TELKOMNIKA, Vol. 11, No. 10, October 2013, pp. 5579 ~ 5587
ISSN: 2302-4046

A Comparison of Improved Artificial Bee Colony Algorithms Based on Differential Evolution

Jianfeng Qiu[1], Jiwen Wang[2], Dan Yang[3], Juanxie*[4]
[1, 2, 3] School of Computer Science and Technology, Anhui University (AHU), Hefei 230039, Anhui, China
[4] Department of Mathematics & Physics, Anhui University of Architecture (AIAI), Hefei 230022, Anhui, China
Corresponding author, e-mail: qiujianf@ahu.edu.cn[1], wjw@ahu.edu.cn [2], yangdan19891025@yahoo.cn [3], qoujianf@sohu.com[4]

Abstract

The Artificial Bee Colony (ABC) algorithm is an active field of optimization based on swarm intelligence in recent years. Inspired by the mutation strategies used in Differential Evolution (DE) algorithm, this paper introduced three types strategies ("rand", "best", and "current-to-best") and one or two numbers of disturbance vectors to ABC algorithm. Although individual mutation strategies in DE have been used in ABC algorithm by some researchers in different occasions, there have not a comprehensive application and comparison of the mutation strategies used in ABC algorithm. In this paper, these improved ABC algorithms can be analyzed by a set of testing functions including the rapidity of the convergence. The results show that those improvements based on DE achieve better performance in the whole than basic ABC algorithm.

Keywords: *Artificial Bee Colony, Differential Evolution, Search Strategy, Best, Rand, Current-to-Best.*

1. Introduction

Optimization is one of attractive fields in not only academic research but also engineering practice. Most problems in real world can be reduced to solve a class of optimization problems. Usually, a class of unconstrained optimization task can be formulated as follows:

$$\min f(\vec{x}), \vec{x} = [x_1, x_2, ..., x_D] \quad (1)$$

where $\vec{x}$ is optimized variable and *D* is the number of parameters to be optimized. As the complexity of the optimization problem increasing, traditional optimization methods cannot solve such problems well.

Recently, biological-inspired optimization algorithms have been proposed to solve such as high-dimension, nonlinear optimization problem in real world. The ant colony optimization (ACO) is inspired by assignment and cooperation among different colonies to solve optimization problems [1]. The particle swarm optimization (PSO) is a meta-heuristic search method based on social behavior of birds and has been widely used to solve various optimization problems [2] [3].

The Differential Evolution (DE) algorithm which is simulating biology evolution process has been one of competitive form evolution algorithm [4] [5]. It has been successful in solving high-dimension, non-linear, large-scale, multimodal optimized problems using DE algorithms and their variants. The performance of the DE algorithms is relying on three stages: mutation, crossover and selection. According to different mutation strategies and adopted number of difference vector, the frequently used mutation strategies in literature include DE/rand/1, DE/rand/2, DE/best/1, DE/best/2, DE/current-to-best/1, and DE/current-to-best/2.The detailed analysis and comparisons can be described in [6]. Each DE algorithm variants may be effective over some problems and poor over other problems. It is not possible to make one DE algorithm always available over all problems [7].

Received April 23, 2013; Revised June 22, 2013; Accepted July 10, 2013

The Artificial Bee Colony (ABC) algorithm is a meta-heuristic algorithm introduced by Karaboga in 2005 [8]. A result about the performance about ABC, PSO, and DE show that ABC is better than or similar to those above listed in [9]. The standard ABC algorithm divided into three stages, such as employed bees stage, onlooker bees stage, and scout bees stage, by imitating foraging behavior of bee colony. The Bee Swarm Intelligence has been showed by division of labor and local interacting among bee colony. To achieve more nectar, each employed bee in bee colony adjusts its searching direction according to its visual information in the neighborhood of the one in its memory.

The standard ABC algorithm is easy to implement and fewer parameters. However, the searching strategies used in standard ABC is more getting trapped into local optimization in solving multimodal problems or slower convergence speed in solving unimodal problems. In order to improve the performance of ABC, some mutation strategies in DE will be used and some analysis and comparison will also be made simultaneously in this paper.

The rest of this paper is organized as follows. Section 2 introduces the basic ABC algorithm and some variants. A comprehensive improved ABC algorithms based on the mutation strategies used in DE algorithm are elaborated in Section 3. Experimental setting and results are presented in Section 4. Finally, conclusions are summarized in Section 5.

2. Artificial Bee Colony Optimizer

In this section, we outline the procedure of basic ABC algorithm and some variants of ABC algorithm. Meanwhile, some questions existing in the above have been proposed.

2.1. Basic Artificial Bee Colony Optimizer

ABC algorithm imitates the foraging behavior of honey bee. The individual of bee colony are classed into one of three types according the different division of labor, that is, employed bees, onlooker bees, and scout bees.

Before searching in search space, the first thing we should do is initialization. The initialization in ABC algorithm is randomly producing food sources to cover the whole search space as much as it possibly can. The position of a food source represents a possible solution in the D-dimension search space. The position x_i is produced as follows [11]:

$$x_{ij} = x_j^{\min} + rand(0,1)(x_j^{\max} - x_j^{\min}) \quad (2)$$

where $x_i = (x_{i1}, x_{i2}, \cdots, x_{ij}, \cdots x_{i,D})$ is the D-dimension position vector of the ith food source; x_{ij} is the jth component of the ith vector; $rand(0,1)$ is a random number in the range [0, 1]. After initialization, the search process is conducted by the employed bees, onlooker bees, and scout bees. In ABC, employed bees produce modification to the current food source in the neighbor of the food source according to its memory (that is similar to the mutation in DE).The modification can be described as follows:

$$v_{ij} = x_{ij} + \Phi_{ij}(x_{ij} - x_{kj}) \quad (3)$$

After producing v_i, the fitness value is calculated for every v_i:

$$fitness_i = 1/(1+f_i) \quad \text{if} \quad f_i \geq 0$$
$$1 + abs(f_i) \quad \text{if} \quad f_i < 0 \quad (4)$$

where f_i is function value of the candidate solution v_i.

For onlooker bees, they always appear in the food source where abundant nectar amount is. So, the probability value P_i for every food source is described as follows:

$$P_i = \frac{fitness_i}{\sum_{i=1}^{NP} fitness_i} \quad (5)$$

In onlooker bees' stage, a random number in the range (0, 1) is produced for every food source. If probability value P_i is greater than the random number, the onlooker bee becomes employed bee. After that, the onlooker bees' search strategy for every food source is the same as the employed bees in (3).

Whenever a food source is depleted by employed bees, the employed bees associated with it will abandon the food source, and become scout. The scout bees perform global exploration in search space and the search process can be defined as (2). The flowchart of basic ABC algorithm is given in Figure 1.

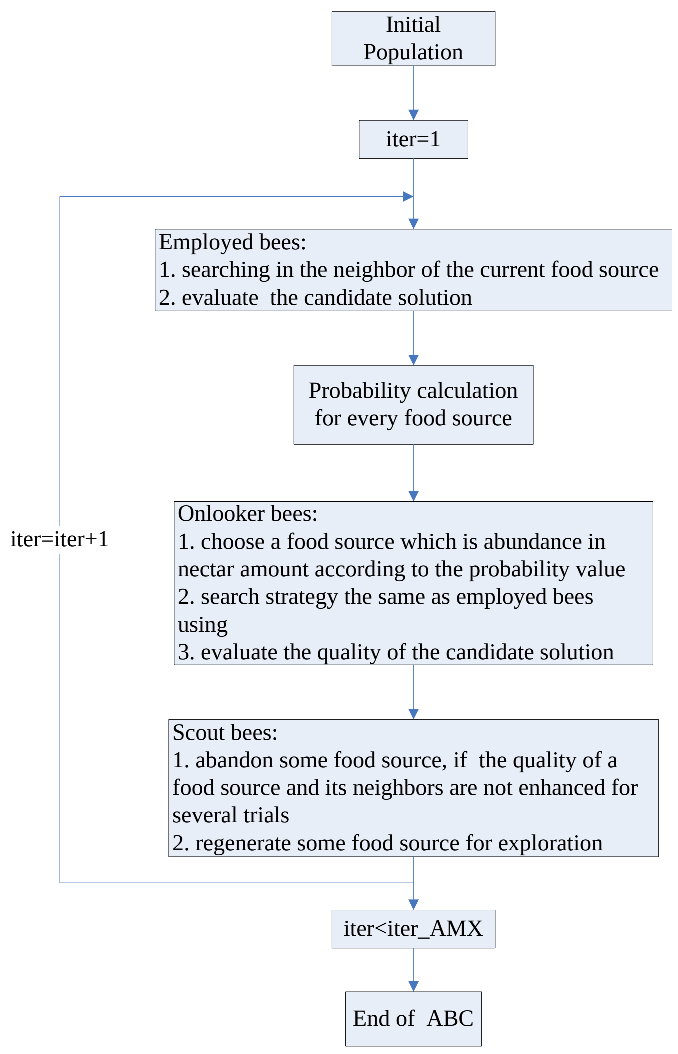

Figure 1. Flowchart of the basic ABC.

2.2. Some Variational ABC Algorithms

Since it was first introduced in 2005 by D. Karaboga [8], it has attracted many attentions in recent years. In this section, some of the variations of ABC are briefly reviewed.

A modified ABC algorithm which the frequency and the magnitude have perturbation is introduced by D. Karaboga [11]. In basic ABC, in order to produce a new solution, there is only one parameter in x_i to be changed. In [11], a modification rate is introduced to improve the convergence speed. Inspired by PSO, an improved ABC algorithm, GABC, is proposed by incorporating the information of global best solution to the current search space. The improvement of the search strategies enhances the exploitation ability which is poor at the basic ABC algorithm [12]. Literature [13-16] introduced some variants of ABC based "best" and "rand" strategies which are used in DE. These improvements which are based on DE are not a comprehensive versions and lack of crosswise comparison. So, in this paper, we will make use of six mutation strategies in DE to improve basic ABC algorithm. Meanwhile, a comprehensive comparison among the six improved ABC algorithm will be explained later.

3. Improved ABC Algorithms based on DE

There are six mutation strategies which are used in DE algorithm. The formulas can be expressed as follows [6]:

"DE/rand/1:" $$\vec{V}_{i,G} = \vec{X}_{r_1^i,G} + F*(\vec{X}_{r_2^i,G} - \vec{X}_{r_3^i,G}) \quad (6)$$

"DE/best/1:" $$\vec{V}_{i,G} = \vec{X}_{best,G} + F*(\vec{X}_{r_1^i,G} - \vec{X}_{r_2^i,G}) \quad (7)$$

"DE/current-to-best/1:" $$\vec{V}_{i,G} = \vec{X}_{i,G} + F*(\vec{X}_{best,G} - \vec{X}_{i,G}) + F*(\vec{X}_{r_1^i,G} - \vec{X}_{r_2^i,G}) \quad (8)$$

"DE/rand/2:" $$\vec{V}_{i,G} = \vec{X}_{r_1^i,G} + F*(\vec{X}_{r_2^i,G} - \vec{X}_{r_3^i,G}) + F*(\vec{X}_{r_4^i,G} - \vec{X}_{r_5^i,G}) \quad (9)$$

"DE/best/2:" $$\vec{V}_{i,G} = \vec{X}_{best,G} + F*(\vec{X}_{r_1^i,G} - \vec{X}_{r_2^i,G}) + F*(\vec{X}_{r_3^i,G} - \vec{X}_{r_4^i,G}) \quad (10)$$

"DE/current-to-best/2:" $$\vec{V}_{i,G} = \vec{X}_{i,G} + F*(\vec{X}_{best,G} - \vec{X}_{i,G}) + F*(\vec{X}_{r_1^i,G} - \vec{X}_{r_2^i,G}) + F*(\vec{X}_{r_3^i,G} - \vec{X}_{r_4^i,G}) \quad (11)$$

In these mutation strategies, there are two aspects differentiate one DE from another. One is the variance type ("rand"," best"), the other is the numbers of the disturbance vector (one or two). The DE/rand achieved good results in solving unimodal and separable functions. The difference between DE/rand/1 and DE/rand/2 is the number of disturbance vector. DE/rand/2 gets better optimization ability in multimodal and non-separable problems than DE/rand/1. The greedy variants ("best" or "current-to-best") introduce the best food source (solution) to the current population. The fast convergence speed can be achieved in solving optimization especially multimodal problems. But, on the other hand, the increasing of convergence speed may lead to some problems such as premature convergence in solving multimodal problems. DE/current-to-best which utilize not only the current solution (food source), but also the best solution reduce the chance premature relative to the DE/best. From the previous studies, it cannot be expected that an algorithm can find optimized solution for any type problems. Every mutation strategies in DE adapt to one class problem.

In basic ABC, employed bees find new food source which is similar to mutation strategies in DE. Inspired by this, we will import the six mutation strategies to the basic ABC algorithm and make analysis and comparisons systemati-cally and completely. The improved ABC algorithms based on DE are described as follows:

"ABC/rand/1:" $$\vec{V}_{i,m} = \vec{X}_{r_1,m} + \varphi*(\vec{X}_{r_2,m} - \vec{X}_{r_3,m}) \quad (12)$$

"ABC/best/1:" $\vec{V}_{i,m} = \vec{X}_{best,m} + F*(\vec{X}_{r_1,m} - \vec{X}_{r_2,m})$ (13)

"ABC/current-to-best/1:" $\vec{V}_{i,m} = \vec{X}_{i,m} + \varphi*(\vec{X}_{best,m} - \vec{X}_{i,m}) + \varphi*(\vec{X}_{r_1,m} - \vec{X}_{r_2,m})$ (14)

"ABC/rand/2:" $\vec{V}_{i,m} = \vec{X}_{r_1,m} + \varphi*(\vec{X}_{r_2,m} - \vec{X}_{r_3,m}) + \varphi*(\vec{X}_{r_4,m} - \vec{X}_{r_5,m})$ (15)

"ABC/best/2:" $\vec{V}_{i,m} = \vec{X}_{best,m} + \varphi*(\vec{X}_{r_1,m} - \vec{X}_{r_2,m}) + \varphi*(\vec{X}_{r_3,m} - \vec{X}_{r_4,m})$ (16)

"ABC/current-to-best/2:" $\vec{V}_{i,m} = \vec{X}_{i,m} + \varphi*(\vec{X}_{best,m} - \vec{X}_{i,m}) + \varphi*(\vec{X}_{r_1,m} - \vec{X}_{r_2,m}) + \varphi*(\vec{X}_{r_3,m} - \vec{X}_{r_4,m})$ (17)

In order to test and verify the ability (12) – (17) in solving unimodal and multimodal problems, we will make multi-group experiments for different type of test functions.

4. Experiment Arrangement and Results

4.1. Test Functions

In this section, a set of basic test functions (unimodal and multimodal) will be used to test six improved ABC algorithms (12)-(17). The testing functions are listed in table1.

Table 1. Basic Function

Function	Global min	Search range	Formula
Shpere(UM)	0	[-100,100]^D	$f(x)=\sum_{i=1}^{n} x_i^2$
Rosenbrock(UM)	0	[-2.048,2.048]^D	$f(x)=\sum_{i=1}^{n-1}[100(x_{i+1}-x_i^2)^2+(x_i-1)^2]$
Ackley(MM)	0	[-32.768,32.768]^D	$f(x)=-20\exp(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n} x_i^2}) - \exp(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi x_i))+20+e$
Griewank(MM)	0	[-600,600]^D	$f(x)=\frac{1}{4000}\sum_{i=1}^{n} x_i^2-\prod_{i=1}^{n}\cos(\frac{x_i}{\sqrt{i}})+1$
Weierstrass(MM)	0	[-0.5,0.5]^D	$f(x)=\sum_{i=1}^{D}(\sum_{k=0}^{k\max}[a^k\cos(2\pi b^k(x_i+0.5))]) - D\sum_{k=0}^{k\max}[a^k\cos(a^k\cos(2\pi b^k 0.5))], a=0.5, b=3, k\max=20$
Rastrigin(MM)	0	[-5.12,5.12]^D	$f(x)=\sum_{i=1}^{n}[x_i^2-10\cos(2\pi x_i)+10]$
Schwefel(MM)	0	[-500,500]^D	$f(x)=n*418.982887-\sum_{i=1}^{n}(x_i\sin(\sqrt{\lvert x_i\rvert}))$

4.2. Parameter Settings and Arrangement

The experiments will be divided into three parts. First, aimed at these testing functions in table1, the population size was 10, and the maximum function evaluations was 30,000 for 10-dimension. All experiments were repeated 30 times. In order to compare our algorithms to other existed algorithms based swarm intelligence, what the parameter setting we used was the same as [11]. The experimental results are listed in Table2. Second, the comparisons for convergence ability among the six improved ABC were shown in Figure 2 – Figure 13. Finally, in order to investigate the adaptability for every improved ABC algorithms, the ability for optimizing every testing function by all six improved ABC algorithms will be showed in Figure 14 – Figure 21.

4.3. Experimental Results

In table 2, we compare the optimization ability of the six improved ABC algorithms for unimodal and multimodal problems. For unimodal problem, ABC/…/2 outperforms better mean and standard deviation than ABC/…/1. Under the same disturbance vector, the introduction of best information can get better convergence results than "rand" strategy. We also find that there exist certain risks getting into premature for the "best" strategy. The experimental results also show that ABC/current-to-best performs better than other ABC variants on almost multimodal function.

For Unimodal function, the graph of convergence can be drawn in Fig 1-Fig7. Under the same strategy ("best" or "rand"), the two disturbance vector will be achieve faster convergence speed than one disturbance vector. The "current-to-best" strategy which is incorporating the current solution acquires stable result in multimodal experiments. From the view of the final outcome in solving multimodal, the "current-to-best" strategy does better than the other strategies in most cases. The "best" strategy is easy trapped into local minimum value while things get better if we increase the number of disturbance vector.

In Fig 15.-Fig 21, we adopt the different variants of ABC to optimize different testing functions. For unimodal function, the introduction of the best solution makes the direction of population move faster toward to the optimized value. For Weierstrass and Rastrigin function, the "ABC/rand/2","ABC/best/2","ABC/current-to-best/1",and "ABC/current-to-best/2" have reached the minimal value.

Table 2. NP=10, D=10, Max.Eval =30, 000, runtime=30, limit=200, UM: Unimodal; MM: Multimodal

		UM Shpere	UM Rosenbrock	MM Ackley	MM Griewank	MM Weierstrass	MM Rastrigin	MM Schwefel
Basic ABC	Mean	7.09e-017	2.08e+000	4.58e-016	1.57e-002	9.01e-006	1.61e-016	7.91e+000
	Std	4.11e-017	2.44e+000	1.76e-016	9.06e-003	4.61e-005	5.20e-016	2.95e+000
MABC[11]	Mean	7.04e-017	4.42e-001	3.32e-016	1.52e-002	1.18e-016	1.14e-007	3.96e+000
	Std	4.11e-017	8.67e-001	1.84e-016	1.28e-002	6.38e-016	6.16e-007	2.13e+000
ABC/best/1	Mean	1.46e-002	9.82e+000	4.08e-001	1.59e-001	5.44e-002	1.31e+000	1.10e+002
	Std	4.17e-002	1.57e+001	6.72e-001	2.03e-001	6.19e-002	1.40e+000	1.32e+002
ABC/rand/1	Mean	4.28e-002	5.25 e+000	3.33e-001	1.95e-001	4.76e-002	1.52e+000	1.04e+002
	Std	1.93e-001	9.02 e+000	5.74e-001	4.01e-001	8.45e-002	1.22e+000	1.17e+002
ABC/current-to-best/1	Mean	5.39e-124	7.87e-001	8.5857e-015	9.31e-003	**0**	**0**	**1.25e-004**
	Std	2.69e-123	1.57 e+000	1.8853e-015	6.72e-003	0	0	4.45e-004
ABC/best/2	Mean	**4.02e-156**	2.24 e+000	6.2172e-015	2.42e-002	**0**	3.32e-002	1.27e-004
	Std	2.20e-155	2.26 e+000	1.8067e-015	2.28e-002	0	1.81e-001	2.30e-003
ABC/rand/2	Mean	1.38e-148	2.66e-001	7.7568e-015	9.82e-003	**0**	**0**	2.43e+001
	Std	7.60e-148	3.88e-001	9.0135e-016	7.51e-003	0	0	5.71e+001
ABC/current-to-best/2	Mean	2.84e-112	**1.0e-001**	7.8752e-015	**7.23e-003**	**0**	**0**	2.20e-001
	Std	1.51e-111	8.23e-002	1.4703e-015	9.01e-003	0	0	1.20e+000

Figure 2 – Figure 7. Convergence graph for Unimodal functions using (12)-(17)

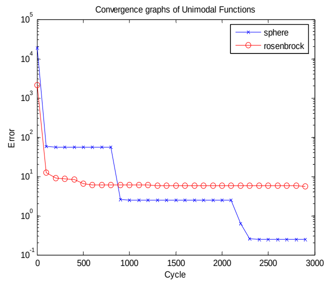

Figure 2. ABC/best/1

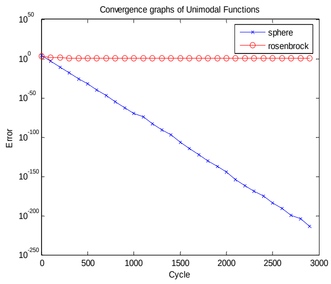

Figure 3. ABC/best/2

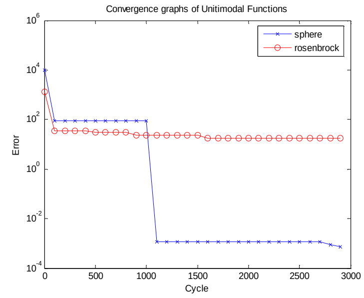

Figure 4. ABC/rand/1

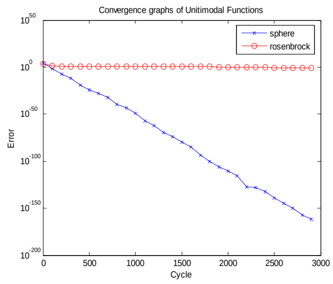

Figure 5. ABC/rand/2

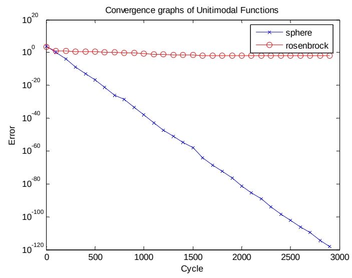

Figure 6. ABC/current-to-best/1

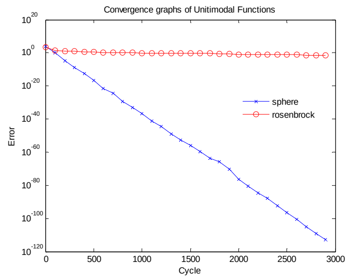

Figure 7. ABC/current-to-best/2

Figure 8 – Figure 13. Convergence graph for Multimodal functions using (12)-(17)

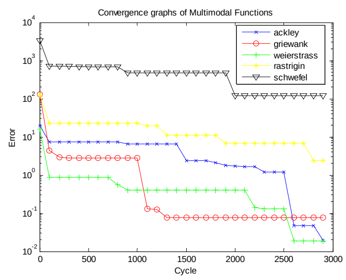

Figure 8. ABC/best/1

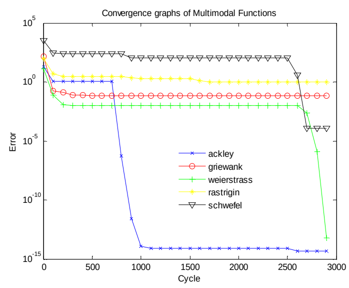

Figure 9. ABC/best/2

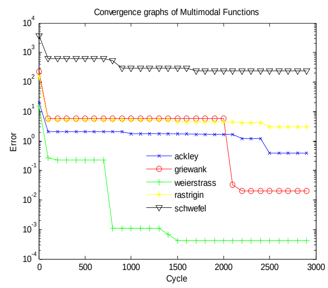

Figure 10. ABC/rand/1

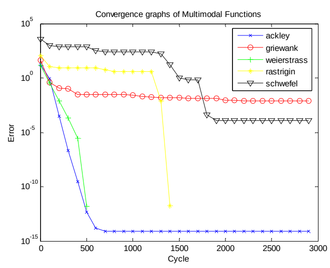

Figure 11. ABC/rand/2

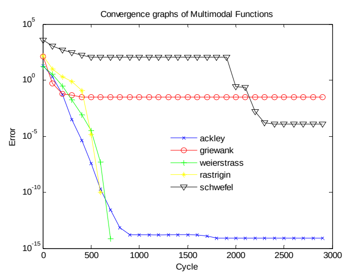

Figure 12. ABC/current-to-best/1

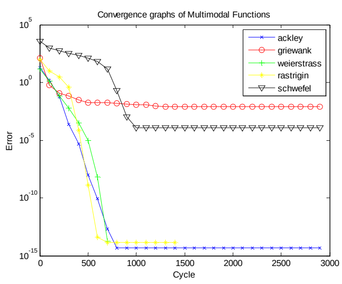

Figure 13. ABC/current-to-best/2

Figure 15 – Figure 21 Convergence ability for every testing function by all six variants of ABC

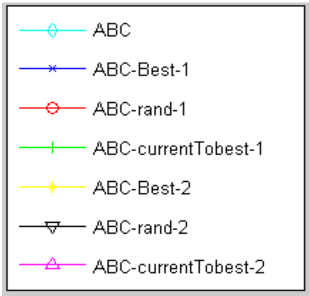

Figure 14. Legend used in Figure 15 – Figure 21

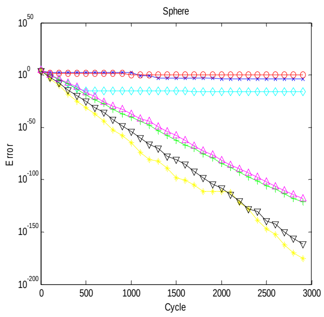

Figure 15. Sphere (UM)

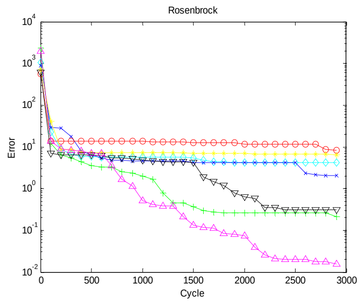

Figure 16. Rosenbrock (UM)

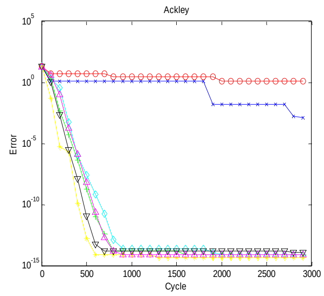

Figure 17. Ackley(MM)

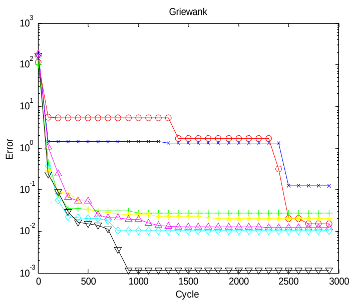

Figure 18. Griewank(MM)

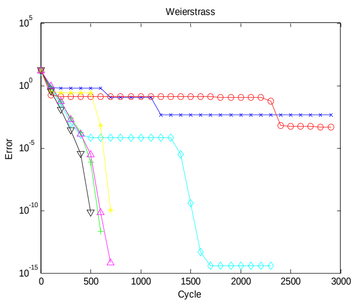

Figure 19. Weierstrass (MM)

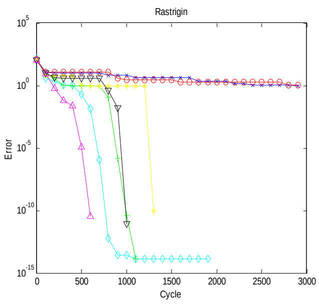

Figure 20. Rastrigin (MM)

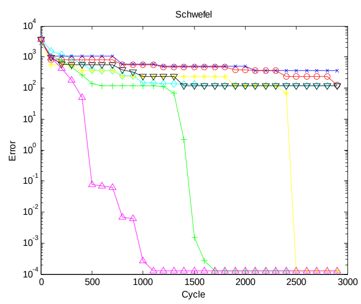

Figure 21. Schwefel (MM)

5. Conclusion

In this work, we investigated roundly the performance of variants of ABC based on the mutation strategies used in DE. Besides comparing with that already exists, we especially analyze the ability of convergence using the different improved ABC algorithms. From the results, we can conclude that the variants of ABC algorithms can enhance the optimization abilities for different type problems.

Acknowledgments

This work is supported by National Nature Science Foundation of China (Grant No. 61075049), the Excellent Young Talents Foundation Project of Anhui Province (Grant No.2011SQRL018), the Youth Foundation of Anhui University (Grant No. KJQN1015), the University Natural Science Research Project of Anhui Province (Grant No. KJ2012B038), the Provincial Natural Science Foundation of the Anhui Higher Education Institutions of China (Grant No. KJ2013A009).

References

[1] Dorigo M, Maniezzo V, Colorni A. “Ant system: optimization by a colony of cooperating agents”. *IEEE Trans on Systems, Man, and Cybernetics, Part B: Cybernetics.* 1996; 26(1): 29-41.
[2] RC Eberhart and J Kennedy. *“A new optimizer using particle swarm theory”.* in Proc. 6th Int. Symp. Micromachine Human Sci, Nagoya, Japan. 1995: 39-43.
[3] J Kennedy and RC Eberhart. *“Particle swarm optimization”.* in Proc. IEEE Int. Conf. Neural Networks. 1995: 1942–1948.
[4] Storn R, Price K. “Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces”. *Journal of global optimization*. 1997; 11(4): 341-359.
[5] Mezura-Montes E, Velázquez-Reyes J, Coello Coello CA. “*A comparative study of differential evolution variants for global optimization*”. Proceedings of the 8th annual conference on Genetic and evolutionary computation. 2006: 485-492.
[6] S Das and PN Suganthan. *“Differential evolution—A survey of the state-of-the-art”. IEEE Trans. Evol. Comput*. 2011; 15(1): 4-31.
[7] E Mezura-Montes, J Velázquez-Reyes, CA Coello. “*A comparative study of differential evolution variants for global optimization*”. in Proc. GECCO. 2006: 485-492.
[8] D Karaboga. “*An idea based on honeybee swarm for numerical optimization*”. *Technical Report TR06*, Erciyes University, Engineering Faculty, Computer Engineering Department; 2005.
[9] D Karaboga, B Basturk. “A comparative study of artificial bee colony algorithm”, *Applied Mathematics and Computation*. 2009; 214: 108-132.
[10] E Bonabeau, M Dorigo, G Theraulaz, “*Swarm Intelligence: From Natural to Artificial System*”. New York, NY: Oxford University Press; 1999.
[11] B Akay, D Karaboga. “A modified artificial bee colony algorithm for real-parameter optimization”. *Information Sciences*. 2012; 192: 120-142.
[12] GP Zhu, S Kwong, “Gbest-guided artificial bee colony algorithm for numerical function optimization”. *Applied Mathematics and Computation*. 2010; 217(7): 3166-3173.
[13] Weifeng Gao, Sanyang Liu. “Improved artificial bee colony algorithm for global optimization”. *Information Processing Letters*. 2011; 111: 871-882.
[14] Weifeng Gao, Sanyang Liu, Lingling huang, “A global best artificial bee colony algorithm for global optimization”. *Journal of Computational and Applied Mathematics*. 2012; 236: 2741-2753.
[15] Parvaneh Mansouri. The Modify Version of Artificial Bee Colony Algorithm to solve Real Optimization problems. *International Journal of Electrical and Computer Engineering*. 2012; 2(4): 473-480.
[16] Vimal Nayak, Haresh A Suthar, Jagrut Gadit. Implementation of Artificial Bee Colony Algorithm. *IAES International Journal of Artificial Intelligence*. 2012; 1(3): 112-120.