered} \tag{11}$$

In the test, the parameter settings of EBC algorithm and EBA are the same. The details of the test functions are listed in Table 1. The initialization and search range of each function, named 'Range' in the table, is different. All benchmark functions are run with

5, 10, 15, 20, 30 dimensions respectively. Each test is repeated 25 runs independently with different random seeds, and the population size is 20.

Table 1. Parameters of Benchmark functions used in the experiments

Fun	**Range**	**Dimensions**	f_{min}	**No. of cycle**	**No. of Iteration**	**No. of population**
f_1	-32,32	5,10,15,20,30	0.0	25	7000	20
f_2	-0.5,0.5	5,10,15,20,30	0.0		7000	
f_3	0,π	5,10,15,20,30	0.0		8000	
f_4	-100,100	5,10,15,20,30	0.0		4000	
f_5	-5.12,5.12	5,10,15,20,30	0.0		5000	

As shown in Table 1, the parameters, dimensions, iteration runs and population size are equal to each other to evaluate the performances of the two algorithms.

4.2. **The results.** In this section, the test results of the two algorithms are given in Table 2, where the best, worst and mean values are listed in bold. In the table, parameters “num” and “time” means the iteration runs and iteration times.

According to the data in the table, EBC algorithm is a remarkable potential algorithm, which combines the scheming idea of ABC algorithm and movement law of EBA, the convergence curves for different function are shown in Fig.1. On the basis of the data above, the convergence is enhanced, while the dimensionality of iteration is decreased, such as benchmark function 1and 4. The convergence of function 1 in 5-dimension is better than 10-dimension , as drawn in Fig.1.(a) and (b). On the other hand, the performance of EBC algorithm is enhanced, as the dimensionality is increased of benchmark function 2, 3 and 5. We will working on improving the performance in different dimensions in the further work. The Fig.1 shows the fitness value of EBC algorithm and EBA.

5. Conclusions.

In this research, a evolved algorithm EBC algorithm, which is based on the EBA combined the characteristic of ABC, is presented to solve the multi-objective optimization problems. In the evolved method, the roles information is obtained to avoid the blind search: the optimal 50% of the bat colony as employed bats search in each specific area by greedy selection strategy, the remaining 50% bats as onlookers search the area according to employed bats by the probability values. On the other hand, the movement law obeys the movement of EBA, it is very simply and effective. The performance of EBC algorithm and EBA is evaluated with 5 benchmark functions. The results indicate that: the EBC algorithm provides superior solutions than EBA, with the some emulational condition, such as the number of cycles, population size, benchmark functions ect. The advantage of the EBC algorithm is simple framework, fewer parameter and strong robustness. However, many notions might be considered in the further work of EBC algorithm, such as how to improve the performance of the EBC algorithm in different dimension.

Acknowledgment. This work was supported by the Fujian Provincial Education Fund (Grant No. JAT160332, JAT170372), Fujian Province Planning Subject Fund (Grant No. FJJKCG15-172) and experimental didactical subject of Fujian University of Technology (Grant No. SJ2017008).

Table 2. The results EBC algorithm and EBA

	fn	IBA					EBA				
		best	worst	average	num	time	best	worst	average	num	time
Dim=5	F1	0.3404	19.4896	2.9588	6789	3.6933	0.0215	13.2856	5.3256	5988	3.3168
	F2	9.5201	9.5289	9.5258	6963	3.5837	9.5298	9.5291	9.5296	3725	1.9612
	F3	-3.0874	-4.2348	-3.6596	7210	3.5173	-3.6520	-4.6922	-4.4363	7142	3.6281
	F4	0.0213	0.1064	0.0567	3959	1.8550	0	0	0	2804	1.4020
	F5	-10.1633	-11.8973	-10.9234	4794	8.2391	-13.6167	-15.2081	-14.4755	4805	8.0421
Dim=10	F1	0.8958	20.3531	8.8672	6668	3.8542	5.1270	13.3830	9.6522	3567	1.9765
	F2	18.2733	19.0272	18.8019	6668	3.4897	19.0538	19.0586	19.0565	6905	3.7931
	F3	-3.4900	-5.4117	-4.4157	7944	4.1574	-6.3971	-8.5146	-7.4986	7404	4.0386
	F4	0.2533	0.7048	0.4344	3917	1.9509	0	100	7.32	3936	2.0817
	F5	-24.9992	-32.3459	-28.4948	4648	12.0778	-37.7412	-48.9851	-44.9909	4346	10.9830
Dim=15	F1	2.1068	19.9026	13.1140	6999	3.8822	7.7075	14.7518	11.2071	5840	3.3112
	F2	27.6087	28.3685	28.0277	6682	3.455	28.5665	28.583	28.5766	3720	1.9727
	F3	-4.0366	-5.8305	-4.8207	7982	4.3724	-7.2390	-10.1947	-8.9319	4569	2.6223
	F4	0.6930	1.7091	1.3007	3917	1.8441	0	233	46.72	3999	2.0102
	F5	-44.6573	62.3890	-51.5646	4589	16.0021	-78.4999	-99.2300	-89.1747	3876	-13.1157
Dim=20	F1	2.7540	20.2079	17.3250	6908	3.8699	9.2185	14.7448	12.4090	6482	3.7069
	F2	35.2073	37.5107	36.7643	6932	3.6590	38.0849	38.1074	38.0942	6913	3.7918
	F3	-3.9975	-5.7799	-4.9944	7431	4.3792	-8.0631	-12.4568	-10.1544	5628	3.3306
	F4	1.8250	4.4287	2.8049	3987	1.8802	54	1358	648.9200	3999	2.049
	F5	-72.6245	-100.0183	-80.3733	4907	21.51	-126.4554	-163.1233	-147.0674	4571	10.4349
Dim=30	F1	16.2807	20.5623	19.5582	6999	3.9877	8.9853	15.0132	12.9225	6617	4.0456
	F2	51.8474	55.6519	54.1278	6955	3.9298	57.0781	57.1464	57.1141	6692	3.9084
	F3	-4.9625	-6.3014	-5.5483	7981	5.4258	-10.2209	-15.6289	-12.3842	4476	3.0233
	F4	4.2282	15.0868	10.5355	4000	2.2461	361	3792	2.1023e+03	3998	2.1101
	F5	-126.2826	-181.7947	-144.4758	4558	29.6668	-250.5537	-330.1979	-290.7509	3080	18.9257

REFERENCES

[1] A. Rekaby, Directed artificial bat algorithm(DABA) - A new bio-inspired algorithm, *2013 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, India, pp. 1241-1246, 2013.

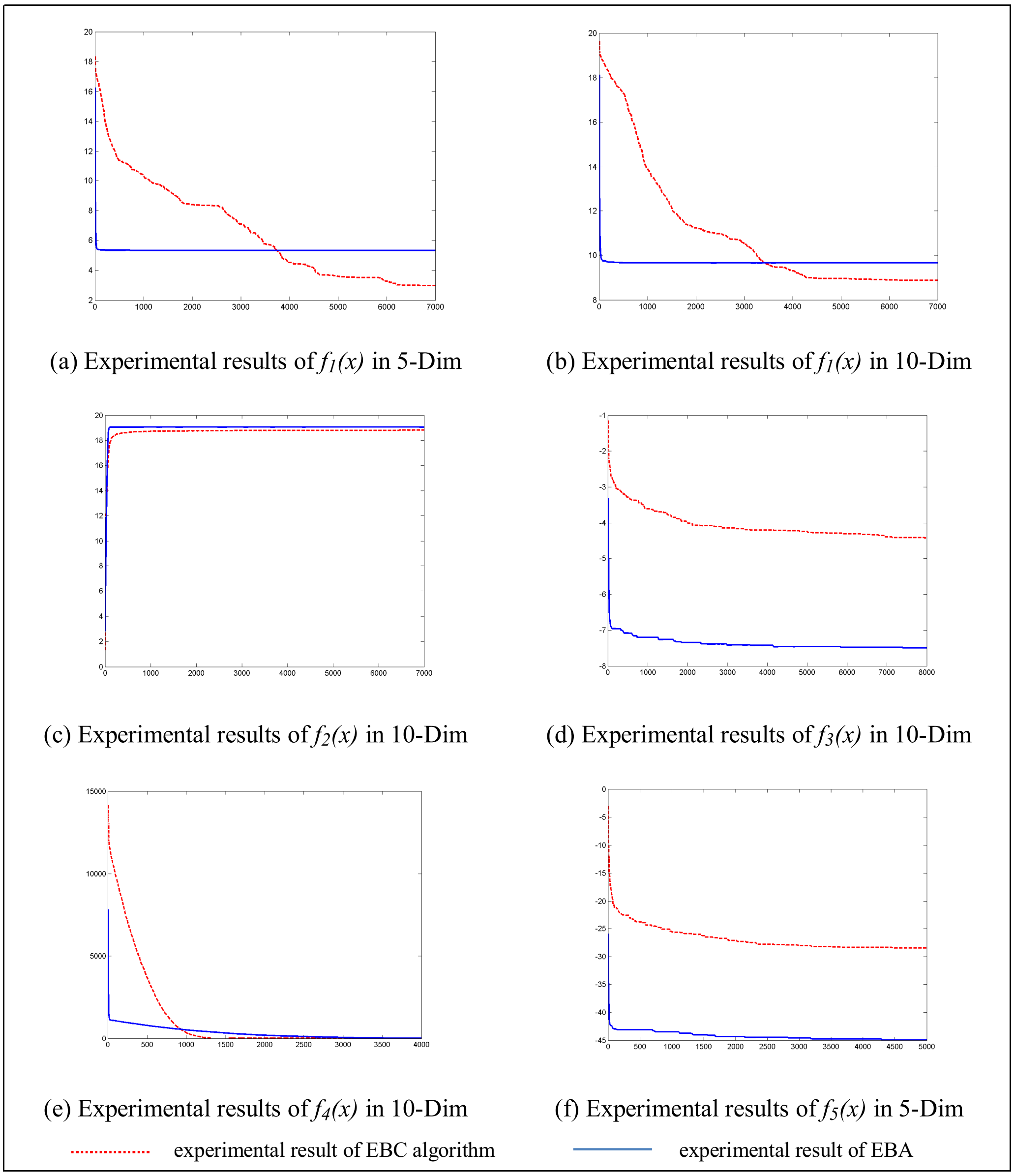
[2] P.W. Tsai, S.J. Cai, V. Istanda and J.S. Pan, Utilizing directional information in evolved bat algorithm for guiding artificial agents, *ICIC Express Letters*, Part B: Applications, vol. 7, no. 7, pp. 1461-1466, July, 2016.

[3] Z.H. Li, Q. Jin, C.C. Chang, A.H. Wang and L. Liu, A novel pixel grouping scheme for AMBTC based on particle swarm optimization, *Journal of Information Hiding and Multimedia Signal Processing*, vol. 7, no. 5, pp. 970-982, September, 2016.

[4] J.S. Pan, Z.Y. Meng, S.C. Chu and J.F. Roddick, QUATRE algorithm with sort strategy for global optimization in comparison with DE and PSO variants, *Advances in Intelligent Systems and Computing*, vol. 682, pp. 314-323, 2018.

[5] Z.Y. Meng, J.S. Pan and X.Q. Li, The QUasi-Affine transformation evolution (QUATRE) algorithm: An overview, *Advances in Intelligent Systems and Computing*, vol. 682, pp. 324-333, 2018.

[6] Z.Y. Meng and J.S. Pan, QUasi-affine transformation evolutionary (QUATRE) algorithm: The framework analysis for global optimization and application in hand gesture segmentation, *International Conference on Signal Processing Proceedings, ICSP*, pp. 1832-1837, March 13, 2017.



(a) Experimental results of $f_1(x)$ in 5-Dim

(b) Experimental results of $f_1(x)$ in 10-Dim

(c) Experimental results of $f_2(x)$ in 10-Dim

(d) Experimental results of $f_3(x)$ in 10-Dim

(e) Experimental results of $f_4(x)$ in 10-Dim

(f) Experimental results of $f_5(x)$ in 5-Dim

experimental result of EBC algorithm — experimental result of EBA

Figure 1. Experimental results of EBC algorithm and EBA

[7] Z.Y. Meng and J.S. Pan, Monkey king evolution: A new memetic evolutionary algorithm and its application in vehicle fuel consumption optimization, *Knowledge-Based Systems*, vol. 97, pp. 144-157, April, 2016.

[8] J.S. Pan, Z.Y. Meng, S.C. Chu and H.R. Xu, Monkey king evolution: An enhanced ebb-tide-fish algorithm for global optimization and its application in vehicle navigation under wireless sensor network environment, *Telecommunication Systems*, vol. 65, no. 3, pp. 351-364, July, 2017.

[9] J. Tang, J.S.Pan, Y.M. Tseng, P.W. Tsai and Z.Y. Meng, Optimal economic dispatch of fuel cost based on intelligent monkey king evolutionary algorithm, *Smart Innovation, Systems and Technologies*, vol. 82, pp. 236-243, 2018.

[10] W. Liu, L.B. Liu, T.Q. Zhang and J. Liu, Multimodal function optimization based on improved ABC algorithm, *Proceedings - 2016 9th International Symposium on Computational Intelligence and Design, ISCID 2016*, vol. 1, pp. 246-249, January, 2017.

[11] E. Osaba, X.S.Yang, F. Diaz, P. Lopez-Garcia and R. Carballedo, An improved discrete bat algorithm for symmetric and asymmetric traveling salesman problems, *Engineering Applications of Artificial Intelligence*, vol. 48, pp. 59-71, February, 2016.

[12] S.J. Cai and P.W. Tsai, Echolocation guided evolved bat algorithm, *Journal of Information Hiding and Multimedia Signal Processing*, vol. 7, no. 1, pp. 153-162, January, 2016.

[13] A.K. Jagadish, S. Goswami, P. Saha, S. Chakrabarty and K. Rajgopal, Artificial bee colony (ABC) based variable density sampling scheme for CS-MRI, *2016 IEEE Region 10 Conference (TENCON) - Proceedings of the International Conference*, pp. 1254-1257, November, 2016.

[14] M.D. Li, H. Zhao, X.W. Weng and H.Q. Huang, Artificial bee colony algorithm with comprehensive search mechanism for numerical optimization, *Journal of Systems Engineering and Electronics*, vol. 26, no. 3, pp. 603-617, June, 2015.

[15] Y. Liu, X.X. Ling, Y. Liang and G.H. Liu, Improved artificial bee colony algorithm with mutual learning, *Journal of Systems Engineering and Electronics*, vol. 23, no. 2, pp. 265-275, April, 2012.

[16] H.D. Xu, M.Y. Jiang and K. Xu, Archimedean copula estimation of distribution algorithm based on artificial bee colony algorithm, *Journal of Systems Engineering and Electronics*, vol. 26, no. 2, pp. 388-396, April, 2015.

[17] J. Zhao, L. Lv, H. Wang, D.K. Tan, J. Ye, H. Sun and Y.T. Hu, Artifcial bee colony based on special central and adapt number of dimensions learning, *Journal of Information Hiding and Multimedia Signal Processing*, vol. 7, no. 3, pp. 645-652, May, 2016.