

Performance Optimization of Benchmark Functions Using VTS-ABC Algorithm

Twinkle Gupta¹ and Dharmender Kumar²

Submitted in July, 2014; Accepted in September, 2014

Abstract - A new variant based on tournament selection called VTS-ABC algorithm is provided in this paper. Its performance is compared with standard ABC algorithm with different size of data on several Benchmark functions and results show that VTS-ABC provides better quality of solution than original ABC algorithm in every case.

Index Terms – Artificial Bee Colony Algorithms, Nature-Inspired Meta-heuristics, Optimizations, Swarm Intelligence Algorithms, Tournament selection.

NOMENCLATURE

ABC – Artificial Bee Colony
ACO – Ant Colony Optimization
BFS – Blocking Flow-Shop Scheduling
DE – Differential Evolution
EA – Evolutionary Algorithm
GA – Genetic Algorithm
MCN – Maximum Cycle Number
PSO – Particle Swarm Optimization
TS – Tournament size
TSP – Travelling Salesman Problem

1.0 INTRODUCTION

For optimization problems, various algorithms have been designed which are base donnature-inspired concepts [1]. Evolutionary algorithms (EA) and swarm optimization algorithms are two different classes in which nature inspired algorithms are classified. Evolutionary algorithms like Genetic algorithms (GA) and Differential evolution (DE) attempt to carry out the phenomenon of natural evolution [2]. However, a swarm like ant colony, a flock of birds can be described as collection of interacting agents and their intelligence liein their way of interactions with other individuals and the environment [3]. Swarm optimization includes Particle swarm optimization (PSO) model on social behavior of bird flocking [4], Ant colony optimization (ACO) model on swarm of ants and Artificial Bee Colony (ABC) model on the intelligent foraging behaviour of honey bees [5]. Some important characteristics of ABC algorithm which makes it more attractive the another optimizational algorithms are:

- 1) Employs only three control parameters (population size, maximum cycle number and limit) [6].

¹Master of Technology (CSE), Guru Jambheshwar University of Science and Technology, Hisar, Haryana, India.

²Associate Professor (CSE), Guru Jambheshwar University of Science and Technology, Hisar, Haryana, India.

E-mail: ¹twinkle2803.gupta@gmail.com and

²dharmindia24@gmail.com

- 2) Fastconvergencespeed.
- 3) Quite simple, flexible and robust [7] [8].
- 4) Easyintegrationwiththeroptimizationalgorithms.

Therefore, ABC algorithm is a very popular nature inspired meta-heuristic algorithm used to solve various kinds of optimization problems. In recent years, ABC has earned so much popularity and used widely in various application such as: Constrained optimization, Image processing, Clustering, Engineering Design, Blocking flow shop scheduling (BFS), TSP, Bioinformatics, Scheduling and many others [9]-[18]. Similar to other stochastic population-based approaches like GA, Ant Colony etc. ABC algorithm also applied Roulette Wheel selection mechanism which chooses best solution always with high selection pressure and leads the algorithm into premature convergence. With ever-growing size of dataset, optimization of algorithm has become a big concern. This calls for a need of better algorithm.

The aim of this paper is to create such an algorithm named VTS-ABC algorithm. This new variant is based on tournament selection mechanism and selects variable tournament size each time in order to select the employed bees sharing their information with onlooker bees. Onlooker bees select solution from selected tournament size of solutions with less selection pressure so that high fitness solutions can't dominate and give better quality of solutions with large data set as well. A worst solution is also replaced by better solution generated randomly in each cycle.

Rest of the paper is divided in different sections as follows: Introduction to standard ABC algorithm is described in section 2. Section 3 describes the proposed VTS-ABC algorithm. Experiments and its simulation results to show performance on several Benchmark functions are described in section 4 and in the last; Conclusion of the paper is discussed.

2. ARTIFICIAL BEE COLONY ALGORITHM

In 2005, Karaboga firstly proposed Artificial Bee Colony algorithm for optimizing numerical problems [19] which includes employed bees, onlooker bees and scouts. The bee carrying out search randomly is known as a scout. The bee going to the food source visited by it before and sharing its information with onlooker bees is known as employed bee and the bee waiting on the dance area called onlooker bee. ABC algorithm as a collective intelligence searching model has three essential components: Employed bees, Unemployed bees (onlooker and scout bees) and Food sources. In the view of optimization problem, a food source represents a possible solution. The position of a good food source indicates the solution providing better results to the given optimization problem. The quality of nectar of a food source represents the fitness value of the associated solution.

Initially, a randomly distributed food source position of SNsize, the size of employed bees or onlooker bees is generated. Each solution x_i is a D-dimensional vector that represents the number of optimized parameters and produced using the equation 1:

$$x_{i,j} = x_{min,j} + rand(0,1)(x_{max,j} - x_{min,j}) \quad (1)$$

where, x_{max} and x_{min} are the upper and lower bound of the parameter x_i , respectively and j denotes the dimension. The fitness of food sources to find the global optimal is calculated by the following formula:

$$fit_m(x_m) = \begin{cases} \frac{1}{1 + f_m(x_m)} & , f_m(x_m) > 0 \\ 1 + |f_m(x_m)| & , f_m(x_m) < 0 \end{cases} \quad (2)$$

where, $f_m(x_m)$ is the objective function value of x_m . Then the employed bee phase starts. In this phase, each employed bee x_i finds a new food source v_i in its neighborhood using the equation 3:

$$x_{i,j}(t + 1) = x_{i,j}(t) + \phi(x_{i,j}(t) - x_{k,j}(t)) \quad (3)$$

where, t : Cycle number; x_k : Randomly chosen employed bee and k is not equal to i ; $\phi()$: A series of random variable in the range $[-1, 1]$. The fitness of new solution produced is compared with that of current solution and memorizes the better one by means of a greedy selection mechanism.

Employed bees share their information about food sources with onlooker bees waiting in the hive and onlooker bees probabilistically choose their food sources using fitness based selection technique such as roulette wheel selection shown in equation 4:

$$P_i = \frac{F(\theta_i)}{\sum_{k=1}^S F(\theta_k)} \quad (4)$$

where, P_i : Probability of selecting the i^{th} employed bee, S : Size of employed bees, θ_i : Position of the i^{th} employed bee and $F(\theta_i)$: Fitness value. After that onlooker bees carried out randomly search in their neighborhood similar to employed bees and memorize the better one.

Employed bees whose solutions can't be improved through a predetermined number of cycles, called limit, become scouts and their solutions are abandoned. Then, they find a new random food source position using the following equation 5:

$$x_{i,j} = x_{jmin} + r \cdot (x_{jmax} - x_{jmin}) \quad (5)$$

Where, r : A random number between 0 and 1 and these steps are repeated through a predetermined number of cycles called Maximum Cycle Number (MCN).

3. PROPOSED WORK: VTS-ABC ALGORITHM

In every meta-heuristic algorithm mainly two factors need to be balanced for global optimization outcome i.e. Exploration and Exploitation but ABC is a poor balance of these two factors. Various variants of ABC have been modelled for its improvement in different phases by number of researchers like Sharma and Pant have proposed a variant of ABC called RABC for solving the numerical optimization problem [20] and Tsai et al. have presented an interactive ABC optimization algorithm to solve combinational optimization problem [21] in which the

concept of universal gravitational force for the movement of onlooker bees is introduced to enhance the exploration ability of the ABC algorithm. D. Kumar and B. Kumar also reviewed various papers on ABC and give a modified RABC algorithm based on topology for optimization of benchmark functions [22] [23].

Intelligence of ABC algorithm mainly depends upon the communication between individual agents. Employed bees share their information with onlooker bees waiting in the hive and flow of this information from one individual to another depends on the selection mechanism used. Different selection schemes select different individuals to share the information which affect the communication ability of individuals and primarily the outcome of the algorithm. ABC algorithm uses Roulette wheel selection mechanism in which each onlooker bee selects the food source based on certain probability. Each onlooker bee selects the best food source with high selection pressure and lead to premature convergence. To overcome this problem, its new variant is proposed in which Tournament Selection method is applied based on Cycle number and number of employed bees.

In Tournament selection, a tournament size (TS) is chosen to select the number of employed bees sharing the information with onlooker bees. For better exploration, $TS=2$ i.e. Binary Tournament is applied in early stages and for better exploitation, variable tournament size is applied based on the current cycle number (CYL) and size of employed bee in middle stages. As the stages grow, this method works similar to Roulette wheel method in the end. Hence, the selection pressure is less in early stages and more in final stages which provide us better quality of solution. As variable size of tournament is used at different stages of the algorithm, hence the algorithm named VTS-ABC (Variable Tournament Size Artificial Bee Colony) algorithm. Method used for calculating TS is shown in equation 6 and equation 7:

If $SN \geq 20$

$TS =$

$$\begin{cases} (SN) * \frac{i}{10}, & \text{if } MCN * \frac{i-1}{10} \leq CYL \leq MCN * \frac{i}{10} \\ & \text{and } i = 1, 2 \dots 10 \end{cases} \quad (6)$$

If $SN < 20$

$TS =$

$$\begin{cases} 2, & \text{if } CYL \leq \frac{MCN}{5} \\ TS + 1, & \text{if } \frac{MCN}{5} * i < CYL < \frac{MCN}{5} * (i + 1) \\ & \text{and } TS < SN \text{ and } SN < 10 \\ TS + b, & \text{if } \frac{MCN * i}{5} < CYL < \frac{MCN * (i + 1)}{5} \\ & \text{and } TS < SN \text{ and } 10 \leq SN \leq 20 \\ & SN, & \text{if } CYL < MCN \end{cases} \quad (7)$$

Where $b = \frac{SN - mod(SN,5)}{5}$

Here, two equations are shown for calculating tournament size of tournament selection method. The purpose of using these two equations is to increase the speed of algorithm. When the

size of employed bee i.e. given population of food source positions is small like 10, a solution can be easily found by changing the tournament size by 1 but as the size grows i.e. when best food source position is to be found in large set of population for example when SN=40 or more than 40, increasing size of tournament by 1 and 2 only is a very tedious task as it will take more time to run the algorithm. Hence, in order to increase speed of algorithm, the tournament size based on current cycle and size of population is increased.

One more concept is applied to increase its convergence speed. At each iteration or cycle, a new solution is generated randomly similar to scout and its fitness value is calculated. Greedy selection mechanism is applied between new solution and worst one and the better solution is memorized. Hence, it helps in finding good quality of solution as well as improving the convergence speed and provides better balance between exploration and exploitation.

4. EXPERIMENTS AND SIMULATION RESULTS

4.1 Benchmark Functions

The Benchmark Functions used to compare the performance of VTS-ABC algorithm with original ABC algorithm are illustrated below:

1) Sphere Function:

$$ObjVal = \sum_{j=1:p} (X^2)'$$

2) Schwefel Function:

$$ObjVal = \sum_{j=1:p} ObjVal + (X(1) - X(j))^2 + (X(1) - 1)^2$$

3) Griewank Function:

$$ObjVal = \left(\sum \left(\frac{X^2}{4000} \right) \right)' - \prod \left(\cos \left(\frac{X}{\sqrt{\alpha}} \right) \right)' + 1$$

Where $\alpha = repmat(1:p, [n 1])$

4) Ackley Function:

$$ObjVal = -20 * e^{-0.2 * \sqrt{\frac{1}{p} * (\sum (X^2))'}} - e^{\frac{1}{p} * (\sum (\cos(2 * \pi * X)))'} + 20 + e^1$$

Here, ObjVal is the function value calculated for each food source position. A food source is represented by X and population size is taken of n*p matrix where n is the no. of possible food source positions and p represents the dimension of each position.

4.2 Performance Measures & Simulation Result

The experimental results of VTS-ABC and ABC algorithm in MATLAB are taken under the parameter of size of food source positions (n*p) i.e. different size of population with different dimensions are taken to run and compare both algorithms. MCN is set as 2000 and each algorithm is run for 3 iteration i.e. Runtime=3. Limit for scouts is set equals to 300. In order to provide the quantitative assessment of the performance of an optimization algorithm, Mean of Global Minimum i.e. mean of minimum objective function value at each cycle of all iterations

are taken as performance measure whose values are shown in table1 and figure 1-4.

Benchmark function	Algorithm	20*10	100*100	150*100
Sphere	ABC	0.3404 62	13.0503	0.012460 8
	VTS-ABC	0.0988 222	6.70776	0.011053
Schwefel	ABC	1.2323 1	0.107861	19.0437
	VTS-ABC	0.7290 75	0.107592	14.3503
Griewank	ABC	0.0619 326	0.00052670 3	0.447714
	VTS-ABC	0.0146 616	0.00043907	0.189238
Ackley	ABC	1.5651 3	0.0648988	2.57993
	VTS-ABC	0.9468 86	0.0604899	2.13692

Table1: Mean of Global minimum on different size of data

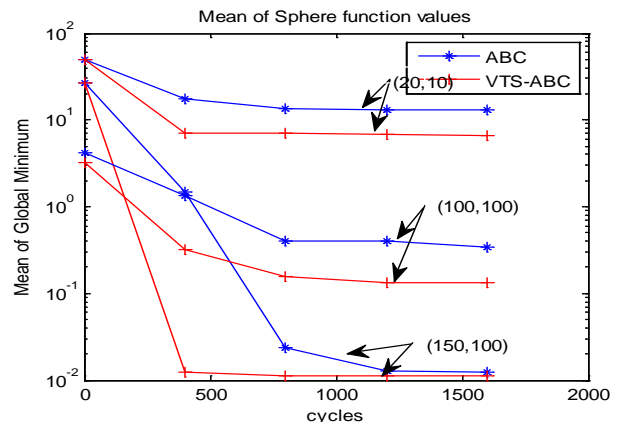


Figure 1: Mean of Sphere function values on different size of data

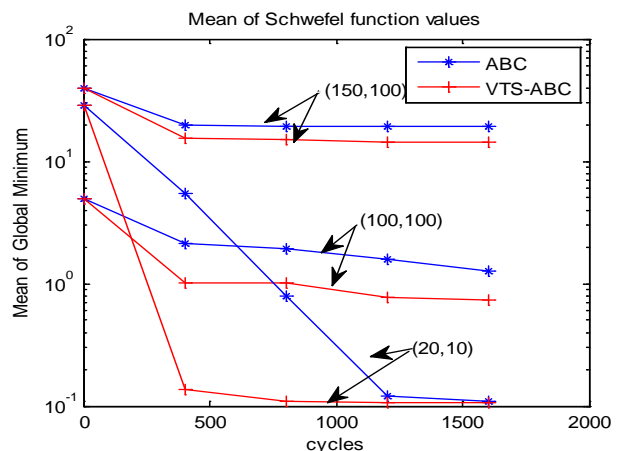


Figure 2: Mean of Schwefel function values on different size of data

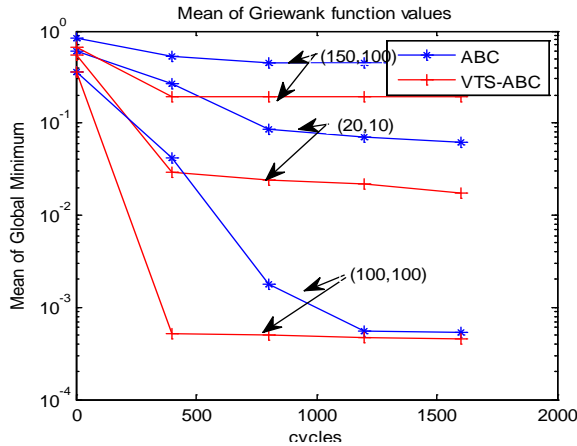


Figure 3: Mean of Griewank function values on different size of data

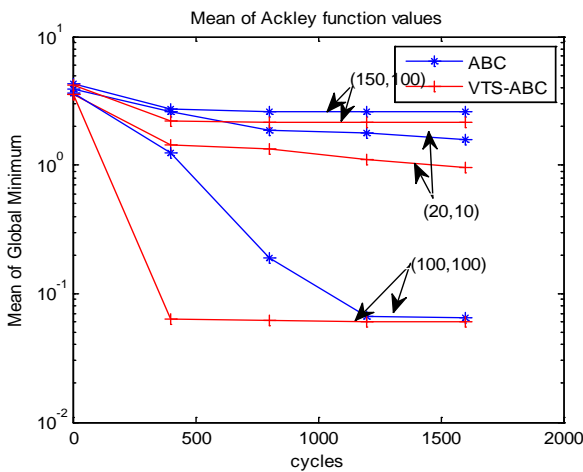


Figure 4: Mean of Ackley function values on different size of data

Figure 1 to 4 show simulation results of ABC and VTS-ABC algorithm with different size of data on Sphere, Schwefel, Griewank, Ackley respectively and reveal that VTS-ABC algorithm provides us better quality of solution than original ABC algorithm by minimizing objective function value or producing higher fitness solutions.

5. DISCUSSION AND CONCLUSION

In this paper, a new algorithm VTS-ABC is presented. In this algorithm, firstly variable tournament size (TS) is applied to select the food source position for onlooker bees which helps to achieve diversity in solution. Then to increase convergence speed, a new solution is generated in each cycle which replaced the worst one. In order to demonstrate the performance of proposed algorithm, it is applied on several Benchmark functions with different size of data set as input. Simulation results show that it provides better quality of solution than original ABC algorithm in every case. Therefore, it can be applied in different fields of optimization with large and higher dimensions data set efficiently.

6.0 REFERENCES

- [1]. Yugal Kumar and Dharmender Kumar, "Parametric Analysis of Nature Inspired Optimization Techniques" International Journal of Computer Applications, vol. 32, no. 3, pp. 42-49, Oct. 2011.
- [2]. P. J. Angeline, J. B. Pollack and G.M. Saunders, "An evolutionary algorithm that constructs recurrent neural networks," Neural Networks in IEEE Transactions on, vol. 5, no. 1, 1994, pp. 54-65.
- [3]. J. Kennedy and R. Eberhart, "Particle swarm optimization," in Proceedings of IEEE international conference on neural networks, 1995, vol. 4, pp. 1942-1948.
- [4]. E. Bonabeau, M. Dorigio, and G. Theraulaz, "Swarm intelligence: from neural network to artificial intelligence," NY: oxford university press, New York, 1999.
- [5]. D. Karaboga, "An idea based on honey bee swarm for numerical optimization," Techn.Rep. TR06, Erciyes Univ. Press, Erciyes, 2005.
- [6]. D. Karaboga and B. Akay, "A comparative study of artificial bee colony algorithm," Applied Mathematics and Computation, vol. 214, no. 1, pp. 108-132, 2009.
- [7]. R. S. Rao, S. V. L. 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, vol. 1, no.2, pp. 116-122, 2008.
- [8]. A. Singh, "An artificial bee colony algorithm for the leaf-constrained minimum spanning tree problem," Applied Soft Computing, vol. 9, no. 2, pp. 625-631, Mar. 2009.
- [9]. D. Karaboga and B. Basturk, "Artificial bee colony (ABC) optimization algorithm for solving constrained optimization problems," in Foundations of Fuzzy Logic and Soft Computing, Springer, 2007, pp. 789-798.
- [10]. C. Chidambaram and H. S. Lopes, "A new approach for template matching in digital images using an Artificial Bee Colony Algorithm," in World Congress on Nature Biologically Inspired Computing, 2009. NaBIC 2009, IEEE, 2009, pp. 146-151.
- [11]. N. K. Kaur Mann, "Review Paper on Clustering Techniques," Global Journal of Computer Science and Technology, vol. 13, no. 5, 2013.
- [12]. S. Okdem, D. Karaboga, and C. Ozturk, "An application of Wireless Sensor Network routing based on Artificial Bee Colony Algorithm," in 2011 IEEE Congress on Evolutionary Computation (CEC), 2011, pp. 326-330.
- [13]. T. K. Sharma, M. Pant, and J. C. Bansal, "Some modifications to enhance the performance of Artificial Bee Colony," in 2012 IEEE Congress on Evolutionary Computation (CEC), 2012, pp. 1-8.
- [14]. L. Bao and J. Zeng, "Comparison and analysis of the

- selection mechanism in the artificial bee colony algorithm,” in *Hybrid Intelligent Systems, 2009. HIS'09. Ninth International Conference on*, 2009, vol. 1, pp. 411–41.
- [15]. C. M. V. Benítez and H. S. Lopes, “Parallel Artificial Bee Colony Algorithm Approaches for Protein Structure Prediction Using the 3DHP-SC Model,” in *Intelligent Distributed Computing IV*, M. Essaïdi, M. Malgeri, and C. Badica, Eds. Springer Berlin Heidelberg, 2010, pp. 255–264.
- [16]. D. L. González-Álvarez, M. A. Vega-Rodríguez, J. A. Gómez-Pulido, and J. M. Sánchez-Pérez, “Finding Motifs in DNA Sequences Applying a Multiobjective Artificial Bee Colony (MOABC) Algorithm,” in *Evolutionary Computation, Machine Learning and Data Mining in Bioinformatics*, C. Pizzuti, M. D. Ritchie, and M. Giacobini, Eds. Springer Berlin Heidelberg, 2011, pp. 89–100.
- [17]. L. Wang, G. Zhou, Y. Xu, S. Wang, and M. Liu, “An effective artificial bee colony algorithm for the flexible job-shop scheduling problem,” *Int J Adv Manuf Technol*, vol. 60, no. 1–4, pp. 303–315, Apr. 2012.
- [18]. S.-W. Lin and K.-C. Ying, “Increasing the total net revenue for single machine order acceptance and scheduling problems using an artificial bee colony algorithm,” *J Oper Res Soc*, vol. 64, no. 2, pp. 293–311, Feb. 2013.
- [19]. D. Karaboga, “An idea based on honey bee swarm for numerical optimization,” *Techn.Rep. TR06*, Erciyes Univ. Press, Erciyes, 2005.
- [20]. T. K. Sharma, M. Pant, and J. C. Bansal, “Some modifications to enhance the performance of Artificial Bee Colony,” in *2012 IEEE Congress on Evolutionary Computation (CEC)*, 2012, pp. 1–8.
- [21]. TSai, Pei-Wei, et al. , “Enhanced artificial bee colony optimization.” *International Journal of Innovative Computing, Information and Control*, vol. 5, no. 12, 2009, pp.5081-5092.
- [22]. B. K. Verma and D. Kumar, “A review on Artificial Bee Colony algorithm,” *International Journal of Engineering & Technology*, vol. 2, no. 3, pp. 175–186, 2013.
- [23]. D. Kumar and B. Kumar, “Optimization of Benchmark Functions Using Artificial Bee Colony (ABC) Algorithm,” *IOSR Journal of Engineering*, vol. 3, no. 10, pp. 09-14, October 2013.