

An Improved Artificial Bee Colony using Cultural Algorithm for Optimization Problem

Busayo Hadir Adebisi
Department of Electrical and
Computer Engineering,
Ahmadu Bello University Zaria,
Nigeria

Abdoulie Momodou Sunkary
Tekanyi
Department of Electrical and
Computer Engineering,
Ahmadu Bello University Zaria,
Nigeria

Tijani Ahmed Salawudeen
Department of Electrical and
Computer Engineering,
Ahmadu Bello University Zaria,
Nigeria

ABSTRACT

Artificial Bee Colony (ABC) algorithm is a global optimization algorithm which is motivated by the foraging behaviour of swarm of honey bee. The ABC has been successfully employed in solving many kind of complex engineering design problem. But due to the lack of imbalance between exploration and exploitation and insufficient guiding parameters ABC usually get stuck into local minima. In order to address this shortcoming a proper guiding parameter which has the ability to change dynamically depending on the nature of problem needs to be introduced. Therefore, this paper proposes an improved ABC algorithm using knowledge inherent in Cultural Algorithm (CA). Two new variants of ABC were developed using situation and normative knowledge. The performance of the developed variants was evaluated using a total of twenty (20) applied mathematical optimization benchmark functions. Simulation results clearly show that, all the newly proposed CABCA variants performed much better than the ABC.

General Terms

Artificial Bee Colony, Cultural Artificial Bee Colony Algorithm and solution search equation.

Keywords

Artificial Bee Colony, Cultural Algorithm, Exploration, Exploitation, variants, optimization and convergence.

1. INTRODUCTION

The Artificial Bee Colony (ABC) algorithm is a recently introduced swarm intelligence optimization algorithm which mimic the intelligent foraging behaviours of honey bees. Since its advent [6], ABC and its variants are often successfully employed to diverse ranges of optimization problems [1-4]. The ABC has the advantages of strong robustness, fast convergence and, fewer setting parameters. However, the evolving population of candidate solutions loses its diversity and explorative search early on, due to insufficient guiding parameters. This leads to blindness of search space and prematurely convergence and subsequently leading to local solution [5]. Another problem that is faced by the ABC algorithm is fitness stagnation, where all the candidate solutions fail to improve their fitness values for indeterminately continued iterations, even without any premature convergence around the local optimal. The risk of premature convergence and fitness stagnation problems usually occur with increase in exploitations and decrease in explorations [5]. But increasing the exploration may lead to unacceptable slow convergence speed. In order to address these shortcomings, this paper present a knowledge based

ABC called the CABAC using both situational and normative cultural knowledge.

2. LITERATURE REVIEW

Several researchers have proposed a various modification to ABC in order to address most of the shortcomings mentioned earlier. For example in the work of [6], a diversity strategy (DABC) was introduce to preserve sufficient amount of diversity among the candidate solutions by switching between exploration and exploitation. The EABC [5] is another variant of ABC algorithm that customizes the degree of exploitations and explorations at the individual solution level, separately for each candidate solution of the bee population. The JA ABC [7] enhanced the average fitness of the population by mutating poor possible solutions around the best possible solution. In the best-so-far method [8], onlooker bees compare the information from all employed bees to select the best-so-far candidate food source. This will bias the solution handled by onlooker bees towards the optimal solution. A chaotic local search method was applied to solve the accuracy problem of global optimal value [9]. This was done by increasing the numbers of the scout to enhance better initialization, which in turns improved the convergence of ABC algorithm.

It is evident that successful efforts have been made by scholars to address some of the problem associated with the ABC. However, experience of individuals and the knowledge derived from the parent as a result of evolution has not been efficiently utilized. In this research, this individual experience will be employed through cultural evolution process to propose new variations of ABC called the CABCA. This is expected to improve the original ABC algorithm but not necessarily complicate the algorithm.

3. ARTIFICIAL BEE COLONY (ABC) ALGORITHM AND CULTURAL ALGORITHM (CA)

3.1 The ABC algorithm

The standard ABC Algorithm consists of three kinds of bees; employed bees, onlooker bees and scout bees. Employed bees are responsible for exploiting the nectar sources explored and take the information obtained about the quality of the food source locations which they are exploiting to the waiting bees (onlooker bees). Onlooker bees wait in the hive and decide on a food source to be exploited based on the information received from the employed bees. Scouts bees randomly search the environment in order to find a new food source depending on an internal motivation or based on possible external traces. In the 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 profitability (fitness) of the associated solution [4]. The ABC as an iterative algorithm starts with the employed bee searching for food source within the neighbourhood of the food source. Suppose, an employed bee is currently positioned at a food source position. During this stage, each employed bee searches in the neighbourhood of its current position to produce new trial food source using (1), where $j \in \{1, 2, \dots, D\}$ and $k \in \{1, 2, \dots, SN\}$ are randomly picked indices, D is dimension of the problem, SN is the number of food positions and ϕ_{ij} is a uniform random value $[-1, 1]$ [1].

$$v_{ij} = x_{ij} + \phi_{ij}(x_{kj} - x_{ij}) \quad (1)$$

Thus, the new solution v_i is produced from x_i by perturbing its randomly picked j^{th} parameter and using the information of x_i and another randomly picked solution x_k . If v_i has better fitness than the old food position x_i , then x_i is replaced by v_i . Otherwise the previous position x_i is retained. For the problem of function optimization, where f is the function to be minimized, ABC computes the fitness of a candidate solution x_i using (2):

$$fitness(x_i) = \begin{cases} \frac{1}{(1 + f(x_i))} & \text{if } f(x_i) \geq 0 \\ 1 + |f(x_i)| & \text{otherwise} \end{cases} \quad (2)$$

After all the employed bees complete their search, they share their information related to the nectar amounts and the positions of their sources with the onlooker bees in the dance area. The dance area is where exchange of information about the quality of food sources occurs. An onlooker bee evaluates the nectar information taken from all employed bees and chooses a food source location with a probability related to its nectar amount. The probability P_i that the employed bee with food source x_i would be picked by an onlooker bee is computed using (3), making the probability P_i to be proportional to $fitness(x_i)$.

$$P_i = \frac{fitness(x_i)}{\sum_{n=0}^{SN} fitness(x_n)} \quad (3)$$

Like the employed bees, each onlooker bee also employs (2) to produce trial food source in the vicinity of its current food source position. If v_i has better fitness than x_i is replaced by v_i . Otherwise, x_i is discarded.

A scout bee is created only when a particular food source cannot be improved through a predetermined number of trials limit. The employed bee now becomes a scout bee and its

food source is positioned randomly across the search space using (4) where $j = 1, 2, \dots, D$ and x_j is the search space along the dimension [10].

$$x_{ij} = \min_j + rand(0,1)(\max_j - \min_j) \quad (4)$$

In a robust search process, exploration and exploitation processes must be carried out together in equal proportion. The onlooker and employed bees carried out the exploitation process in the search space, while the scouts are responsible for the exploration process. In the case of real honeybees, the recruitment rate represents a measure of how quickly the bee colony finds and exploits a newly discovered food source [1]. This recruiting could also represent the measurement of the speed with which the feasible solutions of the difficult optimization problems can be discovered.

3.2 The CA Algorithm

Cultural Algorithms (CAs), is an evolutionary model which are inspired by model of cultural evolution process. The CAs has been developed in order to model the evolution of the cultural component of an evolutionary computational system over time as it accumulates experience. CAs can provide an explicit mechanism for global knowledge and a useful framework within which to model self-adaptation in an evolutionary or swarm intelligence system [11-13] CAs model two levels of evolution (population level and belief space level). The population space consists of a set of possible solutions to the problem, and can be modelled by using any population-based optimization method. The belief space is the place where the information about the knowledge on the solution of the problem is developed and stored. The five basic categories of cultural knowledge have been identified: Normative, Situational, Domain, History, and Topographical Knowledge [12]. In this paper, the population component of the CA will be the ABC and the global knowledge that has been learned by the population will be expressed using situational and normative knowledge.

Situational knowledge represents the best individuals found at a certain time of evolution and it contains a number of individuals considered as a set of exemplars to the rest of the population. The Situational Knowledge is updated when the best individual of the population is found, that is at iteration t , $S = \langle S^t | S^t = \{s_1^t, s_2^t, \dots, s_n^t\} \rangle$. This situational knowledge equation can be represented as [14]:

$$S_j^{t+1} = \begin{cases} X_{best,j}^{t+1} & \text{if } f(X_{best,j}^{t+1}) < f(s^t) \\ s_j^t & \text{otherwise} \end{cases} \quad (5)$$

Normative knowledge provides standards for interpreting and determining individual behaviours through guidelines within which individual adjustments can be made. The normative component, N , is a set of interval information for each parameter. Each of the N intervals in the belief space is represented as a triple [11].

$$N = \langle I, L, U \rangle \quad (6)$$

Where I, L and U , are n -dimensional vectors, and I_j denotes the closed interval for variable j that is a continuous set of real numbers x represented as a number pair:

$$I_j = [l_j, u_j] = \{x | l_j \leq x \leq u_j\} \quad (7)$$

n is the number of the variables, l_j and u_j are the lower and upper bounds for the j^{th} variable, respectively, L_j and U_j are the values of the fitness function associated with the bound l_j and u_j are usually initialized with positive infinity.

Usually the normative knowledge leads individuals “to jump into the good range” if they are not there yet [12].

The normative knowledge is updated as follows [14]:

$$l_j^{t+1} = \begin{cases} x_{i,j}^t & \text{if } x_{i,j}^t \leq l_j^t \quad \text{or} \quad f(x_i^t) < L_j^t \\ l_j^t & \text{otherwise} \end{cases} \quad (8)$$

$$u_j^{t+1} = \begin{cases} x_{k,j}^t & \text{if } x_{k,j}^t \geq u_j^t \quad \text{or} \quad f(x_k^t) < U_j^t \\ u_j^t & \text{otherwise} \end{cases} \quad (9)$$

$$L_j^{t+1} = \begin{cases} f(X_i^t) & \text{if } x_{i,j}^t \leq l_j^t \quad \text{or} \quad f(X_i^t) < L_j^t \\ L_j^t & \text{otherwise} \end{cases} \quad (10)$$

$$U_j^{t+1} = \begin{cases} f(X_k^t) & \text{if } x_{k,j}^t \geq u_j^t \quad \text{or} \quad f(X_k^t) < U_j^t \\ U_j^t & \text{otherwise} \end{cases} \quad (11)$$

Where the i^{th} individual affects the lower bound for variable j , and the k^{th} individual affect the upper bound for variable j . It should be noted that t denotes the current generation of individuals.

The number of the individuals accepted for the update of the belief space is choice randomly of a part of population using the following expression [11, 14, 15].

$$\alpha = N \cdot \beta + \lfloor N \cdot \beta / t \rfloor \quad (12)$$

where N is the size of the population, t is the current number of generation and β is a parameter given by the user t (in this work, 0.2 is adopted). That is, top 20% of the population.

4. CULTURAL ARTIFICIAL BEE COLONY ALGORITHM (CABCA)

The knowledge will influence the evolutionary operation of ABC, in two ways: (1) determining the size of the changes, step size (2) and determining the direction of change. These two ways will serve as an adaptive and guiding influence of explorations and exploitations. Four different instantiations of

cultural algorithms are produced by using these different influence functions.

4.1 CABCA(Ns)

This is the simplest form of cultural influence. This version uses only the normative knowledge to determine the step size of evolution. For all components $i=1, \dots, t$ and $j=1, \dots, n$. This can be expresses mathematically as:

$$x_{ij}^{t+1} = x_{ij}^t + \left| size(I_j) \right| \times \phi_{ij} (x_{ij}^t - x_{kj}^t) \quad (13)$$

where $size(I_j) = u_j - l_j$ is the size of the belief space interval for parameter i , which is decided by normative knowledge for the i^{th} variable

4.2 CABCA(Sd)

In this variant only situational knowledge of the current exemplar or best solution found so far, is used to decide the direction of evolution. The variation is influenced as follows:

$$x_{ij}^{t+1} = \begin{cases} x_{ij}^t + \left| \phi_{ij} (x_{ij}^t - x_{kj}^t) \right| & \text{if } x_{ij}^t < s_j^t \\ x_{ij}^t - \left| \phi_{ij} (x_{ij}^t - x_{kj}^t) \right| & \text{if } x_{ij}^t > s_j^t \\ x_{ij}^t + \phi_{ij} (x_{ij}^t - x_{kj}^t) & \text{otherwise} \end{cases} \quad (14)$$

5. SIMULATION RESULTS

In order to evaluate the performance of the proposed CABCA, a total of twenty (20) applied mathematical optimization benchmark function were used. These benchmarks contain a collection of different types of functions such as separable, non-separable, unimodal and multimodal functions. The performance of CABCA has been compared with the standard ABC algorithm with reference to the global solution. The parameters setting of this work is indicated in Table 1. For each system and for each test case, the average of ten (20) tests performed using MATLAB R2015a were recorded. The simulations were performed on HP-Pavilion g7-2270us on an Intel(R) Core i3 with 2.40GHz processor and 6.00GB RAM with 64-bit Windows 8 pro Operating System (OS).

Table 1. Parameter Setting for ABC

S/N	Parameters	Values
1	Population size	50
2	Dimension	30
3	Maximum Cycle Number	5000
4	Limit	50

5.1 Comparisons of ABC with CABCA

In this section, ABC is compared with all the variants of the proposed CABCA variants. The t-test is introduced to clearly indicate the statistical significance of the algorithms.

- **ABC vs CABCA(Ns):** Out of the 20 functions, CABCA(Ns) performs better than ABC on 13 functions; while ABC performs better only on one (1) function. On the remaining 6 functions, their

results are similar (i.e., the performance difference is not statistically significant in t-tests with at least 99% degree of confidence). Thus the overall performance of CABCA(Ns) is better than standard ABC.

- **ABC vs CABCA(Sd):** It can be observed from Table 1 that on all of the 20 test functions, CABCA(Sd) produced the best result in 11 and

ABC has best performance in 2 functions. On the remaining 7 functions, their results are similar.

In total, the variants of CABCA (i.e. CABCA(Ns) and CABCA(Sd)) outperformed the standard ABC in all the optimization test functions. The superiority of CABCA justifies the essence of knowledge introduction in the belief space for self-adaptation. This has substantially improved the performance of the standard ABC in the search for global solution.

Table 2: Performance of the proposed algorithm CABCA, compared to the basic ABC algorithm on the benchmark functions. Results are averaged over 20 independent runs. Better performance by CABCA variants is marked with boldface font. In case the performance difference is not significant by t-Test with at least 99% level of confidence (i.e., $\lambda = 0.99$), it is marked as “Similar”.

Test Function	G_{min}	ABC	CABCA(Ns)	CABCA(Sd)	Best Performance (t-Test with $\lambda = 0.99$)
Ackley	0.000E+00	2.1157E-13	4.1466E-15	1.9851E-13	CABCA(Ns)
Axis parallel	0.0000E+00	9.5846E-16	1.0472E-16	8.7434E-16	CABCA(Ns)
CM	-3.0000E+00	-3.0000E+00	-3.0000E+00	-3.0000E+00	Similar
DeJongf4	0.0000E+00	5.4743E-16	4.3421E-16	5.1616E-16	CABCA(Ns)
ExpFun	1.0000E+00	1.0000E+00	0.0000E+00	1.0000E+00	Similar
Griewangk	0.0000E+00	0.0000E+00	1.0000E+00	0.0000E+00	Similar
Hyperelliptic	0.0000E+00	1.0000E-15	8.8185E-43	9.1751E-16	CABCA(Ns)
Michalewicz	-9.6602E+00	-26.5754E+00	-2.3832E-13	-24.3579E+00	CABCA(Sd)
Neumaier3	-4.9300E+03	-2.8331E+03	-2.9000E-04	-2.9523E+03	CABCA(Sd)
PM1	0.0000E+00	1.2349E-15	3.7832E-27	1.2310E-15	CABCA(Ns)
PM2	0.0000E+00	1.55319E-11	2.5959E-27	4.4354E-12	Similar
Quartic	0.0000E+00	10.9770E+00	8.4591E-01	8.3048E+00	CABCA(Ns)
Rastrigin	0.0000E+00	2.3628E-12	1.1806E-13	2.1652E-12	CABCA(Ns)
Rosenbrock	0.0000E+00	1.1236E+00	5.9490E-02	1.1179E+00	CABCA(Ns)
Sal	0.0000E+00	9.9873E-02	9.9873E-02	9.9873E-02	Similar
Schaffer	0.0000E+00	5.7452E-15	7.4541E-19	6.5000E-15	CABCA(Ns)
Schwefel	0.0000E+00	3.5536E+03	3.5542E+03	3.5535E+03	Similar
Sphere	0.0000E+00	1.0875E-15	9.8378E-16	9.1795E-16	CABCA(Sd)
Step	0.0000E+00	0.0000E+00	0.0000E+00	0.0000E+00	Similar
Zakharov	0.0000E+00	6.8857E-13	1.1469E-15	1.3933E-15	CABCA(Sd)

6. CONCLUSION AND FUTURE WORKS

This paper introduces CABCA — a novel modified of the basic ABC algorithm. The normative and situational knowledge inherent in cultural algorithm were utilized to guide the step size as well as the direction of evolution of ABC, in order to combat the disparity between exploration and exploitation associated with the basic ABC, which results to poor convergence and optimization efficiency. Two new variants CABCA(Ns) and CABCA(Sd) were developed in MATLAB R2015a and the performance of the developed variants were evaluated using a total of twenty applied mathematical optimization test functions (such as Ackley, Michalewicz, Step, etc). Simulation result obtained shows that the two variants of CABCA has a better precession accuracy and higher ability of escaping local minimal when compared

with the basic ABC. It would be interesting to study how well CABCA performs on many other existing problems, especially the discrete and real world ones. Some interesting examples of discrete and real world problems on which CABCA could be employed are industrial process control [16], telecommunications [17], science and engineering analysis and design [18] and many others.

7. REFERENCES

- [1] Karaboga, D. and B. Basturk, *On the performance of artificial bee colony (ABC) algorithm*. Applied soft computing, 2008. **8**(1): p. 687-697.
- [2] Hsieh, T.-J., H.-F. Hsiao, and W.-C. Yeh, *Forecasting stock markets using wavelet transforms and recurrent neural networks: An integrated system based on artificial*

- bee colony algorithm*. Applied soft computing, 2011. **11**(2): p. 2510-2525.
- [3] Gao, W. and S. Liu, *Improved artificial bee colony algorithm for global optimization*. Information Processing Letters, 2011. **111**(17): p. 871-882.
- [4] Karaboga, D., *An idea based on honey bee swarm for numerical optimization*. 2005, Technical report-tr06, Erciyes university, engineering faculty, computer engineering department.
- [5] Shapla, S.S., H. Haque, and M.S. Alam, *Explorative Artificial Bee Colony Algorithm: A Novel Swarm Intelligence Based Algorithm For Continuous Function Optimization*. International Journal of Science and Research (IJSR), 2015. **4**(7).
- [6] Lee, W.-P. and W.-T. Cai. *A novel artificial bee colony algorithm with diversity strategy*. in *Natural Computation (ICNC), 2011 Seventh International Conference on*. 2011. IEEE.
- [7] Sulaiman, N., J. Mohamad-Saleh, and A.G. Abro. *A modified artificial bee colony (JA-ABC) optimization algorithm*. in *Proceedings of the International Conference on Applied Mathematics and Computational Methods in Engineering*. 2013.
- [8] Banharsakun, A., T. Achalakul, and B. Sirinaovakul, *The best-so-far selection in artificial bee colony algorithm*. Applied Soft Computing, 2011. **11**(2): p. 2888-2901.
- [9] Yan, G. and C. Li, *An effective refinement artificial bee colony optimization algorithm based on chaotic search and application for pid control tuning*. Journal of Computational Information Systems, 2011. **7**(9): p. 3309-3316.
- [10] Alam, M.S., M.M. Islam, and K. Murase, *Artificial Bee Colony Algorithm with Adaptive Explorations and Exploitations: A Novel Approach for Continuous Optimization*.
- [11] Reynolds, R.G. and C. Chung. *Fuzzy approaches to acquiring experimental knowledge in cultural algorithms*. in *Tools with Artificial Intelligence, 1997. Proceedings., Ninth IEEE International Conference on*. 1997. IEEE.
- [12] Reynolds, R.G. and B. Peng. *Cultural algorithms: modeling of how cultures learn to solve problems*. in *Tools with Artificial Intelligence, 2004. ICTAI 2004. 16th IEEE International Conference on*. 2004. IEEE.
- [13] Reynolds, R.G. and B. Peng, *Cultural algorithms: computational modeling of how cultures learn to solve problems: an engineering example*. Cybernetics and Systems: An International Journal, 2005. **36**(8): p. 753-771.
- [14] Chung, C.-J., *Knowledge-based approaches to self-adaptation in cultural algorithms*. 1997.
- [15] Salawudeen, A.T., *Development of an Improved Cultural Artificial Fish Swarm Algorithm with Crossover*. 2015.
- [16] El-Telbany, M.E., *Tuning PID controller for DC motor: An artificial bees optimization approach*. International Journal of Computer Applications, 2013. **77**(15).
- [17] Ozen, A. and C. Ozturk. *A novel modulation recognition technique based on artificial bee colony algorithm in the presence of multipath fading channels*. in *Telecommunications and Signal Processing (TSP), 2013 36th International Conference on*. 2013. IEEE.
- [18] Akay, B. and D. Karaboga, *Artificial bee colony algorithm for large-scale problems and engineering design optimization*. Journal of Intelligent Manufacturing, 2012. **23**(4): p. 1001-1014