Multi-Population Artificial Bee Colony (MPABC) Algorithm for Numerical Optimization

To cite this article: Ming Zhao and Peihong Wang 2018 *IOP Conf. Ser.: Mater. Sci. Eng.* **452** 032003

View the [article online](#) for updates and enhancements.
Multi-Population Artificial Bee Colony (MPABC) Algorithm for Numerical Optimization

Ming Zhao1, *, Peihong Wang2
1 Research Institute of Yunnan Power Grid Co. Ltd., Kunming, Yunnan, China.
2 School of energy & environment, Southeast University, Nanjing, Jiangsu Province 210096, China.

*Corresponding author e-mail: zming64@163.com

Abstract. This paper aims to propose a variant artificial bee colony algorithm (ABC), called multi-population artificial bee colony (MPABC) algorithm so as to optimize numerical functions with single and/or multiple solutions where the global optimization can be achieved. In MPABC, the solution space (i.e., food source) is partitioned into some subspaces in which a subpopulation of bees are parallel produced and employed to search the local optimization. Among these local optimizations, the ones with highest adaptability are taken as the global optimizations in each iteration step, and the corresponding local solutions are thus the global solutions. With a reasonable partition of solution space, all the global optimization and all the associated global solutions can be found by using MPABC. This property can not be committed by the ABC, as the ABC is interested in finding the global optimization with one solution in one running time. In addition, the MPABC holds higher abilities on convergence speed and accuracy than the ABC. Some experiments were conducted with some numerical functions so as to validate such conclusions.

Keywords: swarm intelligence; artificial bee colony algorithm; numerical optimization, multi-population artificial bee colony; multiple solutions

1. Introduction

Recently proposed ABC algorithm by Karaboga in 2005 is inspired by the intelligent behavior of real honey bees swarms [1, 2]. The accuracy and efficiency of the ABC are compared with the differential evolution (DE) [3], the PSO [4], and GA [5] for numerical optimizations with multi-dimensions, and the ABC is proven to be a better heuristic for global numerical optimizations. Therefore, there emerges a lot of interests on ABC, fox example, see [6-9], without claiming of completeness.

Although above ABC family achieve significant contributions, it is still interesting and important to search all the existing global solutions for optimization. On the one hand, when dealing with the numerical optimization, we are not only interested in finding the global best objective with one solution, but also achieving all the associated solutions. On the other hand, in practice engineering, finding all the existing solutions associated to global optimization is necessary and important, because each one of these existing solutions always corresponds to an operation mode of equipment. Although all these operation modes can be operated on the equipment to obtain the same economic benefits, different
operation modes can consume different consumptions. We desirably adopt the solution which leads to
the best economic benefit with the least operation consumption. By repeating several running times, we
may find several of such solutions with the ABC. However, it is not allowed to run the algorithms
repeatedly in practice.

Furthermore, in the viewpoint of biology, the individuals in the real world can live as a personal life
and also as a member of a society. The communications not only occur between different individuals in
a society, but also happen among different societies. Within a society, the information is spread through
communication between the individuals. Among the different societies, the information associated to
different societies is communicated in a deduced way. This phenomenon provides another motivation
to propose MPABC algorithm. The objective of this paper is to propose a variant of ABC, called multi-
population artificial bee colony algorithm, to deal with numerical optimization with multiple solutions.
Interestingly, we also find that the proposed MPABC algorithm holds high abilities on convergence
speed and accuracy, by comparing with the original ABC algorithm.

The rest of this paper is organized as follows. Section 2 briefly introduces the foraging behavior of
real bees and then ABC algorithm. simulating such behavior. In section 3, the MPABC is proposed.
Section 4 conducts some experiments to test the proposed MPABC with ABC. The last section 5
concludes the paper.

2. Artificial Bee Colony Algorithm
In the ABC algorithm, the position of a food source represents a possible solution of 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, a randomly initial population of n solutions (food source positions),
denoted as xi, can be initialized according to

\[ x_{i,j} = x_{\text{min},j} + \text{rand}[0,1] \times (x_{\text{max},j} - x_{\text{min},j}) \]  

(1)

Where xmin,j and xmax,j are the lower bound and upper bound of the jth dimension, respectively.
The symbol rand[0, 1] denotes a uniform random number in the range [0, 1].

An onlooker bee chooses a food source depending on the probability value associated with that food
source, pi, calculated by

\[ p_i = \frac{\text{fit}_i}{\sum_{l=1}^{n} \text{fit}_l} \]  

(2)

Where fiti is the nectar amount (i.e., the fitness value) of the ith food source xi. Obviously, the higher
the fiti is, the more probability that the fiti food source is selected.

In order to produce a candidate food position from the old one, the ABC uses the following
mechanism:

\[ x_{\text{new},j} = x_{\text{new},j} + \text{rand}[-1,1] \times (x_{i,j} - x_{k,j}) \]  

(3)

Where k ∈ {1, 2, ..., ne} and k ≠ i, and j ∈ {1, 2, ..., d} are randomly chosen indexes. rand[-1, 1] is
a uniformly distributed real number in interval [-1, 1]. If a parameter produced by this operation exceeds
its predetermined band, the parameter can be set to its band.

The food source whose nectar is abandoned by the bees is replaced with a new food source by the
scouts. In the ABC algorithm this is simulated by randomly producing a position and replacing it with
the abandoned one. In addition, if a position cannot be improved further through a predetermined
number of cycles called limit, then that food source is assumed to be abandoned. After each candidate
source position xnew, j is produced and then evaluated by the artificial bee, its performance is compared
with that of \( x_i, j \). If the new food has equal or better nectar than the old source, it is replaced with the old one in the memory. Otherwise, the old one is retained.

3. MPABC: Multi-population Artificial Bee Colony algorithm

3.1. Individual society: Partition of solution space

Suppose variable \( x \) with \( d \)-dimension. Symbol \( x_{i,j} \) denotes the \( i \)th observation of variable \( x \) in the \( j \)th dimension. The boundary of the variable \( x \) is symbolized as \([LB, UB]\), and the corresponding boundary in the \( j \)th dimension can be further symbolized as \([LB_j, UB_j]\) in a deduced way. Thus, the whole solution space \([LB, UB]\) of variable \( x \) can be partitioned in two ways; uniform partition and heterogeneous partition. The uniform partition means that one or more of the boundaries \([LB_j, UB_j]\) of \( x_j \) are divided into sets of equal intervals; otherwise, the partition is called heterogeneous. For conveniently, we adopt the uniform partition scheme.

Note that a subspace induces a society of honeybees. Suppose the \( j \)th dimension of the space is divided into numbers \( K_j \) of partitions. Then, the number \( N \) of societies of honeybees can be calculated as

\[
N = \prod_{j=1}^{d} K_j
\]

Consequently, the boundary of the arbitrary individual society, indicated as \( \tilde{k} \in \{(k_1,\ldots,k_j,\ldots,k_d)\mid k_j = 1,2,\ldots,K_j\} \), can be defined as \([LB^{\tilde{k}}, UB^{\tilde{k}}]\), of which the boundary in the \( j \)th dimension can be calculated as:

\[
[LB^{\tilde{k}}, UB^{\tilde{k}}] = [LB_j + (k_j - 1)\Delta_j, LB_j + k_j\Delta_j], \quad j = 1,2,\ldots,d; k_j = 1,2,\ldots,K_j
\]

Where \( \Delta_j = (UB_j - LB_j) / K_j \) is the width of the intervals in the \( j \)th dimension of \([LB_j, UB_j]\).

3.2. Individual society of honeybees

In each subspace, an individual society of honeybees is produced. Within the individual society, the behaviors of the bees are identical to those of the ABC algorithm. More precisely, MPBAC performs (1)~(3) in each individual society \( \tilde{k} \). In what follows, to distinct different societies, we use \( \tilde{k} \) as superscript for all solutions, fitness and variables.

3.3. Connections among individual societies

In the MPABC algorithm, the connections among individual societies are necessary. In our model, these individual societies are connected by using two ways: sharing local optimizations and competing to living.

- Sharing local optimizations

Suppose the local optimization of the \( l \)th individual society \( \tilde{k}_l \) (\( l = 1, 2, \ldots, N \)) is searched as \( x^{\tilde{k}_l,C} \) at the cycle \( C \). The global optimization is determined as following expression at the cycle \( C \):

\[
x^{best,C} = \max_{\tilde{k}} \{x^{\tilde{k}_l,C}\}
\]
Competing to living

When a number of cycle times have been made, not all individual societies are the survivors. In the real world, the real bees had to compete to living. In course of time the society of real bees would be elimination if such society of bees can not find sufficient food sources.

Suppose the size of individual societies is N, we call an individual society is the winner if the local optimization of such individual society is the best; otherwise, the individual society is loser. During a period, if the probability that the role of an individual society is the winner is smaller than a threshold, called living probability, such individual society of honeybees need to be eliminated from the multiple populations. The living probability of an individual society is defined as:

$$ p_k^l = \frac{\sum_{m=0}^{C-1} \text{sgn}\left(x_{k,m}^{l}\right)}{C_0}, C > C_0 $$

(7)

Where $x_{k,m}^{l}$ denotes the solution associated to the local optimization for the individual society $k$ at the m-th cycle. C is the current cycle time, and $C_0$ is a preset living period. The symbol $\text{sgn}\left(x_{k}^{l}\right)$ is the function defined as:

$$ \text{sgn}\left(x_{k}^{l}\right) = \begin{cases} 1, & x_{k}^{l} = x_{\text{best},C} \\ 0, & \text{otherwise} \end{cases} $$

(8)

By using Eq. (6), MPABC can get the global optimization. According to Eq. (7), the individual societies that impossibly achieve the global optimization are abandoned. These operations can accumulate the convergence speed, and reduce the computational consumption by MPABC.

3.4. Pseudo-code of MPABC algorithm

On the basis of the above interpretation of MPABC, the MPABC algorithm can be summarized as follows.

Algorithm 1: MPABC algorithm

**Input:** Objective function $f(.)$ and the initial solution space

**Output:** The global optimization and associated solutions

1. Initialization of algorithm parameters $n_c$, limit, $K_j$, $C_{\text{max}}$, $C_0$, $P_0$
2. Determine the number of individual societies $N$ by (4), and choose the partition technique;
3. Initialize $N$ number of societies of honeybees for each individual society $k$ ($l = 1, 2, \ldots, N$) by using Eq. (1);
4. Initialize multiple populations of employed bee solutions and evaluate fitness value for each bee in each individual society; set the best fitness value among these individual societies be the initial global optimization, denoted as $\text{global}_\text{opt}$, and its associated solutions be the initial global solutions, denoted as $\text{global}_\text{sol}$;
5. $C = 1$, $l = 1$ % C is the index of cycle and $l$ is the index of the $l$-th individual society
6. Repeat
7. \hspace{1cm} Repeat
8. \hspace{2cm} Select the $l$-th individual society of bees $k_i$;
9. \hspace{2cm} $i = 0$; % the index of the $i$th employed bee
10. \hspace{2cm} Repeat
11. \hspace{3cm} $k$ = a solution in the neighborhood of $i$;
12. \hspace{3cm} $\text{rand}[-1, 1]$ = a random number in the range $[-1, 1]$;
13. Produce new solutions $x_{new,i}^k$ in the neighborhood of $x_i^k$ for the employed bees using Eq. (3);
14. Apply the greedy selection process between $x_{new,i}^k$ and $x_i^k$;
15. Calculate the probability $p_i^k$ for the solutions $x_i^k$ by means of their fitness values using Eq. (2);
16. Assign onlooker bees, i.e., produce new solutions $x_{new,i}^k$ for the employed bees $x_i^k$ according to probabilities and evaluate them;
17. For all onlooker bees Do
   18. Apply the greedy selection process for the onlookers between $x_{new,i}^k$ and $x_i^k$;
   19. If $fit_i^k$ (Best onlooker) < $fit_i^k$ (Employed bees)
      Replace employed bee solution with respective onlooker solution;
   20. End If
   21. End For
22. If $fit_i^k$ (Best Feasible onlooker, $x_i^{k,c}$) < global_opt
23. Find the Best Feasible onlooker $x_i^{k,c}$, and replace with the Best solution;
24. Memorize $x_i^{k,c}$ and $fit_i^k (x_i^{k,c})$.
25. End If
26. $i = i + 1;
27. Until i = n_e, n_e = n$, is the number of employed bees
28. Determine the abandoned solution, if exists, and replace it with a new randomly produce solution $x_i^k$ for the scout bees by Eq. (1);
29. The worst employed bees as many as the number of scout bees in the individual society $k$, are respectively compared with the scout solutions. If the scout solution is better than employed solution, employed solution is the next cycle without any change;
30. $l = l + 1;
31. Until l = N
32. $q = \arg \min_{i,j \in C (k)} \left( fit_i^k (x_i^{k,c}) \right), global_{sol} = x_i^{k,c}, global_{opt} = fit_i^k (x_i^{k,c})$
33. If $C > C_0$
34. For $m = C - C_0 + 1$ to $C$ Do
35. Calculate the living probability $P_i^k$ of each individual society according to Eq. (7);
36. End For
37. For $l = 1$ to $N$ Do
38. If $P_i^k \leq P_0$
39. The $l$-th individual society $k_l$ is eliminated from the multiple populations;
40. $N = N - 1;
41. End If
42. End For
43. $C = C + 1;
44. Until C = C_{max}, C_{max}$ means the maximum cycle
45. Return the best solutions obtained in Step 31 and their associated local optimizations. It is evident that

4. Experimental analysis
In this section, the performance of the proposed MPABC is validated using some numerical functions as shown in Table 1.
Table 1. Numerical benchmark functions

| Function names | Function expressions | Ranges | Minimum value |
|----------------|----------------------|--------|---------------|
| Sphere \((d = 50)\) | \(f_1(\mathbf{x}) = \sum_{i=1}^{n} x_i^2\) | \(-100 \leq x_i \leq 100\) | \(f_1(\mathbf{0}) = 0\) |
| Griewank \((d = 50)\) | \(f_2(\mathbf{x}) = \frac{1}{4000} \sum_{i=1}^{n} (x_i - 100)^2 - \prod_{i=1}^{n} \cos \frac{x_i - 100}{\sqrt{i}} + 1\) | \(-600 \leq x_i \leq 600\) | \(f_2(100) = 0\) |
| Rastrigin \((d = 50)\) | \(f_3(\mathbf{x}) = \sum_{i=1}^{n} (x_i^2 - 10\cos (2\pi x_i) + 10)\) | \(-5.12 \leq x_i \leq 5.12\) | \(f_3(\mathbf{0}) = 0\) |
| Rosenbrock \((d = 50)\) | \(f_4(\mathbf{x}) = \sum_{i=1}^{n} 100(x_i^2 - x_{i+1}^2)^2 + (x_{n+1} - 1)^2\) | \(-10 \leq x_i \leq 10\) | \(f_4(\mathbf{1}) = 0\) |

In Algorithm 1, parameters \(n_e\), limit, \(K_j\), \(C_{max}\), \(C_0\), \(P_0\) need to be predetermined. Colony size of bees in this study takes value form the set \{20, 40, 100\}. Thus, the number of employed bees can take values as 10, 20, and 50, respectively. The limit takes value from set \{0.1 \times d \times n_e\}, and the maximal cycle is set as \(C_{max} = 5000\). In addition, we set \(C_0 = 100\), \(P_0 = 0\), \(K_j \in \{4, 5, 6\}\), where “\(P_0 = 0\)” means that an individual society will be eliminated if such society can not search the global optimization one time during the period \(C_0\).

Table 2 and Figs. 1–4 shows the performances of MPABC algorithm by comparing with ABC algorithm. It reveals that MPABC has fast convergence and most appropriate solutions.

Table 2. Comparison between the MPABC and the ABC *1

| Functions | Mean and std. | MPABC | ABC | |
|-----------|---------------|-------|-----||
| \(f_1(x)\) Mean | 0 (4, 6)(20, 40, 100)(d\times n_e) | 1.4897e-015 | 1.6134e-016 |
| Std. | 0 | 2.8515e-015 |
| \(f_2(x)\) Mean | 0 (4, 6)(20, 40, 100)(d\times n_e) | 3.5638e-015 | 5.3663e-013 |
| Std. | 0 | 1.7139e-012 |
| \(f_3(x)\) Mean | 2.7610e-013 (5)(100)(d\times n_e) | 1.0630e-012 | 2.6345e-013 |
| Std. | 5.3663e-013 | 1.7139e-012 |
| \(f_4(x)\) Mean | 0.0035 (6)(100)(d\times n_e) | 0.0145 | 0.0039 |
| Std. | 0.0039 | 0.0177 |

*1 The numbers in the parentheses refer to as the number of population of the colony bees, the colony size of employed bees, and the number of trivial limit, respectively. For instance, the interpretation of the form “(4, 6)(20, 40, 100)” means that the population number is 4 or 6, the colony size of employed bees is20, 40 or 100.

Fig. 1 Comparisons between MPABC and ABC algorithms on Sphere function.
Fig. 2 Comparisons between MPABC and ABC algorithms on Griewank function.

Fig. 3 Comparisons between MPABC and ABC algorithms on Rastrigin function.

Fig. 4 Comparisons between MPABC and ABC algorithms on Rosenbrock function.
5. Conclusion
This paper proposed a multi-population artificial bee colony (MPABC) algorithm for numerical functions. With a reasonable partition of solution space, all the global optimization and all the associated global solutions can be found by using MPABC. This property cannot be committed by the ABC, as the ABC is interested in finding the global optimization with one solution in one running time. Based on four typical numerical functions, it reveals that the MPABC has higher abilities on convergence speed and accuracy than the ABC.

Acknowledgments
This work is totally supported by The Key Project of Yunnan Power Grid Co. Ltd. (YNYJ2016043).

References
[1] D. Karaboga, An Idea Based on Honey Bee Swarm for Numerical Optimization, Technical Report-TR06, Erciyes University, Engineering Faculty, Computer Engineering Department, 2005.
[2] D. Karaboga, B. Basturk, On the performance of Artificial Bee Colony (ABC) algorithm, Applied Soft Computing 8 (1) (2008) 687 – 697.
[3] U. Maulik, I. Saha, Automatic fuzzy clustering using modified differential evolution for image classification, IEEE Transactions on Geosience and Remote Sensing 48 (9) (2010) 3503 – 3510.
[4] T.A. Runkler, C. Katz, Fuzzy clustering by particle swarm optimization, in: IEEE International Conference on Fuzzy Systems, Sheraton Vancouver Wall Centre Hotel, Vancouver, BC, Canada, July 16 – 21, 2006, pp. 601 – 608.
[5] M. Mitchell, An Introduction to Genetic Algorithms, MIT Press, Cambridge, MA, 1996.
[6] A. Kumar, D. Kumar, S. K. Jarial, A Review on Artificial Bee Colony Algorithms and Their Applications to Data Clustering. Cybernetics and Information Technologies, 17 (3): 3–28, 2017.
[7] D. Karaboga, E. Kaya, An adaptive and hybrid artificial bee colony algorithm (aABC) for ANFIS training, Applied Soft Computing 49 (2016) 423–436.
[8] D. Karaboga, E. Kaya, Training ANFIS by using the artificial bee colony algorithm, Turkish Journal of Electrical Engineering & Computer Sciences 25 (2017) 1669 – 1679.
[9] D. Karaboga, C. Ozturk, A novel clustering approach: Artificial Bee Colony (ABC) algorithm, Applied soft computing, 11 (1) (2011) 652-657.