AN UNDERSTANDING OF ABC ALGORITHM AND ITS APPLICATIONS

[1]Er. Ankit Choubey, [2]Dr. G. L. Prajapati
[1]PG scholar, IET- Devi Ahilya University, Indore, INDIA
[2]Assoc. Prof., Dept. of Computer Engineering, IET- Devi Ahilya University, Indore, INDIA
[1]ankit_choubey@live.com, [2]glprajapati1@gmail.com,

Abstract— Artificial Bee Colony (ABC) is a new swarm-based optimization algorithm, which inspired by the foraging behaviour of honey bees. In ABC, the neighbourhood search strategy is employed in order to find better solutions around the previous ones. Researchers have tested ABC in many practical optimization problems. In this paper we demonstrate the working of ABC for better understanding for the beginners. We also provide a brief survey of the literature on its applications.
Keywords: Artificial Bee Colony Algorithms; Optimizations, swarm Intelligence algorithm

1. Introduction

The last few decades have witnessed the introduction of several optimization algorithms developed based on nature-inspired ideas. Some examples of such algorithms include ant colony optimization, evolutionary algorithm, particle swarm optimization,
Harmony search.

Artificial Bee Colony algorithm (ABC) was initially published by Karaboga in 2005 as a technical report for numerical optimization problems. ABC is a new swarm intelligence algorithm proposed by Karaboga in 2005, which is inspired by the behavior of honey bees [1]. Since the development of ABC, it has been applied to solve different kinds of problems. Artificial bee colony (ABC) algorithm is a recently proposed optimization technique which simulates the intelligent foraging behavior of honey bees.

The major advantages which ABC holds over other optimization algorithms include its:
• Simplicity, flexibility and robustness [2]

• Use of fewer control parameters compared to many other search techniques [3]

• Ease of hybridization with other optimization algorithms.

• Ability to handle the objective cost with stochastic nature [4].
• Ease of implementation with basic mathematical and logical operations.

Particle swarm optimization

Particle swarm optimization (PSO) is a computational method that optimizes a problem by iteratively trying to improve a candidate solution with regard to a given measure of quality. ABC is based on PSO [5]

2. The growth of ABC algorithm

After the invention of ABC by Karaboga (2005), the first conference paper introducing ABC was published in 2006 (Basturk and Karaboga 2006). The first journal article describing ABC and evaluating its performance was presented by Karaboga and Basturk (2007b), in which the performance of ABC was compared to GA, PSO and particle swarm inspired

Evolutionary algorithm. In 2008, the second article presenting a performance evaluation of ABC was published by Karaboga and Basturk (2008). In 2009, a public domain web-site (http://mf.erciyes.edu.tr/abc) dedicated to ABC was constructed. There are several source codes, written in different programming languages, of ABC and many publications about the Modifications to ABC and their applications are presented in the website. The main algorithm of ABC is relatively simple and its implementation is, therefore, straightforward for solving optimization problems and ABC has been found to be very effective in the studies above, being able to produce very good results at a low computational cost. Therefore, after these initial publications many studies have been carried out on ABC.

3. How ABC algorithm works?

The ABC was first proposed to solve numerical optimization problems by Karaboga [6].
ABC consists of employed and unemployed foragers, and food sources. The ABC consists of three groups of artificial bees: employed foragers, onlookers and scouts. The employed bees comprise the first half of the colony whereas the second half consists of the onlookers.

In the basic ABC [6], there are 3 kinds of bees: employed, onlooker, and scout bees.

3.1 Phases of ABC
Outline of ABC shown in the Figure 1. It generally consist of four phases.
1) Initialization of ABC.
Determine the number of artificial bees. 50% are employed bees and 50% are onlooker's bees.
Generate the random initial candidate solutions for employed bees using equation. [6] Determine the limit value.

2) Employed bee phase
For all employed bees
Generate new candidate solution using equation. [6]
Calculate the fitness value of the new solution using
Equation. [6]
If fitness of new candidate solution is better than the existing solution replace the older solution.
Calculate the probability for each individual.

3) Onlooker bee phase.
For all onlooker bees
Select an employed bees using roulette wheel.
Produce new candidate solution.
Compute fitness of individual.
If fitness of new candidate solution is better than the existing solution replace the older solution.

4) Scout bee phase
If any food source exhausted then replace it by randomly generated solution by scout memorize the best solution.
Until (stopping criteria is not met).

4. An example

We consider the following optimization problem for the demonstration.

$$Minimize\ f(x) = x_1^2 + x_2^2 \qquad -5 \leq x_1, x_2 \leq 5$$

Control Parameters of ABC Algorithm are set as;
- Colony size, CS = 6
- Limit for scout, L = (CS*D)/2 = 6

And dimension of the problem, D = 2
First, we initialize the positions of 3 food sources (CS/2) of employed bees, randomly using uniform distribution in the range (-5, 5).
Generate a 2-by-3 matrix of uniformly distributed random numbers between 0 and 1.
r = rand (5)

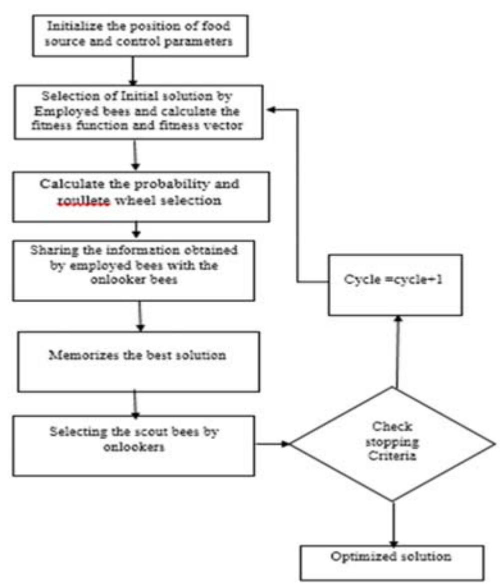

Figure 1. Outline of ABC

x =1.4112(x_{00}) -2.5644(x_{01})
0.4756 (x_{10}) 1.4338(x_{01})
-0.1824(x_{20}) -1.0323(x_{21})

F(x) values are;

8.5678
2.2820
1.0990

$$fit_i = \begin{cases} \frac{1}{1+f_i} & if(f_i \geq 0) \\ 1 + abs(f_i) & if(f_i < 0) \end{cases}$$

Initial fitness vector is:

0.1045
0.3047
0.4764

Maximum fitness value is 0.4764, the quality of the best food source.

Cycle=1

//Employed bee's phase

1st employed bee

$v_{i,j} = x_{i,j} + \emptyset(x_{i,j} - x_{k,j})$ With this formula, produce a new solution.

k=1 //*k* is a random selected index.

j=0 //*j* is a random selected index

Φ = 0.8050 // Φ is randomly produced number in the range [-1, 1].

(x_{00} *is changed while* x_{01} *remains same*)

*υ*0=

2.1644 -2.5644

Calculate *f(υ0)* and the fitness of *υ0*.

f(υ0) = 11.2610 and the fitness value is 0.08155

Apply greedy selection between *x*0 and *υ*0

0.0816 < 0.1045, the solution 0 couldn't be improved, increase its trial Counter

2nd employed bee

o $v_{i,j} = x_{i,j} + \emptyset(x_{i,j} - x_{k,j})$ with this formula produce a new solution.

k=2 //k is a random selected solution in the neighborhood of *i*.

j=1 //j is a random selected dimension of the problem.

Φ = 0.0762 // Φ is randomly produced number in the range [-1, 1].

(x_{11} *is changed while* x_{10} *remains same*)

*υ*1=

0.4756 1.6217

Calculate *f(υ1)* and the fitness of *υ1*.

f(υ1) = 2.8560 and the fitness value is 0.2593.

Apply greedy selection between *x*1 and *υ*1

0.2593 < 0.3047, the solution 1 couldn't be improved, increase its trial counter.

3rd employed bee

$v_{i,j} = x_{i,j} + \emptyset(x_{i,j} - x_{k,j})$ with this formula produce a new solution.

k=0 //*k* is a random selected solution in the neighborhood of *i.*

j=0 //*j* is a random selected dimension of the problem.

Φ = -0.0671 // Φ is randomly produced number in the range [-1, 1].

V_2= -0.0754 -1.0323

o Calculate *f(υ2)* and the fitness of *υ2*.

f(υ2) = 1.0714 and the fitness value is 0.4828.

Apply greedy selection between *x*2 and *υ*2.

0.4828 > 0.4764, the solution 2 was improved, set its trial counter as 0 and replace the solution *x*2 with *υ*2.

Now the solution is

X =

1.4112	-2.5644
0.4756	1.4338
-0.0754	**-1.0323**

f(x) values are;

8.5678
2.2820
1.0714

Fitness vector is:

0.1045
0.3047
0.4828

//Calculate the probability values *p* for the solutions *x* by means of their fitness //values by using the formula;

$$p_i = \frac{fit_i}{\sum_{j=1}^{n} fit_i}$$

p =

0.1172
0.3416
0.5412

On the basis of probability and Roulette wheel selection onlooker bee is selected.

Onlooker bee's phase

Produce new solutions *υi* for the onlookers from the solutions *xi* selected depending on *pi* and evaluate them.

1st onlooker bee

i=2, j=1, k=0

Φ =-0.8350

υ2=

-0.0754 -2.3116

Calculate *f(υ2)* and the fitness of *υ2*.

f(υ2) = 5.0772 and the fitness value is 0.1645.
Apply greedy selection between $x2$ and $\upsilon 2$
0.1645 < 0.4828, the solution 2 couldn't be improved, increase its trial counter.

2nd onlooker bee

i=1,j=0,k=0

Φ =0.42873

υ1=

0.0754 1.4338

Calculate *f(υ1)* and the fitness of *υ1*.
f(υ1) = 2.0855 and the fitness value is 0.3241.
Apply greedy selection between $x1$ and $\upsilon 1$
0.3241 > 0.3047, the solution 1 was improved, set its trial counter as 0 and replace the solution $x1$ with $\upsilon 1$.

x =

1.4112	-2.5644
0.0754	1.4338
-0.0754	-1.0323

f(x) values are;

8.5678
2.0613
1.0714

Fitness vector is:

0.1045
0.3241
0.4828

3rd onlooker bee

i=2, j=0, k=0

υ2=

0.02435 -1.0323

Calculate *f(υ2)* and the fitness of *υ2*.
f(υ2) = 1.0662 and the fitness value is 0.4839.
Apply greedy selection between $x2$ and $\upsilon 2$
0.4838 > 0.4828, the solution 2 was improved, set its trial counter as 0 and replace the solution $x2$ with $\upsilon 2$.

x =

1.4112	-2.5644
0.1722	1.4338
0.02435	-1.0323

f(x) values are;

8.5678
2.0855
1.0669

Fitness vector is:

0.1045
0.3241
0.4838

//Memorize best

Best =

0.02435 -1.0323

//Scout bee phase

Trial Counter =

1
0
0

//There is no abandoned solution since L = 6
//If there is an abandoned solution (the solution of which the trial counter value is
//higher than L = 6); generate a new solution randomly to replace with the
//abandoned one.
Cycle = Cycle+1
The procedure is continued until the termination criterion is attained.

4. Application of ABC

1. Travelling Salesmen Problem

Travelling Salesman Problem (TSP) belongs to the class of NP-Complete problems .ABC finds better solution than GA and other ABC is widely used for optimization. [7]

2. Graph coloring

Graph coloring has been done using artificial bee colony (ABC) optimization algorithm. In Graph coloring no two adjacent edges having same colour. This is solve by using ABC with better result than other algorithm. [8]

3. Bioinformatics application

In the field of Bioinformatics ABC results in optimizing the DNA sequencing problem with better result as compared to other algorithms. [9, 10]

4. Image processing Applications

ABC has its application to image processing. Several difficult problems exist in pattern recognition and image processing research areas. ABC optimization works better and optimized the pattern recognition problem.so it is widely used for image processing. [11]

5. Benchmarking Optimization

Numerous function exits which can be optimized using ABC algorithm. [12, 13, 14]

5. Conclusion.

ABC can be applied to many different problem. Literatures show that ABC results in better optimization of different problem than other algorithm. As a future research it is still needed to identify the problems where ABC algorithm can do better as compared to other optimization algorithms.

6. References

[1] D. Karaboga. An idea based on honey bee swarm for numerical optimization. *Techn. Rep. TR06, Erciyes Univ. Press, Erciyes,* 2005.

[2]A. Singh. An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem. *Applied Soft Computing,* 9(2):625–631, 2009.

[3] D. Karaboga and B. Akay. A comparative study of artificial bee colony algorithm. *Applied Mathematics and Computation,* 214(1):108–132, 2009.

[4] D. Karaboga and B. Basturk. A powerful and efficient algorithm for numerical function optimization: artificial bee colony (abc) algorithm. *Journal of Global Optimization,* 39(3):459–471, 2007.

[5]Kennedy, J.; Eberhart, R. (1995). "Particle Swarm Optimization". Proceedings of IEEE International Conference on Neural Networks **IV**. pp. 1942–1948. Doi:10.1109/ICNN.1995.48896

[6]Dervis Karaboga · Bahriye Basturk, "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm" J Glob Optim (2007), pp 459-471.

[7] A Hybrid Mutation-based Artificial Bee Colony for Traveling Salesman Problem *Lecture Notes on Information Theory Vol. 1, No. 3, September 2013.*

[8] A Novel ABC Optimization Algorithm for Graph Coloring Problem ,Computational Intelligence and Communication Networks (CICN), 2013 5th International Conference.

[9] C. Vargas Ben´ıtez and H. Lopes. Parallel artificial bee colony algorithm approaches for protein structure prediction using the 3dhp-sc model. *Intelligent Distributed Computing IV,* pages 255-264, 2010.

[10] D. González- Álvarez M. Vega-Rodŕıguez, J. Gómez-Pulido, and J. Sánchez-Pérez. Finding motifs in dna sequences applying a multi-objective artificial bee colony (moabc) algorithm. *Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics,* pages 89-100, 2011.

[11] A. Bernardino, E. Bernardino, J. Sa´nchez-Pe´rez, J. Go´mez-Pulido, and M. Vega-Rodr´ıguez. Efficient load balancing for a resilient packet ring using artificial bee colony. *Applications of Evolutionary Computation,* pages 61–70, 2010.

[12]R.S. Rao, SVL Narasimham, and M. Ramalingaraju. Optimization of distribution network configuration for loss reduction using artificial bee colony algorithm. *International Journal of Electrical Power and Energy Systems Engineering,* 1(2):116-122, 2008.

[13] D. Karaboga and B. Akay. A modified artificial bee colony (abc) algorithm for constrained optimization problems. *Applied Soft Computing,* 11:3021-3031, 2011.

[14] P.W. Tsai, J.S. Pan, B.Y. Liao, and S.C. Chu. Enhanced artificial bee colony optimization. *International Journal of Innovative Computing, Information and Control,* 5(12), 2009.