International Journal of Electrical and Computer Engineering (IJECE)
Vol.2, No.4, August 2012, pp. 473~480
ISSN: 2088-8708

The modify version of artificial bee colony algorithm to solve real optimization problems

B. Asady*, P. Mansouri*,**, N. Gupta**
* Department of Mathematics, Arak Branch, Islamic Azad University, Arak-Iran
** Department of Computer Science,Delhi University, Delhi,India

Article Info

Article history:
Received May 9, 2012
Revised Jul 10, 2012
Accepted Jul 26, 2012

Keyword:
ArtificialBee Colony algorithm;
Optimization problem;
Iterative method.

ABSTRACT

The Artificial Bee Colony(ABC) algorithm is one of the best applicable optimization algorithm. In this work, we make some modifications to improve the ABC algorithm based on convergence speed of solution. In order to, we add some conditions to selected food sources by bees. So, if solution have been enough near to optimal solution, then further search exist around the food sources. That, this is near to optimal solution because, we can replace lower and upper bounds of food sources with smaller values relate to last search. Therefore, the new search is near to optimal solution and after some iteration, optimal solution achieves. Finally, we illustrate convergence speed of the MABC algorithm that is faster than ABC algorithm. There are some examples.

Corresponding Author:
P. Mansouri,
Departement of Mathematics,
Eslamic Azad Universir, Branch Arak,
P.O.Box 38135-567, Arak, IRAN
Email: pmansouri393@yahoo.com, p-mansouri@iau-arak.ac.ir

1. INTRODUCTION

Artificial Bee Colony (ABC) is one of the most recently defined algorithms by Dervis Karaboga in 2005, motivated by the intelligent behavior of honey bees [2, 10]. It is as simple as Particle swarm optimization (PSO) and Differential evolution (DE) algorithms, Genetic Algorithm (GA)[1], biogeography based optimization (BBO), and uses only common control parameters such as colony size and maximum cycle number. ABC as an optimization tool, provides a population-based search procedure in which individuals called foods positions are modified by the artificial bees with time and the bee's aim is to discover the places of food sources with high nectar amount and finally the one with the highest nectar. In ABC system, artificial bees fly around in a multidimensional search space and some (employed and onlooker bees) choose food sources depending on the experience of themselves and their nest mates, and adjust their positions. Once all onlookers have selected their food sources, each of them determines a new neighboring food source of its selected food source and computes its nectar amount. Providing that this amount is higher than that of the previous one, and then the bee memorizes the new position and forgets the old one. The employed bee becomes a scout bee when the food source which is exhausted by the employed and onlooker bees is assigned as abandoned. In other words, if any solution cannot be improved further through a predetermined number of cycles which is called limit parameter, the food source is assigned as an abandoned source and employed bee of that source becomes a scout bee. Thus, ABC system combines local search methods, carried out by employed and onlooker bees, with global search methods, managed by onlookers and scouts, attempting to balance exploration and exploitation process. Also, Karaboga and Basturk have compared the performance of the ABC algorithm with other works such as GA, DE and PSO methods on unconstrained problems[3]. Although, ABC is a robust, easy and flexible algorithm, but similar to other evolutionary algorithm have some challenges and problems. For example, accelerating of convergence speed is one of the important goal in ABC research. But, convergence speed of this method is typically slower than those of representative population-based algorithms [11]. Some researchers find application of ABC algorithm to solve hard problems and clustering. P. Mansouri, B. Asady and N. Gupta in 2011[12] introduced a novel iteration method by using ABC algorithm for solve hard problems. D. Karaboga and C. Ozturkin in 2010[13] applied the ABC algorithm fuzzy clustering to classify different data sats: Cancer, Diabetes,and Heart from UCI database, a collection of classification benchmark problems. G. Wei-feng , L. San-yang in 2011[14] introduced the new search mechanism together with the proposed initialization makes up the modified ABC, which excludes the probabilistic selection scheme and scout bee phase. They proposed the new search mechanism which introduces the selective probability P to balance the exploration of the solution search equation:

$$v_{ij} = z_{ij} + \theta_{ij}\left(z_{ij} - \mathbf{z_{kj}}\right) \tag{1}$$

where θ_{ij} is a random number between [−1, 1] and the exploitation of the modified solution search equation:

$$v_{ij} = z_{bestj} + \theta_{ij}\left(z_{r_1 j} - z_{r_2 j}\right) \tag{2}$$

where the indices r_1 and r_2 are mutually exclusive integers randomly chosen from $1, 2, \dots, \mathbf{n}$, and different from the base index i; X_{best} is the best individual vector with the best fitness in the current population and $j = 1, 2, \dots, n$ and randomly chosen indexes. Their idea, the solution search dominated by Eq.1 is random enough for exploration. In other words, the solution search equation described by Eq.1 is good at exploration but poor at exploitation. However, according to Eq.2, In ABC, v_{ij} can drive the new candidate solution only around the best solution of the previous iteration. Therefore, the proposed solution search equation described by Eq.2 can increase the exploitation of ABC. With some example, we show convergence speed of this method is slow. In this work, we introduce modified ABC (MABC) method that is faster than ABC algorithm by adding some limitations to ABC algorithm respect to convergency. In the section 2, ABC algorithm summarize. In section 3, MABC algorithm for improve ABC algorithm present to find global optimal solution of optimization problems. Then computing time (convergence speed) of the MABC and ABC algorithms compared in section 4. Finally, conclusion shows in the last section.

2. ARTIFICIAL BEE COLONY ALGORITHM

In the ABC model, the colony consist three groups of bees: employed bees, onlookers and scouts. In the ABC algorithm, the number of employed bees is equal to the number of food sources which is also equal to the number of onlooker bees. There is only one employed bee for each food source whose first position is randomly generated. At each iteration of the algorithm, each employed bee determines a new neighboring food source of its currently associated food source and computes the nectar amount of this new food source by Equation (1). If the nectar amount of this new food source is higher than that of its currently associated food source, then this employed bee moves to this new food source, otherwise it continues with the old one. After all employed bees complete the search process, they share the information about their food sources with onlooker bees. An onlooker bee evaluates the nectar information taken from all employed bees and chooses a food source with a probability related to its nectar amount by Equation:

$$p = \frac{fit_i}{\sum_{k=1}^{n} fit_k} \tag{3}$$

where $\mathbf{fit_1}$ is the fitness value of the solution i which is proportional to the nectar amount of the food source in the position i and n is the number of food sources which is equal to the number of employed bees. This method, known as roulette wheel selection method, provides better candidates to have a greater chance of being selected. Once all onlookers have selected their food sources, each of them determines a new neighboring food source of its selected food source and computes it's nectar amount Providing that this amount is higher than that of the previous one, and then the bee memorizes the new position and forgets the old one. The employed bee becomes a scout bee when the food source which is exhausted by the employed and onlooker bees is assigned as abandoned. In other words, if any solution cannot be improved further through a predetermined number of cycles which is called limit parameter, the food source is assigned as an abandoned source and employed bee of that source becomes a scout bee. In that position, scout generates randomly a new solution by Equation:

$$z_i i^{\uparrow} j = +rend(0,1)(z_i max^{\uparrow} j - z_i min^{\uparrow} j) \quad (4)$$

where j is determined randomly which is different from i and assume that zi is the abandoned source and $j = 1, 2, \ldots, D.$ where D is the solution vector, the scout discovers a new food source which will be replaced with zi. The employed bee whose food source has been abandoned becomes a scout and starts to search for finding a new food source. Onlookers watch the dances of employed bees and choose food sources depending on dances. Based on the above explanation of initializing the algorithm population, employed bee phase, probabilistic selection scheme, onlooker bee phase and scout bee phase, the pseudo-code of the ABC algorithm is given bellow:

2 .1. Algorithm .1 (Abc Algorithm)

01. Initialize population with random solutions.
02. Evaluate fitness of the population.
03. While (stopping criterion not met) Forming new population.
04. Select sites for neighborhood search.
05. Recruit bees for selected sites (more bees for best sites) and evaluate fitnesses.
06. Select the fittest bee from each patch.
07. Assign remaining bees to search randomly and evaluate their fitnesses.
08. End While.

In first step, the algorithm starts with the scout bees (n) being placed randomly in the search space. In step 2, the fitnesses of the sites visited by the scout bees are evaluated. In step 4, bees that have the highest fitnesses are chosen as "selected bees" and sites visited by them are chosen for neighborhood search. Then, in steps 5 and 6, the algorithm conducts searches in the neighborhood of the selected sites, assigning more bees to search near to the best sites. The bees can be chosen directly according to the fitnesses associated with the sites they are visiting. Alternatively, the fitness values are used to determine the probability of the bees being selected. Searches in the neighborhood of the best sites which represent more promising solutions are made more detailed by recruiting more bees to follow them than the other selected bees. Together with scouting, this differential recruitment is a key operation of the bees algorithm. However, in step 6, for each patch only the bee with the highest fitness will be selected to form the next bee population. In nature, there is no such a restriction. This restriction is introduced here to reduce the number of points to be explored. In step 7, the remaining bees in the population are assigned randomly around the search space scouting for new potential solutions. These steps are repeated until a stopping criterion is met. At the end of each iteration, the colony will have two parts to its new population, those that were the fittest representatives from a patch and those that have been sent out randomly.

3. THE MODIFICATION OF ABC ALGORITHM (MABC)

Consider the optimization hard problem as follows:

$$\min f(x), \qquad x = (x_1, x_2, \ldots, x_n) \in R^n$$

Where domains of variables defined by their lower and upper bounds:

$$lb_i \le x_i \le ub_i$$

By modify the ABC algorithm(MABC), we obtain the iteration method to find global optimal solution of given hard problem that convergence speed is faster than ABC algorithm, with respect to arbitrary accuracy. In MABC same as ABC algorithm, the position of a food source represents a possible solution to the optimization problem and the nectar amount of a food source corresponds to the quality (fitness) of the associated solution. The number of the employed bees or the onlooker bees is equal to the number of solutions in the population. At the first step, the MABC generates a randomly distributed initial population P of n solutions (food source positions), where n denotes the size of population. Each solution $x_i, (i = 1, 2, \ldots, n)$ is a D-dimensional vector. Here, D is the number of optimization parameters. We added two limitations to ABC algorithm, one limitation for changing initial interval that includes solution to obtain small size interval near to global optimal solution as possible and one limitation for convergence condition with respect to arbitrary accuracy. Details of MABC algorithm are as follows:

3.1 Algorithm 2.(MABC Algorithm)

01. Initialize the population of optimal solution $X_i = (x_{i1}, x_{i2}, \dots, x_{in}) \in R^n$. Let X^n represent the i^{th} food source in the population, and each food source is generated by ABC algorithm as follows:

$$x_{ij} = x_{minj} + rand(0,1)(x_{maxj} - x_{minj}), \qquad j = 1,2,\dots,n$$

02. Construct initial employed bee colony solutions by using greedy randomized, adaptive search c heuristic (GRAH),

03.Each employed bee goes to a food source in her memory and determines a neighbor source, then evaluates its nectar amount and dances in the hive(evaluate fitness value for each bee).

04. I=0(number of iteration)

05. Repeat,.Until N=Employed Bee

06. N=0,

07. Repeat, Until I=MaxCycles

08. Each onlooker watches the dance of employed bees and chooses one of their sources depending on the dances, and then goes to that source. After choosing a neighbor around that, she evaluates its nectar amount.

09.For each bee employed, replace initial lower(lb) and upper(ub) bounds to closer bounds to optimal solution as follows:

10. If the values of optimization problem in the best food source in iteration i and i+1 $(1 < i <<$ Maxiteration) get close together,

11 Then, we can say, by replacing initial lower(lb) and upper(ub) bounds of sources to smaller sizes and closer to optimal solution as possible as following,

$$\begin{aligned}
&if\ ((Cycle \geq 2) and\ (GlobalMins \geq last\ GlobalMins)),\\
&\quad GlobalMins = last\ GlobalMins,\\
&\quad lb = lb - abs(0.5 - rand),\\
&\quad ub = ub + abs(0.5 - rand),\\
&\quad if(lb < initial\ lb), lb = initial\ lb; if(ub > initial\ ub), \qquad ub = initial\ ub,
\end{aligned}$$

Thus, domain of search will be smaller and consequently convergence speed of MABC method will be faster. Last limitation guarantees that, we don't loose convergence's domain.

12. Else, I= I+1.

13. N=N+1.

14. Abandoned food sources are determined and are replaced with new food sources discovered by scouts. The best food source found so far is registered(best feasible onlooker found and replace with best solution)

15. Check how much these food sources are near to best food source(with arbitrary accuracy),

16. If GlobalMins(k+1)-GlobalMins(k) $<= 1.e^{-t}$

17. Then, GlobalMin= GlobalMins(Cycle), $1 \leq k \leq Maxcycles, \quad \varepsilon = 1.e^{-t}, \quad t >> 1,$

18. UNTIL (requirements are me

where GlobalMin is values of optimization problem to best value of sources.

By using ABC algorithm, a randomly distributed initial population (initial random value of global optimal solution) is generated. After that, the population is subjected to repeat the iteration of the search processes of the employed, onlooker and scout bees respectively. Find the best feasible onlooker, replace with the best solution. Since the ABC algorithm is one of the convergence iterative method, then in cycles k and k+1 (1 $\leq$ k << MaxCycles), during of the search processes, initial bound intervals of the parameters (parameters relate to optimized) was reduced to small size as possible. With respect to convergency, by increasing lower and decreasing upper bounds $[lb_j, ub_j]$, that includes the parameter x_j, the optimal solution obtained as follows:

temp1= ub, temp2=lb, $ub1_j = \dfrac{GlobalParams(k,j) + ub_j}{2}$

$$lb1_j = \frac{GlobalParams(k,j) + lb_j}{2},$$

where the GlobalParams (k,j) is the best solution in Cycle k.

$lb\mathbf{1}_j \le Globalparams(k,j) \le ub\mathbf{1}_j$.

GlobalParams(1,j)= x_j , j=1,2,...,n $(1 \le k << MaxCycles)$, and

$lb1_j$ *and* $ub1_j$ *are lower and upper bounds of* x_j respectively.

$ub = \max(ub1)$, $lb = \min(lb1)$. *It is clear that* $lb \le lb1_j \le ub1_j \le u_j$.

if $temp1 = ub$ *then* $ub = ub - abs(0.5 - rand)$,

and

if $temp2 = lb$, *then* $lb = lb + abs(0.5 - rand)$. So that, if the solution in cycle $k\,(1 << k \le MaxCycles$ has been close to global optimal solution and satisfy to convergence condition with arbitrary accuracy as *follows:*

$$\left|f^{(k+1)}(x) - f^{k}(x)\right| \le \varepsilon_1 \text{ and } (ub - lb) \le \varepsilon_2.$$

Initil bounds of the parameters (parameters relate to optimized)reduced to smaller sizes as possible and global optimal solution founded and finally algorithm terminates. In the next section, we illustrate the modify version of ABC algorithm (MABC) and show that converges speed of MABC algorithm is faster than ABC algorithm.

4. THE EXAMPLES

In this section, some examples are propose to illustrate MABC algorithm and compares with ABC algorithm. Also we compare MABC in this paper and Gao's algorithm that introduced by W.Gao in 2011 [13]. In order to, we consider the four scalable benchmark functions as shown in follows:

Table 1. Benchmark functions

Benchmark function	Functions	search range	Min optimal solution
Rastrigin	$\sum_{i=1}^{D}[x_i^2 - 10\cos[2\pi x_i + 10]]$	[-15,*15]*	0
Griwank	$\frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D}\left(\cos x_i / \sqrt{i}\right) + 1$	[600,600]	0
Rosenbrock	$\sum_{i=1}^{D}[100(x_i^2 - x_{i+1})^2 + (1 - x_i)^2]$	[-15 15]	0
Sphere	$\sum_{i=1}^{D} x_i^2$	[-15,15]	0

We obtain global minimum values of above benchmark functions by using MABC algorithm and ABC algorithm same as Table2(D=1).

Table 2: Comparative results of performance MABC and ABC algorithms:

Optimization	MABC	ABC function	Iteration
Rastrigin	1.04961e-5	2.70487e-2	30
Griewank	3.38547e-8	2.12422e-3	30
Rosenbrock	6.88361e-3	1.411e-1	30
Sphere	1.24433e-9	2.63808e-4	30

As table 2. shows, results of MABC method is better than ABC(Figures 1-4).

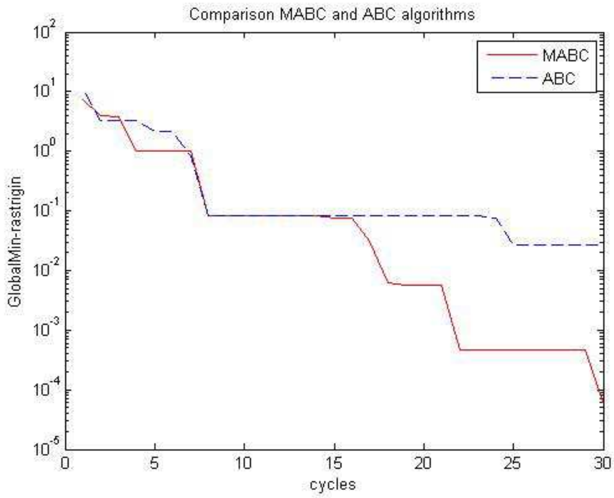

Figure 1. Compare results of MABC and ABC algorithms on optimization problem Rastrigin.

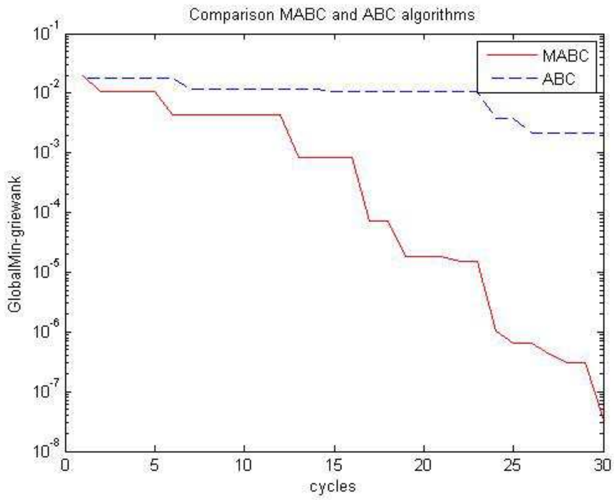

Figure 2. Compare results of MABC and ABC algorithms on optimization problem Griewank.

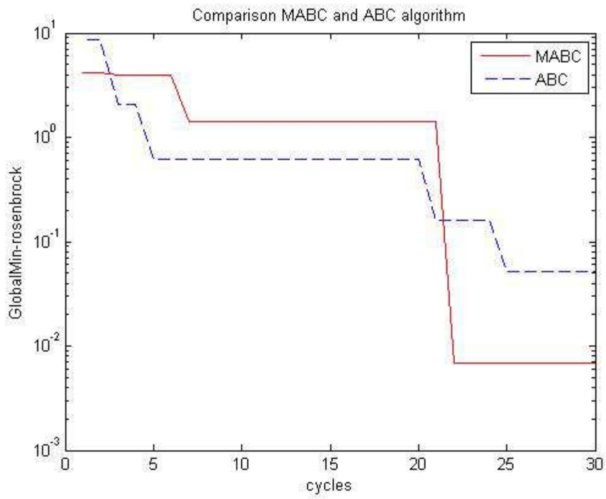

Figure 3. Compare results of MABC and ABC algorithms on optimization problem Resonberge .

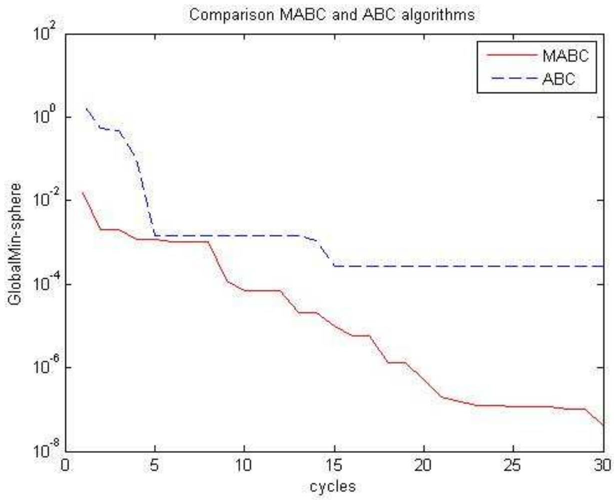

Figure 4. Compare results of MABC and ABC algorithms on optimization problem Sphere.

Comparison of proposed algorithm, ABC algorithm and Gao's algorithm by using benchmark functions of Table.1 when Dim=30 is as follows:

Table 3. Comparative results of performance MABC, ABC and Gao's algorithms

Optimization function	(MABC, Iteration)	(ABC,Iteration)	(Gao,Iteration)	Dim
Rastrigin	(1.04961e-5, 2e+3)	(1, 3e+15)	(0, 8e+4)	30
Griewank	(4.10783e-015, 2e+3)	(1.e-5, 12e+14)	(1.e-15, 12e+14)	30
Rosenbrock	(7.0e-4, 1900)	(7.93e-1, 2e+10)	(1.73e-1,?)	30
Sphere	(2.73264e-017, 2133)	(4.17e-16, 4e+5)	(9.43e-32, 15e+05)	30

5. CONCLUSION

The capability of the ABC algorithm for hard optimization problems was investigated through the performance of several experiments on well-known test problems. In this paper, we present an improved ABC algorithm with adding some limitations. when onlookers chose best food source(best solution in i cycle ($1 \leq i \ll MaxCycle$), by some modifications, the MABC algorithm replace initial bounds of optimal solution to smaller sizes as possible and convergence speed of algorithm will be increase. We solved some well known hard problems. Result of comparison convergence speed between our algorithm and ABC algorithm at the same iteration time, at the Tables 3 and 4 show our algorithm is faster and then, complexity is less than the ABC algorithm. Also comparison of our algorithm with Gao's algorithm, with respect to accuracy of solution shows our algorithm is faster. So it is better, instead of ABC algorithm, we choose MABC algorithm to solving hard problems.

REFRENCES

[1] J.H.Holland," Adaptation in Natural and Artificial Systems," University of Michigan Press, Ann Arbor, MI (1975).
[2] D. Karaboga, B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial Bee Colony (ABC) algorithm," Journal of Global Optimization, vol. 39, no. 3, Feb. 2007, pp. 459–471.
[3] D. Karaboga, B. Basturk, "On the performance of Artificial Bee Colony (ABC) algorithm," Applied Soft Computing, vol. 8, January 2008, pp. 687–697.
[4] D. Karaboga, B. Basturk, C. Ozturk, "Artificial Bee Colony (ABC) Optimization Algorithm for Training Feed-Forward Neural Networks," Springer, 2007, pp. 318–329.
[5] A. Singh, "An Artificial Bee Colony algorithm for the leaf-constrained minimum spanning tree problem," Applied Soft Computing, vol.9, pp. 625–631, March 2009.
[6] N. Karaboga, "A new design method based on Artificial Bee Colony algorithm for digital IIR filters," Journal of The Franklin Institute, vol. 346, May 2009,pp. 328–348.
[7] C. Zhang, D. Ouyang, J. Ning, "An Artificial Bee Colony approach for clustering," Expert Systems with Applications, vol. 37,,July 2010pp 4761-4767.

[8] D. Karaboga, B. Basturk, “Artificial Bee Colony (ABC) Optimization algorithm for Solving Constrained Optimization Problems”, 12th Foundations of fuzzy logic and soft computing, Cancun, Mexico, 2007pp.789-798.
[9] A. Baykaso, L. Özbakýr, P.Tapkan,"Artificial Bee Colony Algorithm and Its Application to Generalized Assignment Problem", University of Gaziantep, Department of Industrial Engineering Erciyes University, Department of Industrial Engineering Turkey.
[10] D. Karaboga," An Idea Based On Honey Bee Swarm for Numerical Optimization," Technical Report-TR06,Erciyes University, Engineering Faculty, Computer Engineering Department 2005.
[11] R. Storn , K. Price,"Differential evolution–Asimple and efficient heuristic for global optimization over continuous spaces", Journal of Global Optimization23, ,2010pp.689–94 .
[12] P. Mansouri, B. Asady, N. Gupta,"A Novel Iteration Method for solve Hard Problems (Nonlinear Equations) with Artificial Bee colony algorithm", Word Academy of Science, Engineering and Technology 59,,2011 , pp.594-96
[13] D. Karaboga, C. Ozturk,"Fuzzy Clustering With Artificial Bee Colony Algorithm", Scientific Research and Essays, Vol.5(14), pp. 1899-1902,18 july, 2010.
[14] Wei-feng. Gao, L. San-yang,"A modified artificial bee colony algorithm",Journal of Computers and Operations,vol. 392012 (Available online 25 June 2011), pp687-697.