The 5th Annual IEEE International Conference on
Cyber Technology in Automation, Control and Intelligent Systems
June 8-12, 2015, Shenyang, China

# A Hybrid Artificial Bee Colony Algorithm with Bacterial Foraging Optimization

L. Li[1], F.F. Zhang[1]
[1]College of Management, Shenzhen University, Shenzhen , China
[2]School of Economic and Management, Bejing University of Technology, Bejing, China

C. Liu[2]*, B. Niu[3]*
[3]Department of Industrial and System Engineering, The Hong Kong Polytechnic University, Hong Kong
E-mail: liuchao@bjut.edu.cn, ben.niu@polyu.edu.hk

*Abstract*—**The Artificial Bee Colony (ABC) algorithm is a new swarm optimization algorithm with good numerical functions optimization results. In order to enhance the performance ability of ABC algorithm, a hybrid ABC (HAB) algorithm is presented where swarming behavior of bacterial foraging optimization algorithm is introduced into the ABC algorithm to do local search. The performance of the proposed method is examined on well-known six numerical benchmark functions and the obtained results are compared with basic ABC algorithm and BFO algorithm. The experimental results show that the proposed approach is very effective method for solving numeric benchmark functions and successful in terms of solution quality and convergence to the global optimum, especially on the multimodal functions.**

***Keywords: Swarm Intelligence; Artificial Bee Colony; Bacterial Foraging Optimization***

## I. INTRODUCTION

Biologically inspired computing is becoming more and more popular because of its immense parallelism and simplicity in computation. Swarm intelligence is a branch of artificial intelligence and the algorithms of swarm intelligence have been developed by inspiring natural behavior of real bees, ants, birds, fishes etc. For example, Ant Colony Optimization algorithm simulates the natural foraging behavior of ants. Particle Swarm Optimization (PSO) [1] algorithm imitates the social behaviors of birds flocking and Bacterial Foraging Optimization (BFO) [2] algorithm is inspired by the social foraging behavior of Escherichia coli. What's more, Artificial Bee Colony (ABC) [3] algorithm simulates the behaviors of a group bees on looking for food sources.

Inspired by the group foraging strategy of a swarm of E.coli bacteria, the BFO algorithm was proposed by Passino. The BFO algorithm consists of four principal mechanisms, namely chemotaxis, swarming, reproduction, and elimination dispersal [2]. In particular, the swarming mechanism makes the bacteria congregate together and thus move as excellent groups with high bacterial density. It is also desired that the bacterium that has found the better path of food can provide guidance for the other bacteria.

Artificial bee colony algorithm is an optimization method which is proposed by mimicking bee behavior. Karaboga first recommended in detail in 2005[3]. ABC algorithm has been widely used in solving many numerical functions optimization [4-5] and practical engineering optimization problems[6-7] because of its convergence rate and fewer parameters than other algorithms. However, with the increase of the number of iterations, the speed of convergence of ABC decreases.

In order to enhance the exploration ability of the ABC, hybridization of evolutionary algorithms with local search has been investigated in many studies. The concept of universal gravitation is introduced to the movement of onlooker bees in the ABC[8] by Tsai et al.. Aiming to avoid the premature and improve the population diversity, Bao conduct research on several selection strategies, such as tournament selection strategy and rank selection strategy[9].

Inspired by the intelligence of BFO algorithm, swarming behavior of BFO algorithm is introduced into employed bees and onlooker bees in this paper, in hope to improve the global optimum ability of ABC algorithm.

The rest of paper is organized as follow. Section II introduces ABC algorithm and the BFO algorithm is presented in section III. Aiming to improve the performance of ABC algorithm, swarming behavior of BFO is introduced into ABC algorithms in section IV. The simulation results are presented and discussed in section V. Finally, the conclusion is in section VI.

## II. ARTIFICIAL BEE COLONY ALGORITHM

ABC algorithm is an optimization approach which refers to the behavior of honey bees warms. In ABC algorithm, artificial honeybees consist of employed bees, onlookers and scouts. A bee, which goes to the food source found by itself previously, is called an employed bee. Onlookers are always waiting for making decisions to select a better place with food source. And a scout is used to doing random search.

In the ABC algorithm, onlookers and employed bees perform the exploration process in the search space, while scouts control the exploration process. The process of bees look for food source is the process of find the optimum solution.

## III. BACTERIAL FORAGING OPTIMIZATION ALGORITHM

Using the theory of E. coli foraging behavior which based on biology and physics, Passino and Liu [10] applicated a series of bacterial swarming and social foraging behaviors, also they researched the matching between the control system in E. coli and the implementation of forage. All bacteria try to move upward the food concentration gradient individually. Each bacteria is carried out in accordance with the four steps (chemotaxis, swarming, reproduction, and elimination and dispersal) to search a better food source. The details of BFO are given in [11-12]. Below we briefly describe each of these processes.

(1) Chemotaxis: An E.coli bacterium can move in two different ways biologically. It often swims in the same direction or it may tumble, and alternate between these two kinds of operation in his life.

(2) Swarming: The cells when stimulated by high level of succinate release an attractant aspartate, which helps them to gather into several subsets and thus move as a team of high bacterial density.

(3) Reproduction: The weakest bacteria eventually die while each of the healthier bacteria produces their own offspring in the same place. This keeps the swarm size constant.

(4) Elimination and Dispersal: It is possible that in the local environment, the lives of a population of bacteria changes either gradually or suddenly due to some other influence. To simulate this phenomenon in BFO some bacteria are liquidated at random with a very small probability while the new replacements are randomly initialized over the search space.

## IV. HAB OPTIMIZATION ALGORITHM

ABC is a swarm optimization algorithm which has been shown to be more effective than the other population based algorithms such as PSO and ant colony optimization (ACO). Since it was invented, it has received significant interest from researchers studying indifferent fields because of having fewer control parameters, high global search ability and ease of implementation. Although ABC is good at exploration, the main drawback is its poor exploitation which results in an issue on solution quality in some cases.

In the canonical ABC algorithm, the solution updating equation of a basic ABC has several issues such as inefficiency during a local search on the solution space. On the contrary, although the BFO algorithm has slow convergence, but it has the better ability to achieve to the global optimum. It is worth mentioning that the swarming mechanism of the BFO algorithm makes the bacteria congregate into groups and then move as a pattern of concentric with high bacterial density. So, the bacterium that has reached the best path of food source should let other bacteria toward the same direction in order that they can find the destination more rapidly and accurately.

With the purpose of improving the efficiency of the ABC, we propose the swarming mechanism of BFO algorithm within the Employed Bees' Phase and Onlooker Bees' Phase of the ABC algorithm. The main steps of the HAB algorithm are listed in Table 1 and the pseudo-code of swarming mechanism is depicted in Table 2.

Table 1. Main steps of the HAB algorithm

```
1: Initialize variables and randomize positions;
2: WHILE ((Iter < MaxCycle))
3: /*Employed Bees' Phase*/
   FOR(i=1:(FoodNumber))
   Produce a new food source;
   Evaluate the fitness of the new food source;
   Swarming mechanism; Greedy selection;
   END FOR
4: Calculate the probability P;
5: /*Onlooker Bees' Phase*/
   FOR(i=1:(FoodNumber))
   Parameter P is determined randomly;
   Onlooker bees find food sources depending on P;
   Produce a new food source;
   Evaluate the fitness of the new food source;
   Swarming mechanism; Greedy selection;
   END FOR
6 : /*Scout Bees' Phase*/
   IF(any employed bee becomes scout bee)
     Parameter P is determined randomly;
     The scout bees find food sources depending on P;
   END IF
7: Memorize the best solution;
   Iter=Iter+1;
   END WHILE
```

Table 2. Swarming mechanism of the BFO algorithm

```
1: Initialize variables
2: Let m = 0;
3: While m < Ns
4: IF(the mutant solution is better than the current Solution)
      Update the solution by the mutant solution;
   END IF
      Let m = m + 1;
5:  Else, let m = Ns.
```

## V. Experiments

### A. Benchmark Functions

In this paper we have used six well-known benchmark functions to test the performance of proposed HAB algorithm, which also compared with standard ABC and BFO algorithm. These functions contain three unimodal functions (SumPowers, Sphere, Schwefel221), and three multimodal functions (Apline, Griewank, Rastrigin). The search ranges of those benchmark functions of BFO and ABC are given in Table 3. In our research, each experiment is conducted twenty times in all the experiments.

Table 3. Search ranges of the benchmark functions

| Name | Function | Range |
|---|---|---|
| SumPowers | $f(x) = \sum_{i=1}^{n} \left\| x_i \right\|^{i+1}$ | [-10,10] |
| Sphere | $f(x) = \sum_{i=1}^{n} x_i^{\ 2}$ | [-100,100] |
| Schwefel221 | $f(x) = \max \left\| x_i \right\|$ | [-10000,10000] |
| Apline | $f(x) = \sum_{i=1}^{n} x_i^{\ 2}$ | [-10,10] |
| Griewank | $f(x) = \frac{1}{4000}\sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n}\cos(\frac{x_i}{\sqrt{i}}) + 1$ | [-600,600] |
| Rastrigin | $f(x) = \sum_{i=1}^{n} (x_i^2 - 10\cos(2\pi x_i) + 10)^2$ | [-5.12,5.12] |

### B. Parameter Settings for the Involved Algorithms

In the experiment, all functions are tested with 20 dimensions and run for 20 times. We choose population sizes are 20. In ABC and HAB, the number of scout bees is only one, and we choose half of the population size equal to employed bees and onlooker bees. What's more, the abandon limit is 100. The main parameter settings for ABC and BFO are presented in Table 4.

Table 4. The parameters setting of ABC and BFO algorithm

| Algorithm | Dim | SS | NP | FoodNumber | Limit | maxCycle |
|---|---|---|---|---|---|---|
| ABC | 20 | 20 | 20 | 10 | 100 | 3000 |
| | Dim | SS | Ped | Fed | C | Ns |
| BFO | 100 | 10 | 0.25 | 5 | 0.05 | 8 |

### C. *Simulation Results for Benchmark Functions*

The mean and standard deviations of function values obtained by ABC, BFO and HAB algorithms for 20 runs are given in Table 4 and Table 5. Best values obtained by the three algorithms for each function were marked as bold.

As shown in Table 5 and Table 6, the HAB algorithm is better than the other two algorithms on SumPowers, Schwefel221, Apline, Griewank and Rastrigin functions while the ABC algorithm shows better performance on Sphere function slightly.

Table 5. Preformance of three algorithms on unimodal benchmark functions

| Algorithm | | SumPowers | Sphere | Schwefel221 |
|---|---|---|---|---|
| ABC | mean | 4.36489e-016 | **4.09210e-016** | 3.08493e-001 |
| | std. | 7.78778e-017 | 1.07959e-016 | 7.20241e-002 |
| SiBFO | mean | 1.31920e-002 | 1.03572e+000 | 6.18504e-001 |
| | std. | 1.13078e-002 | 2.95177e-001 | 8.69007e-002 |
| HAB | mean | **4.39471e-016** | 4.12051e-016 | **2.22546e-001** |
| | std. | **7.29621e-017** | **1.04204e-016** | **6.66939e-002** |

Table 6. Preformance of three algorithms on multimodal benchmark functions

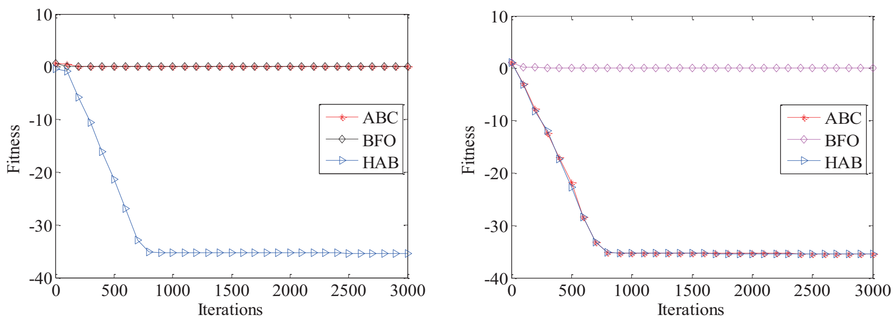
| Algorithm | | Apline | Griewank | Rastrigin |
|---|---|---|---|---|
| ABC | mean | 5.11129e-009 | 3.07840e-003 | 2.34479e-014 |
| | std. | 1.57747e-008 | **2.41519e-003** | 7.73601e-014 |
| SiBFO | mean | 9.72015e-001 | 9.95577e-002 | 7.04353e+001 |
| | std. | 2.78428e-001 | 4.11029e-002 | 6.35446e+000 |
| HAB | mean | **5.73396e-010** | **2.46701e-003** | **3.10862e-015** |
| | std. | **9.56626e-010** | 2.53095e-003 | **7.42265e-015** |

The mean best function value profiles are shown in Figure 1. For most functions, the ABC algorithm converges fast at the beginning and traps in the local optimum soon, but the BFO algorithm achieve better solution quality finally.

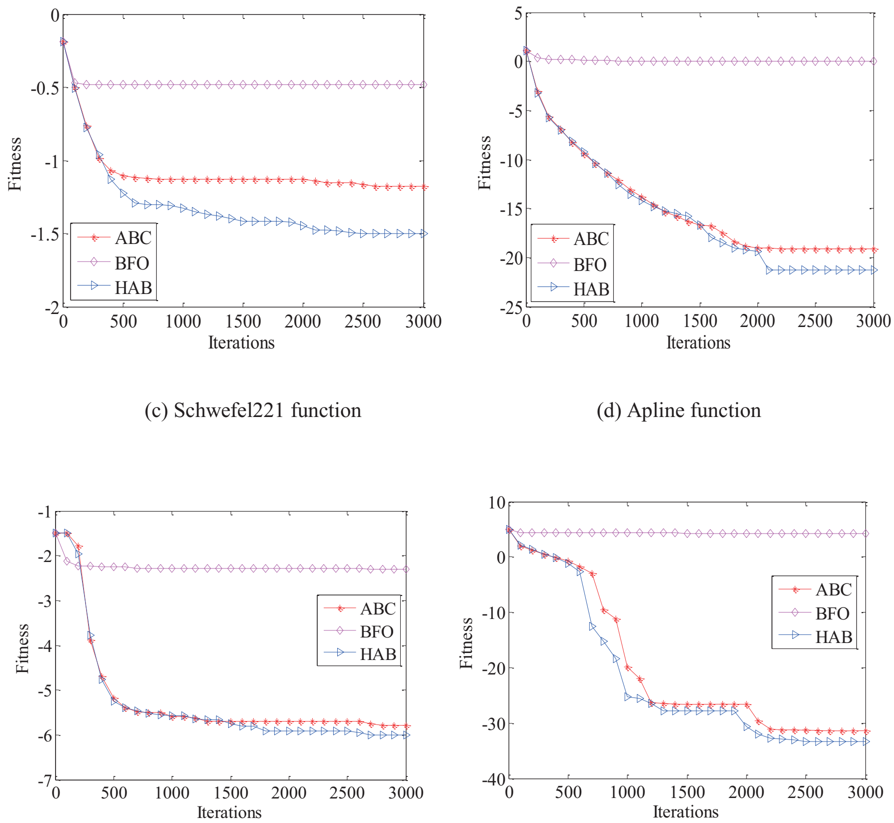
SumPowers, Sphere and Schwefel221 functions are three unimodal functions. On SumPowers function, HAB algorithm shows the best performance compared with ABC algorithm and BFO algorithm obviously. As it can be seen in Fig.1. (b), the convergence profiles of the HAB algorithm is much similar with the ABC algorithm, though results on Sphere is a little worse than the ABC algorithm. as HAB>ABC>BFO. Specially, on the Rastrigin function, the convergence speed is ordered as HAB>ABC>BFO. Overall, the HAB algorithm offers improved performance

As for Schwefel221 function, ABC algorithm converges fast at the beginning and then traps in the local optimum immediately, as shown in Fig.1. (c). However, the HAB algorithm can escape from the trap, shows the best performance. The search performance order of on this function is HAB>ABC>BFO.

Apline, Griewank and Rastrigin functions are three multimodal functions. For these three functions, the results obtained by HAB are all the best. In special, the HAB algorithm not only stands out on the convergence speed but also on the solution quality on the Rastrigin function. The search performance on these three functions is ordered over the standard ABC and BFO on most functions, especially on the unimodal functions.



(a) SumPowers function (b) Sphere function




(c) Schwefel221 function (d) Apline function

(e) Griewank function (f) Rastrigin function

Fig. 1. The mean best function value profile of benchmark functions. (a) SumPowers function. (b) Sphere function. (c) Schwefel221 function. (d) Apline function. (e) Griewank function. (f) Rastrigin function.

## VI. Conclusion

This paper presents a hybrid algorithm of ABC algorithm and BFO algorithm called Hybrid Artificial Bee Colony algorithm with Bacterial Foraging Optimization (HAB). The swarming operator of BFO is introduced in it to improve the original ABC algorithm.

To verify the optimizing performance of HAB, we compared it with canonical ABC and PSO algorithms. The performance and accuracy of the proposed method were examined on benchmark functions. These functions contain three unimodal functions, three multimodal functions. Simulation results have shown that the solutions produced by HAB are better than those produced by the standard ABC algorithm.

## Acknowledgment


This work is partially supported by The National Natural Science Foundation of China (Grants nos. 71001072, 71271140, 61263005, 61273230), China Postdoctoral Science Foundation (Grant nos. 20100480705), Special Financial Grant from the China Postdoctoral Science Foundation (Grant nos. 2012T50584), the Natural Science Foundation of Guangdong Province (Grant nos. S2012010008668, 9451806001002294), and Program for New Century Excellent Talents in University (approval number: NCET-12-1027) .


## References


[1] J. Kennedy and R. Eberhart, "Particle swarm optimization," in Neural Networks, 1995. Proceedings., IEEE International Conference on, 1995, pp. 1942-1948 vol.4.

[2] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," Control Systems, IEEE, vol. 22, pp. 52-67, 2002.

[3] D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Technical Report-TR06, Erciyes

University, Engineering Faculty, Computer Engineering Department, 2005.

[4] D. Karaboga and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm," Journal of Global Optimization, vol. 39, pp. 459-471, 2007/11/01 2007.

[5] D. K. B. Basturk, "An artificial bee colony (abc) algorithm for numeric function optimization," In: IEEE Swarm Intelligence Symposium 2006, Indianapolis, Indiana, USA, 2006.

[6] B. A. D. Karaboga, C. Ozturk, "Artificial bee colony (ABC) optimization algorithm for training feed-forward neural networks," Modeling Decisions for Artificial Intelligence, vol. 4617, pp. 318-329, 2007.

[7] L. O. A. Baykasoglu, P. Tapkan, "Artificial bee colony algorithm and its application to generalized assignment problem, Swarm Intelligence: Focus on Ant and Particle Swarm Optimization," I-Tech Education and Publishing, pp. 113-144, 2007.

[8] J. S. P. P.W. Tsai, B.Y. Liao, and S.C. Chu, "Enhanced artificial bee colony optimization. International Journal of Innovative Computing," Information and Control, vol. 5, pp. 1-12, 2009.

[9] L. B. a. J. C. Zeng, "Comparison and Analysis of the Selection Mechanism in the Artificial Bee Colony Algorithm," Ninth International Conference on Hybrid Intelligent Systems, August 12-14, 2009, shenyang, china 2009, 2009.

[10] Y. Liu and K. M. Passino, "Biomimicry of Social Foraging Bacteria for Distributed Optimization: Models, Principles, and Emergent Behaviors," Journal of Optimization Theory and Applications, vol. 115, pp. 603-628, 2002/12/01 2002.

[11] K. a. S. B. Guney, "Interference suppression of linear antenna arrays by amplitude-only control using a bacterial foraging algorithm," Progress In Electromagnetics Research, pp. PIER 79, 475–497, 2008.

[12] B. B. Mangaraj, I. S. Misra, and A. K. Barisal, "Optimizing included angle of symmetrical V-dipole for higher directivity using bacteria foraging algorithm," Progress In Electromagnetic Research B, vol. 3, pp. 295–314,, 2008.