ISSN:0975-9646

Sandeep Kumar et al, / (IJCSIT) International Journal of Computer Science and Information Technologies, Vol. 5 (2) , 2014, 1237-1247

An Improved Memetic Search in Artificial Bee Colony Algorithm

Sandeep Kumar , Vivek Kumar Sharma , Rajani Kumari

Faculty of Engineering and Technology
Jagannath University, Jaipur, India-303901

Abstract— **Artificial Bee Colony (ABC) is a swarm optimization technique. This algorithm generally used to solve nonlinear and complex problems. ABC is one of the simplest and up to date population based probabilistic strategy for global optimization. Analogous to other population based algorithms, ABC also has some drawbacks computationally pricey due to its sluggish temperament of search procedure. The solution search equation of ABC is notably motivated by a haphazard quantity which facilitates in exploration at the cost of exploitation of the search space. Due to the large step size in the solution search equation of ABC there are chances of skipping the factual solution are higher. For that reason, this paper introduces a new search strategy in order to balance the diversity and convergence capability of the ABC. Both employed bee phase and onlooker bee phase are improved with help of a local search strategy stimulated by memetic algorithm. This paper also proposes a new strategy for fitness calculation and probability calculation. The proposed algorithm is named as Improved Memetic Search in ABC (IMeABC). It is tested over 13 impartial benchmark functions of different complexities and two real word problems are also considered to prove proposed algorithms superiority over original ABC algorithm and its recent variants.**

***Keywords*— Artificial bee colony algorithm, Swarm intelligence, Evolutionary computation, Memetic algorithm**

I. INTRODUCTION

Nature Inspired Algorithms (NIAs) are most popular strategies that are used to get to the bottom of optimization problems during the earlier period. Swarm Intelligence one of these emerging and interesting algorithms. It mimics the collective and cooperative behaviour of social creatures. Swarm based optimization techniques find solution by collaborative hit and trial method. Social elements make the most of their ability of social learning to solve very complex and hard problems. The main hidden agenda behind the perfection of these swarm based optimization algorithms is their one to one social learning behaviour. Based on such social learning behaviour a number of algorithms are developed that are in use to solve a large number of problems like, non-linear, non-convex or discrete optimization problems. Research [1]-[4] in last two decade has shown that swarm intelligence based algorithms have enormous prospective to locate solutions of real world optimization problems that are measured as tough. The algorithms that have emerged in recent years consist of ant colony optimization (ACO) [1], artificial bee colony algorithm [6], particle swarm optimization (PSO) [2], bacterial foraging optimization (BFO) [5], firefly algorithm [54] and spider monkey optimization algorithm [55, 57] etc.

Artificial bee colony (ABC) optimization algorithm introduced by Dervis Karaboga [6] is am up to date accumulation in this class. ABC algorithm is inspired by the intelligent social behaviour of honey bees swarm when seeking a quality food source. Reminiscent of other population based optimization techniques ABC algorithm also has inhabitants of impending solutions. The impending solutions are food sources of honey bee insects. The fitness is measured in terms of the quality (nectar amount) of the food source. ABC algorithm is moderately a straightforward, high-speed and population based stochastic search procedure in the field of NIAs. There are two contradictory processes which drive the swarm to keep informed in ABC: the variation process, which enables exploring different areas of the search space, and the selection process, which make certain the exploitation of the previous experience. However, it has been shown that the ABC may infrequently bring to an end while proceeding in the direction of the global optimum even though the population has not converged to a local optimum [7]. It can be observed that the solution search equation of ABC algorithm is good at exploration but a little bit poor at exploitation [8]. For that reason, to maintain the proper balance between exploration and exploitation behaviour of ABC, it is exceedingly obligatory to develop a local search approach in addition to the actual ABC to exploit the search region. In past, very few efforts have been made on this trend. Kang et al. [9] proposed a Hooke Jeeves Artificial Bee Colony algorithm (HJABC) for numerical optimization. In HJABC, authors incorporated a local search technique which is based on Hooke Jeeves method (HJ) [10] with the basic ABC and solve slope stability analysis problem together with a wide range of dimensions.

In this paper, a modified memetic search strategy is proposed. The proposed local search strategy is used in place of employed bee and onlooker bee phase. Additional, the proposed algorithm is compared by experimenting on 13 un-biased test problems (i.e. the problems which solutions do not exist at starting point, axes or diagonal) and two real world problems to the basic ABC and its recent variants named, Memetic ABC (MeABC) [12], Randomized Memetic ABC (RMABC) [13], Modified ABC (MABC) [14], Enhanced local search in ABC [53] and Improved Onlooker bee phase in ABC (IoABC) [15].

Rest of the paper is organized as follows: Sect. 2 describes brief overview of the basic ABC. Section 3 discusses some recent modifications in ABC algorithm. Memetic algorithms explained in Sect. 4. Improved memetic search in ABC (IMeABC) is proposed and tested in Sect. 5. In Sect. 6, a comprehensive set of experimental results are provided. As a final point, in Sect. 7, paper is concluded.

II. ARTIFICIAL BEE COLONY ALGORITHM

The ABC algorithm is moderately recent swarm intelligence based algorithm. The algorithm is inspired by the intelligent food foraging behaviour of honey bees. In ABC, each solution of the problem is called food source of honey bee swarm. The fitness is determined in terms of the excellence of the food source. In ABC, honey bees are classified into three groups that is to say employed bees, onlooker bees and scout bees. The quantity of employed bees is identical to the onlooker bees. The employed bees are the bees which search the food source and congregate the information about the quality of the food source. Onlooker bees are bees which stay in the hive and search for the food sources based on the information collected by the employed bees. The scout bees are those bee which searches for new food sources haphazardly in places of the forsaken foods sources. Analogous to the other population-based algorithms, ABC solution search progression is an iterative procedure. After, initialization of all the ABC parameters and population of bees, it requires the recurring iterations of the three phases namely employed bee phase, onlooker bee phase and scout bee phase. Each of the phases is discussed in detail in subsequent subsections:

A. Resemblance of Artificial Bee Colony Algorithm and Honey bees

The original model of ABC algorithm planned by D. Karaboga [7] is composed of three major elements: employed and unemployed foragers, and food sources. The employed bees are ally with a fitting food source. Employed bees have intimate knowledge about food source. Exploitation of food sources done by employed bees. When a food source deselected (due to poor fitness/quality) employed bee turn out to be unemployed. The unemployed foragers are bees having no information about food sources and searching for a food source to exploit it. One can classify unemployed bees in two categories: scout bees and onlooker bees. Scout bees search at random for new food sources adjacent the beehive. Onlooker bees detect the waggle dance in hive, to opt for a food source for exploitation purpose. The third element is the prosperous food sources close to their hive. In the context of optimization, the number of food sources (that is the employed or onlooker bees) in ABC algorithm, is equivalent to the number of solutions in the population. Moreover, the location of a food source represents the arrangement of a favourable solution to the optimization problem, since the trait of nectar of a food source represents the fitness or quality of the correlated solution.

B. Phases of Artificial Bee Colony Algorithm

The search process of ABC follow three major steps [7]:

Employed bee phase: it sends the employed bees to a food source and calculates the nectar quality.

Onlooker bee phase: onlooker bees select the food sources after gathering information from employed bees and calculating the nectar quality.

Scout bee phase: determine the scout bees and employ them on promising food sources.

The location of the food sources are capriciously selected by the bees at the initial stage and their nectar qualities are measured. The employed bees then share the nectar information of the sources with the onlooker bees waiting at the dance area within the hive. After sharing this information, each employed bee come back to the food source confirmed during the previous cycle, as the location of the food source had been recalled and then selects new food source using its observed information in the neighbourhood of the present food source. At the last stage, an onlooker bee uses the information retrieved from the employed bees at the dance area to select a good food source. The possibility for the food sources to be elected boosts with boost in its quality of nectar. Hence, the employed bee with information of a food source with the highest quality of nectar employs the onlookers to that food source. It afterward chooses another food source close by the one presently in her remembrance depending on observed information. A new food source is randomly generated by a scout bee to replace the one abandoned by the onlooker bees.

1) Initialization of Swarm

All the vectors of the population of food sources are initialized by scout bees and control parameters are set. The ABC algorithm has three parameters: the number of food sources (population), the number of test after which a food source is treated to be discarded (limit) and the termination criteria (maximum number of cycle). In the original ABC, the number of food sources is equal to the employed bees or onlooker bees. Initially it consider an evenly dealt swarm of food sources (SN), where each food source x_i (i = 1, 2 ...SN) is a D-dimensional vector. Each food source is generated using following method [8]:

$$x_{ij} = x_{minj} + rand[0,1](x_{maxj} - x_{minj}) \quad (1)$$

Here rand[0,1] is a function that generates an evenly distributed random number in range [0,1].

2) Employed Bee

Employed bees explore new food sources having additional nectar contained by the proximity of the food source in their remembrance. They find a neighbour food source and then estimate its fertility (fitness). Employed bees phase keep informed the present solution based on the information and individual experiences and the fitness value of the recently found solution. New food source with elevated fitness value replace the existing one. The position update equation for j^{th} dimension of i^{th} candidate during this phase is as follow [8]:

$$V_{ij} = x_{ij} + \varphi_{ij}(x_{ij} - x_{kj}) \quad (2)$$

Here $\varphi_{ij}(x_{ij} - x_{kj})$ is known as step size, $k \in \{1, 2, ..., SN\}$, $j \in \{1, 2, ..., D\}$ are two randomly chosen indices. $k \neq i$ ensure that step size has some indicative perfection.

3) Onlooker Bee

Unemployed bees are composed of a couple of groups of bees: onlooker bees and scout bees. Employed bees share information on the subject of food source with onlooker bees waiting in the beehive and subsequently onlooker bees probabilistically pick their food sources depending on this information. In ABC, an onlooker bee pick a food source depending on the probability values intended using the fitness values provided by employed bees. For this purpose, a fitness based selection technique can be used. The number of food sources for onlooker bee is same as the employed. During this phase all employed bee share fitness information of new food sources with onlooker bees. Onlooker bees calculate the selection probability of each food source generated by the employed bee. The best fittest food source is selected by the onlooker. There are number of method for calculation of probability, but it must include fitness. Probability of each food source is decided using its fitness as follow [8]:

$$P_i = \frac{fit_i}{\sum_{i=1}^{SN} fit_i} \tag{3}$$

4) Scout Bee Phase

The unemployed bees that select their food sources capriciously are called scouts. Employed bees whose solutions cannot be enhanced after a predefined number of trials specified by the user of the ABC algorithm and called "limit" or "abandonment criteria" herein, become scouts and their solutions are discarded. Then, the converted scouts start to search for new solutions, randomly. As the location of a food source is not updated for a predefined number of cycles, then the food source is assumed to be rejected and scout bees phase is initialized. During this phase the bee associated with the neglected food source converted into scout bee and the food source is swapped by the capriciously chosen food source inside the search space. In ABC, the predefined number of cycles is an important control parameter which is called limit for elimination. Now the scout bees replace the abandoned food source with new one using following equation [8].

$$x_{ij} = x_{minj} + rand[0,1](x_{maxj} - x_{minj}) \forall j = 1,2,.D \tag{4}$$

Based on the above description, it is clear that in ABC search procedure there are three main control parameters: the number of food sources SN (same as number of onlooker or employed bees), the limit and the maximum number of cycles.

III. Improvement And Modifications In ABC Algorithm

Over and over again real world provides a number of complex optimization problems that cannot be just dealt with easily reached mathematical optimization methods. Cleverness emerged from societal behaviour of social colony members generally used to solve complex problems when a user is not very well-known about the particular solution of the problem. Honey bees are in the category of social insects. The foraging behaviour of honey bees exhibits an intelligent social behaviour, called as swarm intelligence [16]. This swarm intelligence is simulated and an intelligent search algorithm namely, Artificial Bee Colony (ABC) algorithm is recognized by Karaboga in 2005 [6]. Since its inception, a lot of investigation has been carried out to make ABC more and more proficient and to apply ABC for different types of problems.

In order to get rid of the downsides of original ABC, researchers and scientists have modified ABC in many ways. The potentials where ABC can be improved are fine tuning of ABC control parameters SN, ϕ_{ij} and limit (maximum cycle number). Hybridization of ABC with other population based probabilistic or deterministic algorithms. New control parameters also introduced in different phases of ABC. D. Karaboga [6] has suggested that the value of ϕ_{ij} should be in the range of [-1, 1]. The value of limit (utmost cycle number) should be SN × D, where, SN is the number of solutions and D is the dimension of the problem. W Gao et al. [17] anticipated an enhanced solution search equation in ABC, which is based on the fact that bee searches only around the best solution of the previous iteration to increase the exploitation and introduce a selective probability. A. Banharnsakun et al. [18] proposed a novel variant of ABC that is to say the best-so-far selection in artificial bee colony algorithm. In this algorithm the best possible solutions established so far are shared globally in the midst of the entire population. Thus, the new contender solutions are more plausible to be close to the in progress best solution. In other words, we bias the solution direction toward the best-so-far position. Moreover, every succession adjusts the radius of the search for new individual using a larger radius previously in the search procedure and then reduces the radius as the process comes closer to converging. Finally, it uses a more robust calculation to determine and put side by side the quality of alternative solutions. To enhance the exploitation and exploration processes, they propose to make three major changes by introducing the best-so-far method, an adjustable search radius, and an objective-value-based comparison method in DE. J.C. Bansal et al. [56] wished-for balanced ABC; they introduced a novel control parameter, Cognitive Learning Factor and also modified range of ϕ_{ij} in Artificial Bee Colony algorithm.

F Qingxian and D Haijun proposed a modification in the initialization scheme by making the initial group symmetrical, and the Boltzmann selection mechanism was affianced instead of roulette wheel selection for improving the convergence capability of the ABC algorithm [19]. In order to take full advantage of the exploitation capacity of the onlooker stage, Tsai et al. [21] introduced the Newtonian law of universal gravitation in the onlooker phase of the basic ABC algorithm in which roulette wheel based assortment mechanism used for onlooker bees. A Baykasoglu et al. included the ABC algorithm with shift neighbourhood searches and voracious randomized adaptive search heuristic and applied it to the generalized assignment problem [20]. In addition, modified version of the Artificial Bee Colony algorithm is introduced and applied for efficiently solving real-parameter optimization problems by B Akay and D Karaboga [22]. To adjust ABC behaviour for inhibited search space Mezura et al. [23]

anticipated four modifications related with the selection strategy, the scout bee operator, and the parity and border line constraints. Instead of fitness comparative selection, tournament selection is performed to exploit employed bee food sources by onlooker bees. Subsequent, it employed dynamic forbearance for equality constraints. In 2010, G Zhu and S Kwong [8] planned an improved ABC algorithm named gbest-guided ABC (GABC) algorithm by integrating the information about global best (gbest) solution into the solution search equation to improve the exploitation. GABC is inspired by PSO [16], which, in order to improve the exploitation capability, takes advantage of the information of the global best (gbest) solution to guide the search by contender solutions. J.C. Bansal et al. [12] introduced memetic search in ABC algorithm in order to balance exploitation and exploration. In 2010, T Dereli and GS Das [24] anticipated a hybrid bee(s) algorithm for solving container loading tribulations. In this algorithm, a bee(s) algorithm is hybridized by means of the heuristic filling procedure for the solution of container loading problems. In 2010, Huang and Lin [25] anticipated a novel bee colony optimization algorithm with idle-time-based filtering scheme and its application for open shop-scheduling problems. They categorized the foraging behaviours of bees in two terms frontward pass and rearward pass. Forward pass articulates the process of a forager bee leaving the bee hive and flying towards a food source while Backward Pass denotes the process of a forager bee returning to the bee hive and sharing the food source information with other forager bees (role change). N Suguna et al. [26] projected an autonomous rough set approach hybrid with ABC algorithm for reduction of dimensionality. In the anticipated work, effects of the perturbation rate, the scaling factor (step size), and the limit are considered on real-parameter optimization. ABC algorithm hybridized with genetic algorithm in order to balance exploration and exploitation of search space [11], [27]. In 2012, B Wu et al. [28] anticipated enhancement of Global swarm optimization (GSO) hybrid of PSO and ABC. It makes use neighbourhood solution generation method of ABC and recognize new solution only when it is better than preceding one in order to improve GSO performance. A detailed discussion on the performance, applications, hybrids and modification/improvement in ABC algorithm are presented in literature [29]-[30].

IV. Memetic Algorithms

Memetic algorithms (MA) characterize the most up to date budding areas of research in evolutionary computing. The word MA is now extensively used as a synergy for population-based strategies with separate individual learning or local development procedures for problem search. Fairly, MA is sometimes referred in the literature as cultural algorithms, Baldwinian evolutionary algorithms (EA), genetic local search or Lamarckian EAs. The term Memetic Algorithms was given by PA Moscato [31] to a group of pupils of stochastic global search strategies that, commonly speaking, come together within the carcass of Evolutionary Algorithms (EAs) for the profit of problem-oriented local search multi-agent systems. In ethnic development processes, information is processed and overgenerous by the communicating parts; it is not only transmitted impassive between entities. This enhancement is recognized in MAs by taking on board heuristics, approximation algorithms, metaheuristics, local search techniques, particular recombination operators and truncated exact search methods. In aspect, more or less all MAs can be illustrated as a search technique in which a population of optimizing operator work together and compete. MAs have been successfully imposed to a large scale domains that encircle problems in combinatorial optimization, like E. Burke, J. Newall, R. Weare [32] solved university exam timetabling problem using memetic algorithm. R. Cheng, M. Gen [34] applied memetic algorithms while scheduling parallel machine. R. Carr et al. [33] developed a memetic evolutionary algorithm for alignment of protein structures. C. Fleurent et al. [35] proposed a novel hybrid of Genetic algorithm in collaboration of graph colouring algorithm. It is also applied to find solution of travelling salesman problem [36], reverse routing in telecommunication [24], bin packing [37], dynamic optimization [41], VLSI floor planning [38] continuous optimization [39]-[40] and multi-objective optimization [42].

Exploitation potential of evolutionary computing improved at large scale in association of memetic algorithm. Area of applications for memetic algorithms is eternally expending after its instigation. Y. Wang et al. [43] gives a memetic algorithm for the maximum multiplicity problem inspired by Tabu search. X. Xue et al. [44] Optimize ontology coalition by means of Memetic Algorithm based on partial reference alignment. O. Chertov et al. [45] projected a memetic algorithm for solution for providing group anonymity. JC Bansal et al [12] incorporated memetic search in artificial bee colony algorithm inspired by golden section search (GSS) [47]. F. Kang et al [9] projected a new memetic algorithm HJABC inspired by hooke-jeeves method [10]. I. Fister et al [46] estimated a memetic ABC algorithm for large-scale global optimization. S Kumar et al. [13] proposed a randomized memetic ABC with modified GSS process.

Memetic algorithm given by Kang et al. [9] includes Hooke Jeeves [10] local search course of action in Artificial Bee Colony algorithm. HJABC is a hybrid algorithm of escalation search based on the hooke-jeeves pattern search strategy and the original ABC algorithm. HJABC modify the fitness (fit_i) calculation function and integrate the Hooke-Jeeves local search in original ABC. HJABC contains amalgamation of exploratory move and pattern move in order to search optimum result of problem. The initial, step exploratory move think about one variable at a time in order to choose appropriate direction of search process. The subsequent step is pattern search to speed up search in decisive direction by exploratory move. These two steps repeated until the denunciation criteria meet. The Hooke-Jeeves pattern move is a contentious attempt of the algorithms for the exploitation of promising search directions as it accumulate information from previous flourishing search iteration.

TABLE I TEST PROBLEMS

Test Problem	Objective Function	Search Range	Optimum Value	D	Acceptable Error
Rastrigin	$f_1(x)=\sum_{i=1}^{D}\left[x_i^2-10\cos(2\pi x_i)+10\right]$	[-5.12, 5.12]	*f(0) = 0*	30	1.0*E*-05
Zakharov	$f_2(x)=\sum_{i=1}^{D}x_i^2+(\sum_{i=1}^{D}\frac{ix_i}{2})^2+(\sum_{i=1}^{D}\frac{ix_1}{2})^4$	[-5.12, 5.12]	*f*(0) = 0	30	1.0*E*-02
Salomon Problem	$f_3(x)=1-\cos(2\pi\sqrt{\sum_{i=1}^{D}x_i^2})+0.1(\sqrt{\sum_{i=1}^{D}x_i^2})$	[-100, 100]	*f*(0) = 0	30	1.0*E*-01
Colville function	$f_4(x)=100(x_2-x_1^2)^2+(1-x_1)^2+90(x_4-x_3^2)^2+(1-x_3)^2$ $+10.1[(x_2-1)^2+(x_4-1)^2]+19.8(x_2-1)(x_4-1)$	[-10, 10]	*f*(1) = 0	4	1.0*E*-05
Braninss Function	$f_5(x)=a(x_2-bx_1^2+cx_1-d)^2+e(1-f)\cos x_1+e$	$x_1\in[-5,10]$, $x_2\in[0,15]$	*f(-π, 12.275) = 0.3979*	2	1.0*E*-05
Kowalik function	$f_6(x)=\sum_{i=1}^{11}(a_i-\frac{x_1(b_i^2+b_ix_2)}{b_i^2+b_ix_3+x_4})^2$	[-5, 5]	*f*(0.1928, 0.1908, 0.1231, 0.1357) = 3.07*E*-04	4	1.0*E*-05
Shifted Rosenbrock	$f_7(x)=\sum_{i=1}^{D-1}(100(z_i^2-z_{i+1})^2+(z_i-1)^2+f_{bias}$, $z=x-o+1, x=[x_1,x_2,...x_D], o=[o_1,o_2,.......o_D]$	[-100, 100]	*f(o)=f_{bias}=390*	10	1.0*E*-01
Six-hump camel back	$f_8(x)=(4-2.1x_1^2+\frac{1}{3}x_1^4)x_1^2+x_1x_2+(-4+4x_2^2)x_2^2$	[-5, 5]	f(-0.0898, 0.7126) = -1.0316	2	1.0E-05
Easom's function	$f_9(x)=-\cos x_1\cos x_2 e^{(-(x_1-\pi)^2-(x_2-\pi)^2)}$	[-10, 10]	*f(π, π) = -1*	2	1.0*E*-13
Hosaki Problem	$f_{10}(x)=(1-8x_1+7x_1^2-\frac{7}{3}x_1^3+\frac{1}{4}x_1^4)x_2^2\exp(-x_2)$	$x_1\in[0,5]$, $x_2\in[0,6]$	*-2.3458*	2	1.0*E*-06
McCormick	$f_{11}(x)=\sin(x_1+x_2)+(x_1-x_2)^2-\frac{3}{2}x_1+\frac{5}{2}x_2+1$	$-1.5\le x_1\le 4, -3\le x_2\le 3,$	*f(-0.547, -1.547) =-1.9133*	30	1.0*E*-04
Meyer and Roth Problem	$f_{12}(x)=\sum_{i=1}^{5}(\frac{x_1x_3t_i}{1+x_1t_i+x_2v_i}-y_i)^2$	[-10, 10]	*f(3.13, 15.16,0.78) = 0.4E-04*	3	1.0*E*-03
Shubert	$f_{13}(x)=-\sum_{t=1}^{5}i\cos((i+1)x_1+1)\sum_{i=1}^{5}i\cos((i+1)x_2+1)$	[-10, 10]	*f(7.0835, 4.8580)= -186.7309*	2	1.0*E*-05

The memetic search in ABC (MeABC) expected by JC Bansal et al. [12] enthused by Golden Section Search (GSS) process [47]. In MeABC only the outstanding particle of the current swarm keep informed itself in its propinquity. MeABC also modify position update equation as per the following equation in order to control step size.

$$x'_{ij}=x_{ij}+\phi_{ij}(x_{ij}-x_{kj})+\psi_{ij}(x_{bestj}-x_{ij}) \tag{5}$$

Here Ψij is an arbitrary number in interval [0, C], for some positive constant C.

V. IMPROVED EMPLOYED BEE PHASE AND ONLOOKER BEE PHASE IN ARTIFICIAL BEE COLONY ALGORITHM

Exploration and exploitation are the two imperative uniqueness of the population-based optimization algorithm such as GA (48), PSO (2), DE (49), BFO (5), firefly algorithm [54] and spider monkey optimization algorithm [55]. In these optimization algorithms, the exploration refers to the ability to investigate the various unknown regions in the solution space to discover the global optimum. The exploitation is the ability to use the knowledge of the previous good solutions to find better solutions and exploration is the process that spread the search space. The Exploration and exploitation are the two opposite sides of problem solving by search in order to accomplish better optimization performance these two abilities must remain in balance. D Karaboga and B Akay (7) experienced diverse modifications of ABC for global optimization and bring into being that the ABC shows poor performance and remains inefficient during the exploration of the search space. In ABC, any probable solution updates itself using the information provided by a randomly selected probable solution within the in progress swarm. In this development, a step size which is simply a combination of a untailored number $\phi_{ij}\in[-1, 1]$, current solution and a chaotically selected solution are used. Now the quality of the modified solution generally depends upon this step size. If the step size is too large then simplified solution can surpass the true solution. Large step size may takes place if the difference of current solution and randomly selected solution is large with high unqualified value of ϕ_{ij}, and if this step size is too diminutive then the convergence rate of ABC may considerably reduce as it takes more time to move towards optimum value. An appropriate sense of balance in this step size can balance the exploration and exploitation capability of the ABC at the same time. But, since this step size consists of random element so the balance cannot be done by hand. The exploitation capability can be improved by incorporation of a local search algorithm with the ABC algorithm. For that reason, this paper introduce, an improved employed bee phase and onlooker bee phase inspired by modified GSS process to balance the diversity and convergence speed of ABC. MeABC [12] use memetic search phase after scout bee phase as an additional local search phase while algorithm proposed in this paper use modified memetic search in both employed bee phase and onlooker bee phase along with modified golden section search process. It also modifies the range of two parameters in GSS process and applies GSS based search process in onlooker bee phase. For that reason, in these modifications, better solutions get more chance in search process and minimize the threat of less stability here. In modified GSS process it randomly decides step size as in Randomized ABC [13]. This paper also modifies the fitness calculation and probability selection mechanism inspired by Modified ABC [13]. The improved search strategy in ABC is outlined as follow:

Step 1: Initialize the population of N evenly disseminated individuals. Each individual x_{ij} is a food source (i.e. required solution) and has D number of attributes. D is identified as the dimension of the problem. i^{th} solution in j^{th} dimension denoted as x_{ij}. Where $j \in \{1, 2,\ldots\ldots, D\}$

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

Step 2: Calculate approximately the fitness of each and every individual solution using the following method,

if (solution_value >= 0)

then

$$fit_i = \phi \times (\frac{1}{2 \times sol\_val + 1}) + (1-\phi) \times (1 + fabs(\frac{1}{sol\_val}))$$

else

$$fit_i = (1-\phi) \times (\frac{1}{2 \times sol\_val + 1}) + \phi \times (1 + fabs(\frac{1}{sol\_val}))$$

Here $\phi \in [0, 1]$.

Step 3: Each employed bee, placed at a food source that is different from others, search in the proximity of its current position to find a better food source. Apply improved search phase inspired by GSS process. Take $a=-1.2$, $b=1.2$ and ϕ = rand[0.55,0.65]. Compute $f_1=b-((b-a)*\phi)$ and $f_2=a+((b-a)*\phi)$.

Repeat while termination criteria meet

Calculate value of function based on f_1 and f_2.

If $f_{1.val} < f_{2.val}$ then

b = f_2 and the solution lies in the range [a, b]

else

a = f_1 and the solution lies in the range [a, b]

Modify the position of solution using following equation

$$x'_{ij} = x_{ij} + \left(x_{ij} - x_{kj}\right) \times f_l$$

Here k=rand[0,1]*Food Number, j=rand[0,1]*dimension and l={1,2}.

Here, $k \in \{1, 2,\ldots., N\}$ and $j \in \{1, 2, \ldots, D\}$ are randomly chosen indices. N is number of employed bees. ϕ_{ij} is a uniform arbitrary number from [-1, 1].

Step 4: Compute the fitness of both x_{ij} and v_{ij}. Apply greedy selection strategy to select better one of them.

Step 5: Calculate and normalize the probability values, P_{ij} for each solution x_i using the following formula.

$$p_{ij} = \phi \times (\frac{fit_i}{\max\_fitness}) + (1-\phi) \times (\frac{fit_i}{\sum_{i=1}^{N} fit_i}))$$

Here ϕ is a random number in range [0,1]

Step 6: Assign each onlooker bee to a solution, x_i at random with probability proportional to P_{ij}. Apply improved search phase inspired by GSS process. Take $a=-1.2$, $b=1.2$ and ϕ = rand[0.55,0.65]. Compute $f_1=b-((b-a)*\phi)$ and $f_2=a+((b-a)*\phi)$.

Repeat while termination criteria meet

Calculate value of function based on f_1 and f_2.

If $f_{1.val} < f_{2.val}$ then

b = f_2 and the solution lies in the range [a, b]

else

a = f_1 and the solution lies in the range [a, b]

Modify the position of solution using following equation

$$x'_{ij} = x_{ij} + \left(x_{ij} - x_{kj}\right) \times f_l$$

Here k=rand[0,1]*Food Number, j=rand[0,1]*dimension and l={1,2}.

Step 7: Arrange new food sources, x'_{ij} for each onlooker bee.

Step 8: Compute the fitness of each onlooker bee, x_{ij} and the new solution, x'_{ij}. Select the fittest one using greedy selection development.

Step 9: If a particular solution x_{ij} has not been improved over a predefined number of cycles, then select it for denunciation. Replace the solution by placing a scout bee at a food source generated evenly at random within the search space using

$$x_{ij} = x_{\min j} + rand[0,1](x_{\max j} - x_{\min j})$$

for j = 1, 2,……,D

Step 10: Keep track of the best food sources (solution) found so far.

Step 11: Check termination criteria. If the most excellent solution found is satisfactory or reached the maximum iterations, stop and return the best solution found so far. If not go back to second step and repeat again.

TABLE II COMPARISON OF THE RESULTS OF TEST PROBLEMS

Test Problem	Algorithm\Measure	MFV	SD	ME	AFE	SR
f_1	IMeABC	8.75E-06	1.23E-06	8.75E-06	46206.21	100
	ABC	4.62E-01	6.60E-01	4.62E-01	68509.2	60.8
	IoABC	8.79E-06	1.22E-06	8.79E-06	65992.08	100
	MeABC	8.25E-06	1.67E-06	8.25E-06	50753.71	100
	EnABC	7.04E-06	3.28E-06	7.04E-06	129716.9	97.6
	RMABC	7.62E-06	2.78E-06	9.62E-05	200000	0
f_2	IMeABC	9.59E-03	6.43E-04	9.59E-03	73935.62	96.8
	ABC	1.14E+02	1.62E+01	1.14E+02	100048	0
	IoABC	1.22E-02	4.20E-03	1.22E-02	94586.45	48.8
	MeABC	3.23E-02	1.49E-02	3.23E-02	99941.89	0
	EnABC	9.66E-03	4.77E-04	9.66E-03	136300.7	100
	RMABC	9.62E+01	1.78E+01	9.62E+01	200000	0
f_3	IMeABC	9.24E-01	3.45E-02	9.24E-01	16933.15	100
	ABC	1.17E+00	1.46E-01	1.17E+00	98080.93	10.4
	IoABC	9.21E-01	3.14E-02	9.21E-01	20201.28	100
	MeABC	9.19E-01	3.59E-02	9.19E-01	25611.34	100
	EnABC	9.26E-01	3.62E-02	9.26E-01	32312.2	100
	RMABC	9.36E-01	3.32E-02	9.36E-01	87168.16	97
f_4	IMeABC	8.96E-03	1.46E-03	8.96E-03	29124.99	100
	ABC	4.03E-01	3.02E-01	4.03E-01	100016.5	0
	IoABC	8.46E-03	3.64E-03	8.46E-03	41042.38	98.4
	MeABC	8.55E-03	5.69E-03	8.55E-03	41571.1	90.4
	EnABC	1.20E-02	1.38E-02	1.20E-02	93620.7	72.8
	RMABC	1.80E-02	1.62E-02	1.80E-02	159293.8	41
f_5	IMeABC	3.98E-01	6.30E-06	5.41E-06	7993.696	93.6
	ABC	3.98E-01	7.03E-06	6.27E-06	12587.07	89.6
	IoABC	3.98E-01	6.81E-06	6.26E-06	18746.91	82.4
	MeABC	3.98E-01	6.82E-06	6.01E-06	15757.18	85.6
	EnABC	3.98E-01	6.69E-06	5.83E-06	27678.54	87.2
	RMABC	3.98E-01	6.33E-06	5.49E-06	19134.85	91
f_6	IMeABC	4.90E-04	2.05E-04	1.82E-04	61974.88	69.6
	ABC	5.03E-04	8.55E-05	1.96E-04	92354.85	18.4
	IoABC	4.19E-04	6.66E-05	1.11E-04	64745.6	82.4
	MeABC	4.10E-04	4.69E-05	1.02E-04	63350.62	84.8
	EnABC	3.91E-04	2.34E-05	8.31E-05	84898.18	95.2
	RMABC	3.96E-04	3.45E-05	8.86E-05	91857.83	89
f_7	IMeABC	3.91E+02	2.86E+00	1.19E+00	85979.52	31.2
	ABC	3.95E+02	5.24E+00	4.89E+00	99342.04	1.6
	IoABC	3.91E+02	2.51E+00	1.26E+00	83198.75	38.4
	MeABC	3.91E+02	2.05E+00	9.60E-01	76399.38	46.4
	EnABC	3.92E+02	2.17E+00	1.70E+00	183136.3	15.2
	RMABC	3.90E+02	3.82E-02	9.02E-02	101917.5	91
f_8	IMeABC	3.00E+00	4.95E-15	5.24E-15	12157.25	100
	ABC	3.00E+00	8.46E-07	8.73E-08	31468.02	91.2
	IoABC	3.00E+00	4.24E-15	4.78E-15	19112.064	100
	MeABC	3.00E+00	4.93E-15	4.86E-15	16552.12	100
	EnABC	3.00E+00	7.20E-10	8.95E-11	53419.26	91.2
	RMABC	-1.03E+00	7.52E-04	4.03E+00	200027.9	0
f_9	IMeABC	-1.03E+00	1.43E-05	1.83E-05	58728.38	42.4
	ABC	-1.03E+00	3.87E-03	4.65E-03	100047.1	0
	IoABC	-1.03E+00	1.34E-05	1.78E-05	60010.48	42.4
	MeABC	-1.03E+00	1.31E-05	1.58E-05	60744.14	45.6
	EnABC	-1.03E+00	1.40E-05	1.57E-05	112737.6	49.6
	RMABC	-9.85E-01	1.37E-02	4.68E-02	200026	0
f_{10}	IMeABC	-2.48E+04	7.62E-03	4.93E-01	49037.47	82.4
	ABC	-2.47E+04	8.88E+01	9.65E+01	100034.1	0
	IoABC	-2.48E+04	3.19E-02	5.15E-01	74105.98	44.8
	MeABC	-2.48E+04	2.03E-01	6.60E-01	92406.53	16.8
	EnABC	-2.48E+04	2.01E-01	6.34E-01	185437.7	16.8
	RMABC	-5.99E+11	1.43E+11	5.99E+11	200025	0

f_{11}	IMeABC	-2.35E+00	6.35E-06	5.76E-06	17373.28	91.2
	ABC	-2.30E+00	4.64E-02	4.33E-02	100042.7	0
	IoABC	-2.35E+00	8.68E-06	6.45E-06	35245.93	88.8
	MeABC	-2.35E+00	2.23E-05	1.32E-05	57963.38	67.2
	EnABC	-2.35E+00	9.91E-06	7.03E-06	92231.55	80.8
	RMABC	9.35E-01	4.44E-16	3.28E+00	200021.2	0
f_{12}	IMeABC	-1.91E+00	6.62E-06	8.80E-05	7184.832	100
	ABC	-1.89E+00	1.95E-02	2.02E-02	100055.7	0
	IoABC	-1.91E+00	6.98E-06	8.77E-05	17949.83	99.2
	MeABC	-1.91E+00	1.16E-05	9.33E-05	48658.9	80
	EnABC	-1.91E+00	5.67E-05	1.26E-04	149353.4	38.4
	RMABC	1.50E-02	1.73E-18	1.93E+00	200022.7	0
f_{13}	IMeABC	1.91E-03	4.84E-06	1.95E-03	4502.848	99.2
	ABC	1.91E-03	3.55E-06	1.95E-03	25615.13	95.2
	IoABC	1.91E-03	2.77E-06	1.95E-03	4145	100
	MeABC	1.91E-03	2.78E-06	1.95E-03	4380.056	100
	EnABC	1.91E-03	3.00E-06	1.95E-03	13174.1	100
	RMABC	-1.79E+02	7.07E+00	1.79E+02	200023	0
f_{14}	IMeABC	2.63E+00	9.90E-03	5.96E-03	68870.46	55.2
	ABC	2.65E+00	1.21E-02	2.23E-02	98488.2	3.2
	IoABC	2.63E+00	6.86E-03	3.60E-03	72330.82	53.6
	MeABC	2.63E+00	1.04E-02	8.46E-03	86246.32	25.6
	EnABC	2.63E+00	8.94E-03	6.77E-03	161139.6	31.2
	RMABC	2.64E+00	1.06E-02	1.11E-02	185673.8	13
f_{15}	IMeABC	7.88E-16	1.82E-16	7.88E-16	69177.1	100
	ABC	2.51E-10	4.41E-10	2.51E-10	100040	0
	IoABC	9.16E-16	8.21E-17	9.16E-16	74015.57	100
	MeABC	8.80E-16	9.58E-17	8.80E-16	64357.98	100
	EnABC	9.00E-16	1.04E-16	9.00E-16	186071.6	100
	RMABC	-1.61E+38	8.83E+37	1.61E+38	200015.1	0

VI. EXPERIMENTAL RESULTS

A. Test problems under consideration

Artificial Bee Colony algorithm with improvement in onlooker bee phase applied to the twelve benchmark functions for whether it gives better result or not at different probability and also applied for two real world problems. Benchmark functions taken in this paper are of different characteristics like uni-model or multi-model and separable or non-separable and of different dimensions. In order to analyse the performance of IMeABC, it is applied to global optimization problems (f_1 to f_{14}) listed in Table I. Test problems f_1 –f_{14} are taken from [50]-[52].

Compression Spring (f_{14}): The compression spring problem [50] minimizes the weight of a compression spring that is subjected to constraints of shear stress, surge frequency, minimum deflection and limits on outside diameter and on design variables. In case of compression spring three design variables considered: The diameter of wire(x_1), mean coil diameter (x_2) and count of active coils (x_3). Simple mathematical representation of this problem is:

$$x_1 \in \{1,2,3,................,70\} granularity_1$$

$$x_2 \in [0.6;3], x_3 \in [0.207;0.5] granularity_{0.001}$$

And four constraints

$$g_1 := \frac{8c_f F_{\max} x_2}{\pi x_3^3} - S \le 0, g_2 := l_f - l_{\max} \le 0$$

$$g_3 := \sigma_p - \sigma_{pm} \le 0, g_4 := \sigma_w - \frac{F_m ax - F_p}{K} \le 0$$

$$\text{Where}: c_f = 1 + 0.75\frac{x_3}{x_2 - x_3} + 0.615\frac{x_3}{x_2}, F_{\max} = 1000,$$

$$S = 189000, l_f = \frac{F_{\max}}{K} + 1.05(x_1+2)x_3, l_{\max} = 14, \sigma_p = \frac{F_p}{K},$$

$$\sigma_{pm} = 6, F_p = 300, K = 11.5\times10^6 \frac{x_3^4}{8x_1x_2^3}, \sigma_w = 1.25$$

And the function to be minimized is

$$f_{14}(X) = \pi^2 \frac{x_2 x_3^2 (x_1+2)}{4}$$

The best ever identified solution is (7, 1.386599591, 0.292), which gives the fitness value f =2.6254 and 1.0E-04 is tolerable error for compression spring problem.

Pressure Vessel Design (f_{15}): The problem of minimizing total cost of the material, forming and welding of a cylindrical vessel [50]. In case of pressure vessel design generally four design variables are considered: shell thickness (x_1), spherical head thickness (x_2), radius of cylindrical shell (x_3) and shell length (x_4). Simple mathematical representation of this problem is as follow:

$$f_{15}(x) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1611x_1^2x_4 + 19.84x_1^2x_3$$

$$\textit{subject to} \quad g_1(x) := 0.0193x_3 - x_1, g_2(x) := 0.00954x_3 - x_2,$$

$$g_3 := 750\times1728 - \pi x_3^2 (x_4 + \frac{4}{3} x_3)$$

The search boundaries for the variables are

$1.125 \le x_1 \le 12.5, 0.625 \le x_2 \le 12.5,$

$1.0\times10^{-8} \le x_3 \le 240 \quad and \quad 1.0\times10^{-8} \le x_3 \le 240$

The best ever identified global optimum solution is f(1.125, 0.625, 55.8592, 57.7315) = 7197.729 [41]. The tolerable error for considered problem is 1.0E-05.

B. Experimental Setup

To prove the efficiency of IMeABC, it is compared with original ABC algorithm [6], Randomized Memetic ABC (RMABC) algorithm [13], Memetic search in ABC (MeABC) algorithm [12], Enhanced Local Search in Artificial Bee Colony (EnABC) Algorithm [53], improved onlooker bee phase in ABC (IoABC) [15] over well thought-out fourteen problems, following experimental setting is adopted:

- The size of colony= Population size SN =80
- Number of Employed bee = Number of Onlooker bee =SN/2
- The maximum number of cycles for foraging MCN =200000
- Number of repetition of experiment =Runtime =125
- Limit =1500, A food source which could not be improved through "limit" trial is abandoned by its employed bee
- The mean function values (MFV), standard deviation (SD), mean error (ME), average function evaluation (AFE) and success rate (SR) of considered problem have been recorded.
- Experimental settings for ABC, RMABC, MeABC and EnABC, IoABC are same as IMeABC.

C. Result Comparison

Mathematical results of IMeABC with experimental setting as per subsection 5.B are discussed in Table II. Table II show the connection of results based on mean function value (MFV), standard deviation (SD), mean error (ME), average function evaluations (AFE) and success rate (SR). Table II shows that a good number of the times IoABC outperforms in terms of efficiency (with less number of function evaluations) and reliability as compare to other considered algorithms. The proposed algorithm all the time improves AFE and most of the time it also improve SD and ME. It is due to randomness introduced during fitness calculation and probability calculation. Table III contains summary of table II outcomes. In Table III, '+' indicates that the IoABC is better than the considered algorithms and '-' indicates that the algorithm is not better or the difference is very small. The last row of Table III, establishes the superiority of IMeABC over RMABC, EnABC, MeABC, IoABC and ABC.

TABLE III SUMMARY OF TABLE II OUTCOME

Test Problem	IMeABC VS. ABC	IMeABC VS. IoABC	IMeABC VS. MeABC	IMeABC VS. EnABC	IMeABC VS. RMABC
f_1	+	+	+	+	+
f_2	+	+	+	-	+
F_3	+	+	+	+	+
f_4	+	+	+	+	+
f_5	+	+	+	+	+
f_6	+	-	-	-	-
f_7	+	-	-	+	-
f_8	+	+	+	+	+
f_9	+	=	-	-	+
f_{10}	+	+	+	+	+
f_{11}	+	+	+	+	+
f_{12}	+	+	+	+	+
f_{13}	+	-	-	-	+
f_{14}	+	+	+	+	+
f_{15}	+	+	-	+	+
Total number of + sign	14	12	10	11	13

VII. CONCLUSION

This paper, modify the two phases of ABC algorithm (employed bee phase and onlooker bee phase) by introducing modified GSS process and new strategy for probability and fitness calculation. Newly introduced strategy added in employed bee phase and onlooker bee phase. Proposed algorithm modifies search range of GSS process and solution update equation in order to balance intensification and diversification of local search space. Further, the modified strategy is applied to solve 13 well-known standard benchmark functions and two real world problems (Compression spring problem and pressure vessel design problem). With the help of experiments over test problems and real world problems, it is observed that the insertion of the proposed strategy in the original ABC algorithm get better the trustworthiness, efficiency and accuracy as compare to their original version. Table II and III show that the proposed IMeABC is able to solve almost all the considered problems with fewer efforts. Statistical results show that the enhanced algorithm is superior to original ABC algorithm and its recent variants. Proposed algorithm has the ability to get out of a local minimum and has higher rate of convergence. It can be resourcefully applied for separable, multivariable, multimodal function optimization. The proposed strategy also improves results for both real world problems compression spring problem and pressure vessel design problem.

REFERENCES

[1] M Dorigo, G Di Caro (1999) Ant colony optimization: a new metaheuristic. In: Evolutionary computation, 1999. CEC 99. Proceedings of the 1999 congress on, 2. IEEE

[2] J Kennedy, R Eberhart (1995) Particle swarm optimization. In: Neural networks, 1995. Proceedings., IEEE international conference on, 4, pp 1942–1948. IEEE

[3] KV Price, RM Storn, JA Lampinen (2005) Differential evolution: a practical approach to global optimization. Springer, Berlin

[4] J Vesterstrom, R Thomsen (2004) A comparative study of differential evolution, particle swarm optimization, and evolutionary algorithms on numerical benchmark problems. In: Evolutionary computation, 2004. CEC2004. Congress on, 2, pp 1980–1987. IEEE

[5] KM Passino (2002) Biomimicry of bacterial foraging for distributed optimization and control. IEEE Control SystMag 22(3):52–67

[6] D Karaboga (2005) An idea based on honey bee swarm for numerical optimization. Techn. Rep. TR06, Erciyes University Press, Erciyes

[7] D Karaboga, B Akay B (2009) A comparative study of artificial bee colony algorithm. Appl Math Comput 214(1):108–132

[8] G Zhu, S Kwong (2010) Gbest-guided artificial bee colony algorithm for numerical function optimization. Appl Math Comput 217(7):3166–3173

[9] F Kang, J Li, Z Ma, H Li (2011) Artificial bee colony algorithm with local search for numerical optimization. J Softw 6(3):490–497

[10] R Hooke, TA Jeeves (1961) “Direct search” solution of numerical and statistical problems. J ACM (JACM) 8(2):212–229

[11] S Kumar, VK Sharma and R Kumari (2013) A Novel Hybrid Crossover based Artificial Bee Colony Algorithm for Optimization Problem, International Journal of Computer Application 82(8):18-25

[12] JC Bansal, H Sharma, KV Arya and A Nagar, “Memetic search in artificial bee colony algorithm.” Soft Computing (2013): 1-18.

[13] S Kumar, VK Sharma and R Kumari (2014) Randomized Memetic Artificial Bee Colony Algorithm. International Journal of Emerging Trends & Technology in Computer Science (IJETTCS) 3(1): 52-62.

[14] S Kumar and VK Sharma (2014) Modified Artificial Bee Colony Algorithm. International Journal of Information, Communication and Computing Technology. In Print.

[15] S Kumar, VK. Sharma and R Kumari (2014) Improved Onlooker Bee phase in artificial bee colony algorithm, International Journal of Computer Application 90(6): 3-39.

[16] V Agrawal, H Sharma, and JC Bansal. "Bacterial Foraging Optimization: A Survey." Proceedings of the International Conference on Soft Computing for Problem Solving (SocProS 2011) December 20-22, 2011. Springer India, 2012.

[17] W Gao and S. Liu "A modified artificial bee colony algorithm." Computers & Operations Research 39.3 (2012): 687-697.

[18] A Banharnsakun, T Achalakul, and B Sirinaovakul. "The best-so-far selection in artificial bee colony algorithm." Applied Soft Computing 11.2 (2011): 2888-2901.

[19] D. Haijun, F. Qingxian. Bee colony algorithm for the function optimization. Science Paper Online, August 2008.

[20] A Baykasoglu, L Ozbakir, and P Tapkan. Artificial bee colony algorithm and its application to generalized assignment problem. Swarm Intelligence: Focus on Ant and Particle Swarm Optimization, pages 113–144, 2007.

[21] PW Tsai, JS Pan, BY Liao, and SC Chu. Enhanced artificial bee colony optimization. International Journal of Innovative Computing, Information and Control, 5(12), 2009.

[22] B Akay and D Karaboga. A modified artificial bee colony algorithm for real-parameter optimization. Information Sciences, 2010

[23] E. Mezura-Montes and O. Cetina-Dom´ınguez. Empirical analysis of a modified artificial bee colony for constrained numerical optimization. Applied Mathematics and Computation, 2012.

[24] T Dereli and GS Das. A hybrid’bee (s) algorithm’for solving container loading problems. Applied Soft Computing, 2010.

[25] YM Huang and JC Lin. A new bee colony optimization algorithm with idle-time-based filtering scheme for open shop-scheduling problems. Expert Systems with Applications, 2010.

[26] N Suguna and KG Thanushkodi. An independent rough set approach hybrid with artificial bee colony algorithm for dimensionality reduction. American Journal of Applied Sciences, 8, 2011.

[27] S Pandey and S Kumar "Enhanced Artificial Bee Colony Algorithm and It’s Application to Travelling Salesman Problem." HCTL Open International Journal of Technology Innovations and Research, Volume 2, 2013:137-146

[28] B Wu, C Qian, W Ni, and S Fan. The improvement of glowworm swarm optimization for continuous optimization problems. Expert Systems with Applications, 2012.

[29] D. Karaboga, B. Gorkemli, C. Ozturk, and N. Karaboga. "A comprehensive survey: artificial bee colony (ABC) algorithm and applications." *Artificial Intelligence Review* (2012): 1-37.

[30] JC Bansal, H. Sharma and SS Jadon "Artificial bee colony algorithm: a survey." International Journal of Advanced Intelligence Paradigms 5.1 (2013): 123-159.

[31] PA Moscato, “On evolution, search, optimization, genetic algorithms and martial arts: Towards memetic algorithms.” Tech. Rep. Caltech Concurrent Computation Program, Report. 826, California Inst. of Tech., Pasadena, California, USA (1989).

[32] E Burke, J Newall, R Weare.: A memetic algorithm for university exam timetabling. In: E. Burke, P. Ross (eds.) The Practice and Theory of Automated Timetabling, Lecture Notes in Computer Science, vol. 1153, pp. 241–250. Springer Verlag (1996).

[33] R. Carr, W. Hart, N. Krasnogor, E. Burke, J. Hirst, J. Smith, ”Alignment of protein structures with a memetic evolutionary algorithm,” In: Proceedings of the Genetic and Evolutionary Computation Conference. Morgan Kaufman (2002).

[34] R. Cheng, M. Gen, “Parallel machine scheduling problems using memetic algorithms.” Computers & Industrial Engineering 33(3–4), 761–764 (1997).

[35] C. Fleurent, J. Ferland, “Genetic and hybrid algorithms for graph coloring.” Annals of Operations Research 63, 437–461 (1997).

[36] G. Gutin, D. Karapetyan, N. Krasnogor, “Memetic algorithm for the generalized asymmetric traveling salesman problem.” In: M. Pavone, G. Nicosia, D. Pelta, N. Krasnogor (eds.) Proceedings of the 2007Workshop On Nature Inspired Cooperative Strategies for Optimisation. Lecture Notes in Computer Science (LNCS), vol. to appear. Springer (2007)

[37] L. He, N. Mort, “Hybrid genetic algorithms for telecomunications network back-up routeing.” BT Technology Journal 18(4) (2000)

[38] C. Reeves, “Hybrid genetic algorithms for bin-packing and related problems.” Annals of Operations Research 63, 371–396 (1996)

[39] M. Tang, X. Yao, “A memetic algorithm for VLSI floorplanning.” Systems, Man, and Cybernetics, Part B, IEEE Transactions on 37(1), 62–69 (2007). DOI 10.1109/TSMCB.2006.883268.

[40] W. E. Hart, “Adaptive Global Optimization with Local Search.” Ph.D. Thesis, University of California, San Diego (1994)

[41] G. M. Morris,D. S. Goodsell,R. S. Halliday, R. Huey, W. E. Hart, R. K. Belew, A. J. Olson,: “Automated docking using a lamarkian genetic algorithm and an empirical binding free energy function.” J Comp Chem 14, 1639–1662 (1998)

[42] D. Liu, K. C. Tan, C. K. Goh, W. K. Ho, “A multi-objective memetic algorithm based on particle swarm optimization.” Systems, Man, and Cybernetics, Part B, IEEE Transactions on 37(1), 42–50 (2007).

[43] Y Wang, JK Hao, F Glover, Z Lü, “A tabu search based memetic algorithm for the maximum diversity problem.” Engineering Applications of Artificial Intelligence 27 (2014): 103-114.

[44] X Xue, Y Wang, and A. Ren. “Optimizing ontology alignment through Memetic Algorithm based on Partial Reference Alignment.” Expert Systems with Applications 41.7 (2014): 3213-3222.

[45] O. Chertov and Dan Tavrov. "Memetic Algorithm for Solving the Task of Providing Group Anonymity." Advance Trends in Soft Computing. Springer International Publishing, 2014. 281-292.

[46] I. Fister, I. Fister Jr, J. Bres, V. Zumer. "Memetic artificial bee colony algorithm for large-scale global optimization." Evolutionary Computation (CEC), 2012 IEEE Congress on. IEEE, 2012.

[47] Kiefer J (1953) Sequential minimax search for a maximum. In: Proceedings of American Mathematical Society, vol. 4, pp 502–506.

[48] D.E. Goldberg. Genetic algorithms in search, optimization, and machine learning. 1989.

[49] R. Storn and K. Price. Differential evolution-a simple and efficient adaptive scheme for global optimization over continuous spaces. Journal of Global Optimization, 11:341–359, 1997.

[50] H Sharma, JC Bansal, and KV Arya. "Opposition based lévy flight artificial bee colony." *Memetic Computing* (2012): 1-15.

[51] M.M. Ali, C Khompatraporn, and Z.B. Zabinsky. "A numerical evaluation of several stochastic algorithms on selected continuous global optimization test problems." Journal of Global Optimization, 31(4):635–672, 2005.
[52] P.N. Suganthan, N. Hansen, J.J. Liang, K. Deb, YP Chen, A. Auger, and S. Tiwari. "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization." In CEC 2005, 2005.
[53] S. Kumar and VK. Sharma, R Kumari (2014) Enhanced Local Search in Artificial Bee Colony Algorithm. International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS), 7(2):177-184.
[54] XS Yang (2009) Firefly algorithms for multimodal optimization. Stochastic algorithms: foundations and applications. Springer Berlin Heidelberg, 2009. 169-178.
[55] JC Bansal, H Sharma, SS Jadon, & M Clerc,(2013). Spider Monkey Optimization algorithm for numerical optimization. Memetic Computing, 1-17.
[56] JC Bansal, H Sharma, A Nagar, & KV Arya, (2013). Balanced artificial bee colony algorithm. International Journal of Artificial Intelligence and Soft Computing, 3(3), 222-243.
[57] S. Kumar and VK. Sharma, R Kumari (2014) Modified position update in spider monkey optimization algorithm. International Journal of Emerging Technologies in Computational and Applied Sciences (IJETCAS), 7(2):198-204.