To appear in the *Journal of Experimental & Theoretical Artificial Intelligence*
Vol. 00, No. 00, Month 20XX, 1–21

Grasshopper inspired artificial bee colony algorithm for numerical optimization

Nirmala Sharma[a*], Harish Sharma[b], Ajay Sharma[c], and Jagdish Chand Bansal[d]

[a,b]*Rajasthan Technical University, Kota, India,*[c]*Government Engineering College, Jhalawar, India,*[d]*South Asian University, New Delhi, India*

(*Received 00 Month 20XX; final version received 00 Month 20XX*)

Swarm intelligence (SI) based algorithms are performing very well in the field of optimization over the past few decades. A lot of new SI based algorithms are being developed. The existing algorithms are also modified, mostly, either by hybridizing them with some other algorithms or by incorporating local search techniques. This research presents a new local search strategy based on grasshopper jumping mechanism. The proposed local search strategy is termed as *grasshopper local search strategy (GHLS)*. Further, the proposed strategy is incorporated into an efficient SI based algorithm, artificial bee colony (ABC) algorithm. The proposed hybridized algorithm is termed as *grasshopper inspired artificial bee colony (GHABC) algorithm*. The proposed GHABC is tested on 37 numerical benchmark optimization functions. The results indicate that the proposed GHABC algorithm is a competent approach for solving numerical optimization problems.

Keywords: Local search; Grasshopper; Nature inspired algorithms; Swarm intelligence; Optimization

1. Introduction

In the field of optimization a lot of conventional and non-conventional algorithms have been applied in years. The conventional algorithms are sometimes time consuming and non-robust (Yang, 2014). A class of non-conventional techniques namely, swarm intelligence (SI) based techniques emerged due to wide availability of high computational efficiency. SI based algorithms are artificial intelligent techniques that are based on the social grouping behaviour of animals, insects, and birds etc., found in nature. Artificial bee colony (ABC) algorithm is among the most popular SI based algorithms. The ABC algorithm was introduced in 2005 by D. Karaboga (Karaboga, 2005). It is an efficient technique based on the nourishment scavenging behavior of honey bees. The ABC consists of a population of potential solutions as other population based optimization algorithms. The potential solutions are food sources of honey bees. The fitness is determined in terms of the quality (nectar amount) of the food source (Karaboga, Gorkemli, Ozturk, & Karaboga, 2014; Bansal, Sharma, & Jadon, 2013).

There are two fundamental processes which drive the swarm to update the position in the search space in ABC: the variation process, which explores the different areas of the search space, and the selection process, which ensures the exploitation of the previously explored areas based upon the previous experience and knowledge. However, it has been

*Corresponding author. Email: nsharma@rtu.ac.in,

proven that the ABC may sometimes stop proceeding towards the global optimum even though the population has not converged to a local optimum (Karaboga & Akay, 2009). It can be seen that the solution search process of ABC algorithm is good at exploration but poor at exploitation (Zhu & Kwong, 2010). Therefore, to maintain the proper harmony between exploration and exploitation behavior of ABC, it is highly required to develop a local search approach in the basic ABC to exploit the search region.

In this article, a local search strategy inspired from grasshopper jumping phenomenon is developed. Here the position update strategy (modified position of solutions) is derived from the jumping distance of the grasshopper. The proposed local search strategy is termed as grasshopper local search strategy (GHLS). Further, the proposed local search is implemented into ABC process in expectation of improving exploitation ability of the algorithm. The proposed hybridized algorithm is named as grasshopper inspired ABC (GHABC). The performance of GHABC is analyzed through various numerical experiments with respect to accuracy, reliability, and consistency. The obtained numerical results prove the validity of the proposed approach.

The rest of the paper is structured as follows: Section 2, covers the overview of ABC algorithm. The proposed GHLS strategy is explained in Section. 3. Section 4 describes the proposed GHABC algorithm. An extensive analysis of the proposed GHABC algorithm is performed with standard benchmark optimization in Section. 5. Finally the Section. 6 summarizes the proposed work.

2. Overview of ABC Algorithm

The cooperative intelligent behavior of social insects, birds and other social animal have always been an inspiration and interesting field for the researchers of various fields. The grouping behavior of insects and animals is known as the swarm behavior. Swarm intelligence (SI) based techniques are emerging techniques with the advent of computational intelligence. Self organization and division of labor are two key components of SI. ABC algorithm is an SI based optimization algorithm. ABC is inspired from the aggregate intelligent searching exercises of the natural honey bees (Karaboga & Basturk, 2007).

In ABC algorithm, nutriment source location indicates a feasible solution for the optimization problem and the nectar value of a nutriment source resembles to the fitness of the solution (Karaboga & Akay, 2009). The set of the artificial bees is subdivided into three groups, namely employed bees, onlooker bees, and scout bees. The number of employed bees and onlooker bees are equal to the number of nutriment sources. A bee by standing for employed bees for taking a verdict about how to pick the food source is titled as onlooker bee. The employed bees arbitrarily search for the locations of the food source and share its knowledge with the onlooker bees, which halts at the hive. Scout bees search the new food sources arbitrarily depending upon the internal motivation (Karaboga & Akay, 2009).

ABC is an iterative algorithm similar to other state-of-art population-based metaheuristic algorithms. It involves sequence of the four phases namely, initialization of the swarm phase, employed bee phase, onlooker bee phase, and scout bee phase (Akay & Karaboga, 2012). The description of these phases is given below:-

2.1. *Initialization of the swarm phase:*

Firstly, ABC produces a uniformly distributed initial population of SN solutions, where every single solution (food source) x_i (i= 1, 2, . . . ; SN) is a D-dimensional vector. Here, D is the number of decision variables in the optimization problem and x_i is the i^{th} food source in the population. Food sources are produced as per the following Eq. 1:

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

Where, x_{minj} and x_{maxj} are bounds of x_i in j^{th} direction and rand [0, 1] is a uniformly distributed random number in [0, 1].

2.2. *Employed bee phase:*

In the course of this phase, each existing solution is modified based on the information provided by the knowledge of the individual and the fitness value of the recently produced solution, i.e. nectar quantity. If the fitness value of the recently produced solution is better than the earlier solution, the bee apprises its position with the recent one and rejects the previous one (Akay & Karaboga, 2012). The position update equation for i^{th} candidate solution is as follows:

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \tag{2}$$

Where, $k \in \{1, 2, ..., SN\}$ and $j \in \{1, 2, ..., D\}$ are arbitrarily chosen indices, k must be non-identical to i, and ϕ_{ij} is an arbitrary number in the range [-1, 1].

2.3. *Onlooker bee phase:*

The congregated knowledge is communicated by all the employed bees about the new fitness, i.e. nectar of the recently produced solutions (food sources) and their locus information with the onlooker bees in the hive. The available information is examined by the onlooker bees and they pick a solution with a probability p_i, associated to its fitness. The probability p_i is calculated as below:

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

There may be other choices of calculating p_i, but it must be the function of fitness. The fitness value of the i^{th} solution is fit_i. Alike the employed bee phase, it modifies the reformation in the position in its memory and computes for the fitness of the candidate source. In case, the recent fitness is higher than that of the earlier one, the bee memorizes the recently generated position and abandons the earlier one.

2.4. *Scout bee phase:*

The food source is considered to be deserted if the position of a food source is not updated up to a predefined threshold value, i.e number of cycles and then scout bee phase commences. In this phase, the food source is exchanged by a randomly picked food source within the specified area. Assume that the deserted food source is x_i and

$j \in \{1, 2, ..., D\}$ then the scout bee replaces this food source with x_i. This process can be described as follows:

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

Where, x_{minj} and x_{maxj} are bounds of x_i in j^{th} direction.

The pseudo-code of the ABC algorithm is presented by Algorithm 1:

Algorithm 1 Artificial Bee Colony Algorithm:

Initialize the parameters
while Termination criteria is not satisfying do

- Employed bee phase
- Onlooker bee phase
- Scout Bee Phase
- Memorize the best solution found so far

end while
Output the best solution found so far

3. Grasshopper inspired local search strategy

Grasshopper (GH) is an insect. The GHs or locusts have an extraordinary capacity of jumping, which separates them from other insects. Its name is a combination of two words grass and hopper, which implies that it can hop or jump on grass or any other base. GHs are commonly ground-habitat creepy crawlies with capable hind legs which empower them to escape from dangers by jumping, vivaciously. They ensure their security from enemies by camouflage when exposed, many species try to frighten the enemies with a very bright colored wing-flash during jumping and (if adult) launching them into the air, generally flying for just a short distance.

A large GH, for example, a locust jumps around a meter (One meter is equal to twenty body lengths of GH) without utilizing its pinions; the acceleration reaches the highest point at about 20 g (g is gravitational constant). GHs jump by enlarging their hefty posterior legs and propelling opposing the substratum (the base, a stick, an edge of lawn or whatsoever base GH are sitting on); the counteraction dynamism impels them in the midair. They bounce for various motives; to get away from an enemy, to attain trajectory path, or normally to proceed from one position to another position. To get the breakout jump, especially, there is robust finical pressure to exaggerate lift-off pace, since this depicts the span. This entails that the legs utterly propel opposite the base with both great force and a great pace of motion. In any case, an elementary feature of muscle is that it can't shrink with both great force and great pace, which appears as a trouble. The GHs defeat this evident counterstatement by utilizing a catapult operation to exaggerate the mechanical energy generated by their muscles (Offenbacher, 1970). The organism of GH is shown in Fig. 1.

The jump of GH takes place in three steps. To start with, the GH thoroughly expands the lower some portion of the leg (tibia) opposing the upper part (femur) by mobilizing the flexor tibiae muscle (the posterior legs of the young GH as shown in Fig. 1 in this elementary location). Further, there is a time of co-shrinking in which force raised up in the large, pennate extensor tibiae muscle, however, the tibia remains expanded via parallel shrinking of the flexor tibiae muscle. The extensor muscle is substantially athletic in comparison to the flexor muscle, however, the second one is facilitated by expertness in the joint that provide it a substantial operative mechanical power merit over the previous when the tibia is completely flexed. Co-shrinking can persist for as long as half a second, and amid this time the extensor muscle curtails and accumulates elastic strain energy by disfiguring stiff cuticular architecture in the leg. The extensor muscle shrinking is gradual (practically isometric), which permits it to establish great force (up to 14 Newton in the desert locust), but since it is gradual quite small power is required. The last step of the jump is the prompt loosening of the flexor muscle, which discharges the tibia from the flexed state. The resulting fast tibial extension is driven for the most part by the relaxation of the elastic architectures, rather than by further curtailing of the extensor muscle (Heitler, 1974). Hence, the stiff cuticle behaves similar to the elastic of a catapult or the bow of a bow-and-arrow. Energy is stored at small power by gradual but athletic muscle shrinking and recovered from the store at high power by fast relaxation of the mechanical elastic architectures (Bennet-Clark, 1975). If the effects of air resistance are overlooked, the motion of a hopping creature after it lifts-off the base is like the motion of a ball when it's thrown or a bullet after it's shot from a gun. This is called a ballistic movement, and the equations depicting the kinetics of such movements are well known, they were first derived by Isaac Newton in the seventeenth century (Hall, 1996).

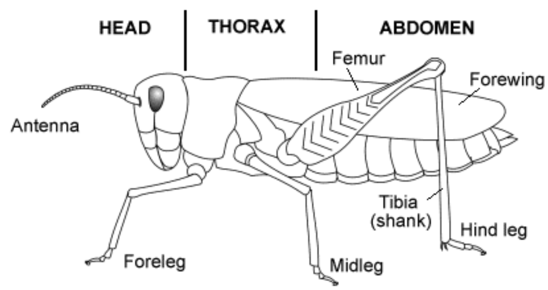

Figure 1: Grasshopper line curve[1]

$$R = \frac{V^2 \sin(2\theta)}{g} \tag{5}$$

The horizontal distance R that a ballistic projectile travels is related to the take-off angle θ and the velocity V at take-off: Where, g is the acceleration due to gravity ($9.81 m/s^2$). To maximize range, therefore, an animal should take off at 45° to the horizontal ($sin 2\theta = sin 90 = 1$). The key point is that if an animal takes off (or a bullet is fired) at this optimal angle of 45°, then its range is entirely dependent on its take-off velocity, whatever the size or weight of the animal.

As highly effective biological mechanisms are very common in nature, this article proposes a new local search strategy based upon the above GH jumping phenomenon and hybridized with ABC. The position update strategy is derived from the GH jumping distance. The distance R as mentioned in Eq. 5 is used as a new position of the best

[1] This figure is accessed on January 2018 from http://idtools.org/id/grasshoppers/glossary.php

solution which is going to update its position during the search process. The proposed local search strategy is named as grasshopper local search strategy (GHLS). The detailed description of GHLS strategy is as follows:-

In the proposed GHLS, Eq. 5 is adapted with some modifications as shown in Eq. 6. It is clear from this equation that the nearby search area of the best solution is exploited during the local search process.

$$x'_{bestj} = \sqrt{(x_{bestj})^2 + (x_{bestj} - x_{ij})^2} \times \sin(2\theta); \tag{6}$$

Where, i is a randomly selected solution from the population, x'_{bestj} is the updated position of the best solution of the swarm in j^{th} direction, and θ represents the angle of rotation. Here, $V^2 = \sqrt{(x_{bestj})^2 + (x_{bestj} - x_{ij})^2}$, which is derived from the self persistence and by inculcating the information from any randomly selected solution of the search space. The value of θ varies from $0°$ to $360°$. The value of θ is calculated as per the Eq. 7.

$$\theta = 10 \times t \tag{7}$$

Here, t represents the current iteration of the local search. The total number of local search iteration T is decided based upon an extensive analysis which is mentioned in the experimental setting. The pseudo-code of the proposed local search strategy GHLS is shown in Algorithm 2.

Algorithm 2 Grasshopper Local Search Strategy (GHLS):

Input optimization function $Minf(x)$;
Select the best solution x_{best} in the swarm which is going to modify its position;
Initialize iteration counter=0 and total iteration of GHLS, T;
while $(t < T)$ **do**
  Generate a new solution x'_{best} using Algorithm 3;
  Calculate the objective value $f(x'_{best})$;
  if $f(x'_{best}) < f(x_{best})$ **then**
    $x_{best} = x'_{best}$;
  end if
  $t = t + 1$;
end while

In Algorithm 3, C_r is the perturbation rate (between 0 and 1) which controls the amount of perturbation in the solution, $U(0, 1)$ is a uniform distributed random number between 0 and 1, D is the dimension of the problem.

4. Grasshopper inspired ABC (GHABC)

Local search strategies are hybridized with optimization algorithms in the hope to improve the exploitation capability of the algorithm. In this article, the developed GHLS strategy is incorporated into the ABC algorithm to improve the convergence speed of the ABC algorithm. The proposed algorithm is named as grasshopper inspired ABC (GHABC). The pseudo-code of the proposed GHABC algorithm is depicted in Algorithm 4.

Algorithm 3 New solution generation:

Input best solution x_{best} from the population;
Randomly select a solution x_i from the population;
Initialize the value of $\theta = 10 \times t$ /* t is the current iteration counter */
for $j = 1$ to D **do**
  if $U(0,1) < C_r$ **then**
    /* C_r is the perturbation rate, a constant in the range (0, 1) */
    $x'_{bestj} = x_{bestj}$;
  else
    $x'_{bestj} = \sqrt{(x_{bestj})^2 + (x_{bestj} - x_{ij})^2} \times \sin(2\theta)$;
  end if
end for
Return x'_{bestj}

Algorithm 4 Grasshopper inspired Artificial Bee Colony Algorithm (GHABC):

Initialize the parameters;
while Termination criteria **do**
  Step 1: Employed bee phase for generating new food sources;
  Step 2: Onlooker bee phase for updating the food sources depending on their nectar amounts;
  Step 3: Scout bee phase for discovering the new food sources in place of abandoned food sources;
  Step 4: Apply Grasshopper local search (GHLS) phase using Algorithm 2.
end while
Print best solution.

It is clear from the Algorithm 4 that the GHLS strategy is incorporated after the scout bee phase of the ABC algorithm. Therefore, in the proposed GHABC algorithm, the best solution found after executing the employed, onlooker, and scout bee phases, is given more chances to search in the vicinity with small step sizes to exploit the nearby area using the GHLS strategy. This will improve the exploitation capability of the ABC algorithm. Further, the incorporation of the GHLS strategy also improves the convergence ability of the ABC algorithm which makes, the proposed GHABC, a cost effective algorithm in terms of number of function evaluations.

5. Performance evaluation of GHABC algorithm

In this section the performance of the proposed GHABC algorithm is evaluated.

5.1. *Benchmark problems*

This set consists of 37 benchmark functions that are adopted from literature (Suganthan et al., 2005; Bansal, Sharma, Jadon, & Clerc, 2014; Bansal, Sharma, Arya, & Nagar, 2013; H. Sharma, Bansal, Arya, & Yang, 2016). The definition and characteristic of the functions are listed in Table 1.

5.2. *Parameter setting*

For validating the performance of the proposed GHABC algorithm, following experimental setting is adopted:

- The number of simulations/run =100,
- Colony size $NP = 50$ and Number of food sources $SN = NP/2$,
- $\phi_{ij} = rand[-1, 1]$ and limit=Dimension×Number of food sources=$D \times SN$ (Karaboga & Akay, 2011),
- The terminating criteria: Either acceptable error (AE), mentioned in Table 1, meets or maximum number of function evaluations (which is set to be 200000) is reached,
- Parameter settings for the algorithms, ABC (Karaboga, 2005), black hole ABC (BHABC) (N. Sharma, Sharma, Sharma, & Bansal, 2015), gbest guided ABC (GABC) (Zhu & Kwong, 2010), best so far ABC (BSFABC) (Banharnsakun, Achalakul, & Sirinaovakul, 2011), particle swarm optimization (PSO-2011) (Clerc & Kennedy, 2011), differential evolution (DE) (Storn & Price, 1997), spider monkey optimization (SMO) (Bansal et al., 2014), memetic ABC (MeABC) (Bansal, Sharma, Arya, & Nagar, 2013), $GbestDE$ (Mokan, Sharma, Sharma, & Verma, 2014), and levy flight ABC (LFABC) (H. Sharma et al., 2016) are same as their pioneer papers, respectively,
- To set termination criteria of GHLS, the performance of GHABC is measured for considered test problems on different values of T and results are analysed in terms of success in Fig. 2. It is clear from Fig. 2 that $T = 36$ gives better results (highest value of sum of success). Therefore, termination criteria is set to be $T = 36$,
- In order to investigate the impact of parameter C_r (perturbation rate of local search) depicted by Algorithm 3 on the performance of $GHABC$, its sensitivity with respect to various values of C_r in the range $[0.1, 1.0]$, is examined in the Fig. 3. It can be seen from Fig. 3 that the algorithm is exceptionally delicate towards c_r and it's value 0.6 gives comparatively better results. Therefore $c_r = 0.6$ is chosen for the experiments in this paper.

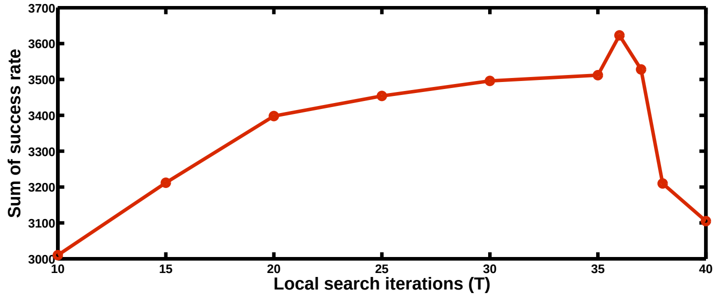

Figure 2: Variation in sum of success rate with local search iterations (T)

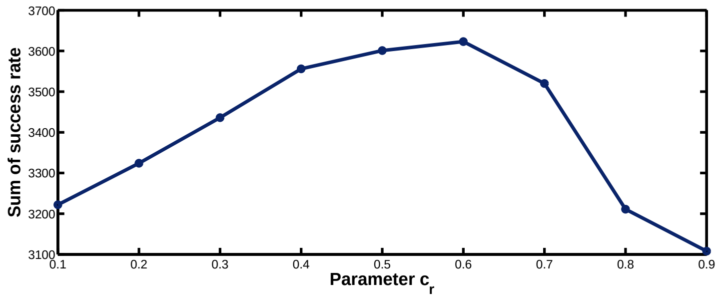

Figure 3: Effect of parameter c_r on success rate

Table 1: Test problems.D: Dimensions, C: Characteristic, U: Unimodal, M: Multimodal, S: Separable, N: Non-Separable, AE: Acceptable Error

Test Problem	**Objective function**	**Search Range**	**Optimum Value**	**D**	**AE**	**C**
Sphere	$f_1(x) = \sum_{i=1}^{D} x_i^2$	[-5.12 5.12]	$f(\vec{0}) = 0$	30	$1.0E-05$	S, U
De Jong f4	$f_2(x) = \sum_{i=1}^{D} i.(x_i)^4$	[-5.12 5.12]	$f(\vec{0}) = 0$	30	$1.0E-05$	S, M
Griewank	$f_3(x) = 1 + \frac{1}{4000}\sum_{i=1}^{D} x_i^2 - \prod_{i=1}^{D}\cos(\frac{x_i}{\sqrt{i}})$	[-600 600]	$f(\vec{0}) = 0$	30	$1.0E-05$	N, M
Rastrigin	$f_4(x) = 10D + \sum_{i=1}^{D}[x_i^2 - 10\cos(2\pi x_i)]$	[-5.12 5.12]	$f(\vec{0}) = 0$	30	$1.0E-05$	N, M
Ackley	$f_5(x) = -20 + e + exp(-\frac{0.2}{D}\sqrt{\sum_{i=1}^{D} {x_i}^3})$ $-exp(\frac{1}{D}\sum_{i=1}^{D}\cos\,(2\pi x_i)x_i)$	[-1 1]	$f(\vec{0}) = 0$	30	$1.0E-05$	N, M
Alpine	$f_6(x) = \sum_{i=1}^{D} \lvert x_i \sin\, x_i + 0.1x_i \rvert$	[-10 10]	$f(\vec{0}) = 0$	30	$1.0E-05$	S, M
Michalewicz	$f_7(x) = -\sum_{i=1}^{D} \sin\, x_i(\sin\,(\frac{i{x_i}^2}{\pi})^{20})$	[0 π]	$f_{min} = -9.66015$	10	$1.0E-05$	N, M
Cosine Mixture	$f_8(x) = \sum_{i=1}^{D} {x_i}^2 - 0.1(\sum_{i=1}^{D}\cos\, 5\pi x_i) + 0.1D$	[-1 1]	$f(\vec{0}) = -D \times 0.1$	30	$1.0E-05$	S, M
Exponential	$f_9(x) = -(exp(-0.5\sum_{i=1}^{D} {x_i}^2)) + 1$	[-1 1]	$f(\vec{0}) = -1$	30	$1.0E-05$	N, M
Zakharov	$f_{10}(x) = \sum_{i=1}^{D} {x_i}^2 + (\sum_{i=1}^{D}\frac{ix_i}{2})^2 + (\sum_{i=1}^{D}\frac{ix_1}{2})^4$	[-5.12 5.12]	$f(\vec{0}) = 0$	30	$1.0E-02$	N, M
Cigar	$f_{11}(x) = {x_0}^2 + 100000\sum_{i=1}^{D} {x_i}^2$	[-10 10]	$f(\vec{0}) = 04$	30	$1.0E-05$	S, U
brown3	$f_{12}(x) = \sum_{i=1}^{D-1}({x_i}^{2^{(x_{i+1})^2+1}} + {x_{i+1}}^{2^{{x_i}^2+1}})$	[-1 4]	$f(\vec{0}) = 0$	30	$1.0E-05$	N, U
Schewel	$f_{13}(x) = \sum_{i=1}^{D} \lvert x_i\rvert + \prod_{i=1}^{D} \lvert x_i\rvert$	[-10 10]	$f(\vec{0}) = 0$	30	$1.0E-05$	N, U
Salomon Problem	$f_{14}(x) = 1 - cos(2\pi\sqrt{\sum_{i=1}^{D} x_i^2}) + 0.1(\sqrt{\sum_{i=1}^{D} x_i^2})$	[-100 100]	$f(\vec{0}) = 0$	30	$1.0E-01$	N, M
Axis parallel hyper-ellipsoid	$f_{15}(x) = \sum_{i=1}^{D} i \times x_i^2$	[-5.12 5.12]	$f(\vec{0}) = 0$	30	$1.0E-05$	S, U
Sum of different powers	$f_{16}(x) = \sum_{i=1}^{D} \lvert x_i\rvert^{i+1}$	[-1 1]	$f(\vec{0}) = 0$	30	$1.0E-05$	S, M
Step function	$f_{17}(x) = \sum_{i=1}^{D} (\lfloor x_i + 0.5\rfloor)^2$	[-100 100]	$f(-0.5 \leq x \leq 0.5) = 0$	30	$1.0E-05$	S, U
Inverted cosine wave	$f_{18}(x) = -\sum_{i=1}^{D-1}\left(\exp\left(\frac{-(x_i^2 + x_{i+1}^2 + 0.5x_i x_{i+1})}{8}\right) \times \mathrm{I}\right)$	[-5 5]	$f(\vec{0}) = -D + 1$	10	$1.0E-05$	N, M
Neumaier 3 Problem (NF3)	$f_{19}(x) = \sum_{i=1}^{D} (x_i - 1)^2 - \sum_{i=2}^{D} x_i x_{i-1}$	[$-D^2$ D^2]	$f_{min} = -\frac{(D(D+4)(D-1))}{6}$	10	$1.0E-01$	N, U
Rotated hyper-ellipsoid	$f_{20}(x) = \sum_{i=1}^{D}\sum_{j=1}^{i} x_j^2$	[-65.536 65.536]	$f(\vec{0}) = 0$	30	$1.0E-05$	S, M
Levy montalvo 1	$f_{21}(x) = \frac{\pi}{D}(10\sin^2(\pi y_1) + \sum_{i=1}^{D-1}(y_i - 1)^2(1 + 10\sin^2(\pi y_{i+1})) + (y_D - 1)^2)$, where $y_i = 1 + \frac{1}{4}(x_i + 1)$	[-10 10]	$f(\vec{-1}) = 0$	30	$1.0E-05$	N, M
Ellipsoidal	$f_{22}(x) = \sum_{i=1}^{D} (x_i - i)^2$	[-30 30]	$f(1, 2, 3, \ldots, D) = 0$	30	$1.0E-05$	S, U
Beale function	$f_{23}(x) = [1.5 - x_1(1 - x_2)]^2 + [2.25 - x_1(1 - x_2^2)]^2 + [2.625 - x_1(1 - x_2^3)]^2$	[-4.5 4.5]	$f(3, 0.5) = 0$	2	$1.0E-05$	N, M
Colville function	$f_{24}(x) = 100[x_2 - x_1^2]^2 + (1 - x_1)^2 + 90(x_4 - x_3^2)^2 + (1 - x_3)^2 + 10.1[(x_2 - 1)^2 + (x_4 - 1)^2] + 19.8(x_2 - 1)(x_4 - 1)$	[-10 10]	$f(\vec{1}) = 0$	4	$1.0E-05$	N, M

to be cont'd on next page

Table 1: Test problems. D: Dimensions, C: Characteristic, U: Unimodal, M: Multimodal, S: Separable, N: Non-Separable, AE: Acceptable Error

Test Problem	**Objective function**	**Search Range**	**Optimum Value**	**D**	**AE**	**C**
Kowalik	$f_{25}(x) = \sum_{i=1}^{23}[a_i - \frac{x_1(b_i^2+b_ix_2)}{b_i^2+b_ix_3+x_4}]^2$	[-5 5]	$f(0.192833, 0.190836, 0.123117, 0.135766) = 0.000307486$	4	$1.0E-05$	N, M
2D Tripod function	$f_{26}(x) = p(x_2)(1+p(x_1)) + \|(x_1 + 50p(x_2)(1-2p(x_1)))\| + \|(x_2 + 50(1-2p(x_2)))\|$	[-100 100]	$f(0, -50) = 0$	2	$1.0E-04$	N, M
Shifted Rosenbrock	$f_{27}(x) = \sum_{i=1}^{D-1}(100(z_i^2 - z_{i+1})^2 + (z_i-1)^2) + f_{bias},\ z = x - o + 1,\ x = [x_1, x_2, ...x_D],\ o = [o_1, o_2, ...o_D]$	[-100 100]	$f(o) = f_{bias} = 390$	10	$1.0E-01$	S, M
Shifted Sphere	$f_{28}(x) = \sum_{i=1}^{D} z_i^2 + f_{bias},\ z = x - o\ , x = [x_1, x_2, ...x_D],\ o = [o_1, o_2, ...o_D]$	[-100 100]	$f(o) = f_{bias} = -450$	10	$1.0E-05$	S, M
Shifted Griewank	$f_{29}(x) = \sum_{i=1}^{D} \frac{z_i^2}{4000} - \prod_{i=1}^{D} \cos(\frac{z_i}{\sqrt{i}}) + 1 + f_{bias},\ z = (x-o),\ x = [x_1, x_2, ...x_D],\ o = [o_1, o_2, ...o_D]$	[-600 600]	$f(o) = f_{bias} = -180$	10	$1.0E-05$	N, M
Shifted Ackley	$f_{30}(x) = -20\exp(-0.2\sqrt{\frac{1}{D}\sum_{i=1}^{D} z_i^2}) - \exp(\frac{1}{D}\sum_{i=1}^{D}\cos(2\pi z_i)) + 20 + e + f_{bias},\ z = (x-o),\ x = (x_1, x_2, ...x_D),\ o = (o_1, o_2, ...o_D)$	[-32 32]	$f(o) = f_{bias} = -140$	10	$1.0E-05$	S, M
Easom's function	$f_{31}(x) = -\cos x_1 \cos x_2 e^{((-(x_1-\pi)^2-(x_2-\pi)^2))}$	[-10 10]	$f(\pi, \pi) = -1$	2	$1.0E-13$	S, M
Dekkers and Aarts	$f_{32}(x) = 10^5x_1^2 + x_2^2 - (x_1^2+x_2^2)^2 + 10^{-5}(x_1^2+x_2^2)^4$	[-20 20]	$f(0, 15) = f(0, -15) = -24777$	2	$5.0E-01$	N, M
McCormick	$f_{33}(x) = \sin(x_1+x_2) + (x_1-x_2)^2 - \frac{3}{2}x_1 + \frac{5}{2}x_2 + 1$	$-1.5 \leq x_1 \leq 4, -3 \leq x_2 \leq 3$	$f(-0.547, -1.547) = -1.9133$	30	$1.0E-04$	N, M
Meyer and Roth Problem	$f_{34}(x) = \sum_{i=1}^{5}(\frac{x_1x_3t_i}{1+x_1t_i+x_2v_i} - y_i)^2$	[-10 10]	$f(3.13, 15.16, 0.78) = 0.4E-04$	3	$1.0E-03$	N, U
Shubert	$f_{35}(x) = -\sum_{i=1}^{5} i\cos((i+1)x_1+1)\sum_{i=1}^{5} i\cos((i+1)x_2+1)$	[-10 10]	$f(7.0835, 4.8580) = -186.7309$	2	$1.0E-05$	S, M
Sinusoidal	$f_{36}(x) = -[A\prod_{i=1}^{D}\sin(x_i - z) + \prod_{i=1}^{D} sin(B(x_i - z))],\ A = 2.5, B = 5, z = 30$	[0 180]	$f(90 \vec{+} z) = -(A+1)$	10	$1.0E-02$	N, M
Moved axis parallel hyper-ellipsoid	$f_{37}(x) = \sum_{i=1}^{D} 5i \times x_i^2$	[-5.12 5.12]	$f(x) = 0; x(i) = 5 \times i, i = 1 : D$	30	$1.0E-15$	S, U

5.3. *Results comparison for benchmark problems*

For validating the performance of the proposed GHABC algorithm, it is compared with the basic version of ABC (Karaboga, 2005), black hole ABC (BHABC) (N. Sharma et al., 2015), gbest guided ABC (GABC) (Zhu & Kwong, 2010), best so far ABC (BSFABC) (Banharnsakun et al., 2011), particle swarm optimization (PSO-2011) (Kennedy, 2011), differential evolution (DE) (Storn & Price, 1997), gbest DE (Mokan et al., 2014), spider monkey optimization (SMO) (Bansal et al., 2014), memetic ABC (MeABC) (Bansal, Sharma, Arya, & Nagar, 2013), and levy flight ABC (LFABC) (H. Sharma et al., 2016). The comparison is performed in terms of four parameters that are standard deviation (SD), mean error (ME), average number of function evaluations (AFE), and success rate (SR). The reported results are demonstrated in Table 2. The obtained outcomes demonstrate that GHABC is competitive than ABC and other considered SI based algorithms for greater part of the benchmark test problems (TPs) independent of their tendency either as far as separability, modality, and other parameters.

The proposed algorithm is also assessed by Mann-Whitney U rank sum test (A. Sharma, Sharma, Bhargava, & Sharma, 2016a), acceleration rate (AR) (A. Sharma, Sharma, Bhargava, & Sharma, 2016b), boxplots analysis (BP) (A. Sharma et al., 2016a), and success performance (SP) (Qu, Liang, Suganthan, & Chen, 2014). The Mann-Whitney U rank sum test is applied on AFEs. For all the considered algorithms the experiment is performed at 5% significance level ($\alpha = 0.05$) and the outcomes for 100 runs are recorded in Table 4.In this table, '+' sign speaks to that GHABC is predominant in examination with the other considered algorithm while '-' sign demonstrates that the other considered algorithm is unrivaled.

Table 4 shows that, in comparison with the other considered significant algorithms GHABC has high caliber than all other considered algorithms for 20 TPs including $f_1 - f_6$, f_{10}, f_8, f_{18}, f_{20} and f_{26} and f_{37} . $GHABC$ performs better than basic ABC for 33 TPs, $f_1 - f_6$, $f_8 - f_{20}$, $f_{23} - f_{27}$, and $f_{29} - f_{37}$. The GHABC shows better results for 27 TPs when compared with BHABC algorithm, $f_1 - f_6$, $f_8 - f_{18}$, f_{20}, $f_{25} - f_{26}$, $f_{29} - f_{32}$, and $f_{35} - f_{37}$. The $GHABC$ performs better for 25 TPs, $f_1 - f_6$, $f_8 - f_{20}$, $f_{24} - f_{26}$, f_{31}, and $f_{36} - f_{37}$ in comparison with GABC. The GHABC performs better for 35 TPs in comparison with BSFABC, $f_1 - f_{30}$, $f_{32} - f_{33}$, and $f_{35} - f_{37}$. In comparison with PSO-2011, GHABC performs better on 31 TPs, $f_1 - f_{18}$, $f_{20} - f_{23}$, f_{26}, $f_{28} - f_{30}$, $f_{32} - f_{33}$, and $f_{35} - f_{37}$. The outcomes for GHABC are better for 29 TPs in comparison with DE, $f_1 - f_{18}$, f_{20}, $f_{25} - f_{29}$, $f_{32} - f_{33}$, and $f_{35} - f_{37}$. The GHABC shows better results for 30 TPs, $f_1 - f_{20}$, $f_{25} - f_{27}$, f_{29}, $f_{31} - f_{33}$, and $f_{35} - f_{37}$ when compared with GbestDE algorithm. The $GHABC$ performs better for 32 TPs, $f_1 - f_{18}$, $f_{20} - f_{22}$, $f_{25} - f_{30}$, $f_{32} - f_{33}$, and $f_{35} - f_{37}$ in comparison with SMO algorithm. The outcomes for $GHABC$ are better for 30 TPs in comparison with MeABC, $f_1 - f_6$, $f_8 - f_{18}$, f_{20}, f_{21}, f_{23}, $f_{25} - f_{26}$, $f_{28} - f_{34}$, and f_{37}. While comparing with LFABC, GHABC shows better results for 23 TPs, $f_1 - f_{18}$, f_{20}, f_{25}, f_{26}, f_{31}, and f_{37}.

The above investigation speaks to that GHABC is a focused candidate in the region of SI based techniques.

Table 2: Comparison of the results of benchmark test problems

Test Problem	Measure	GHABC	ABC	BHABC	GABC	BSFABC	PSO-2011	DE	GbestDE	SMO	MeABC	LFABC
f_1	SD	3.09E-06	1.56E-06	1.44E-06	1.81E-06	2.15E-06	7.55E-07	8.24E-07	6.64E-07	6.10E-07	8.10E-07	1.73E-06
	ME	3.91E-06	8.48E-06	8.53E-06	8.11E-06	7.49E-06	9.17E-06	9.06E-06	9.17E-06	9.33E-06	9.19E-06	8.39E-06
	AFE	642.32	13963.77	22304.92	14347.5	30063	38346	22444	15315.5	38101.5	19659.92	16733.85
	SR	100	100	100	100	100	100	100	100	100	100	100
f_2	SD	2.41E-06	2.92E-06	2.63E-06	2.72E-06	3.12E-06	1.09E-06	8.51E-07	1.22E-06	8.62E-07	1.30E-06	3.02E-06
	ME	1.55E-06	5.46E-06	5.75E-06	5.51E-06	5.31E-06	8.99E-06	9.01E-06	8.51E-06	9.03E-06	8.67E-06	6.62E-06
	AFE	481.61	5629.43	8687.05	8388	24524.5	32442	20859.5	12668.5	32596.5	6112.82	9556.12
	SR	100	100	100	100	100	100	100	100	100	100	100
f_3	SD	3.03E-06	2.10E-03	1.03E-03	3.00E-06	2.97E-06	7.34E-03	4.52E-03	6.63E-07	7.12E-03	1.44E-06	2.03E-06
	ME	3.11E-06	4.26E-04	1.54E-04	6.07E-06	5.67E-06	4.91E-03	2.05E-03	9.17E-06	3.87E-03	8.79E-06	7.95E-06
	AFE	965.7	66525.21	44066	30455.3	62936.12	73624.5	64036.5	29589	113502.5	43249.74	40722.51
	SR	100	96	97	100	100	63	81	100	84	100	100
f_4	SD	2.77E-06	2.34E-06	2.88E-06	2.75E-06	3.11E-06	2.24E+01	5.71E+00	5.92E+00	1.40E+01	1.76E-06	2.41E-06
	ME	3.05E-06	7.58E-06	5.79E-06	6.38E-06	4.05E-06	4.30E+01	1.46E+01	3.99E+00	3.87E+01	8.29E-06	7.18E-06
	AFE	848.38	39982.67	44384.12	34805	122759.5	100050	200050	189893.5	200050	57689.92	40644.63
	SR	100	100	100	100	100	0	0	33	100	100	100
f_5	SD	2.86E-06	1.89E-06	1.85E-06	1.23E-06	1.74E-06	3.54E-07	3.94E-07	4.05E-07	3.66E-07	3.63E-07	1.16E-06
	ME	5.00E-06	8.12E-06	8.26E-06	8.91E-06	8.28E-06	9.67E-06	9.46E-06	9.58E-06	9.69E-06	9.64E-06	9.02E-06
	AFE	1193.77	62447.06	101840.09	30549	72368.5	77172.5	42699	28916	77352	64481.82	35985.01
	SR	100	100	100	100	100	100	100	100	100	100	100
f_6	SD	2.65E-06	2.66E-06	1.60E-06	1.85E-06	6.05E-06	1.55E+00	4.40E-07	4.01E-07	3.23E-07	1.63E-06	1.03E-05
	ME	4.58E-06	7.82E-06	8.46E-06	8.32E-06	8.03E-06	2.30E-01	9.43E-06	9.54E-06	9.63E-06	8.57E-06	9.06E-06
	AFE	1030.21	75594.46	59016.04	54665.5	142277	90070	60983	51527	93046.5	104485.84	85238.42
	SR	100	100	100	100	96	72	100	100	98	100	98
f_7	SD	3.96E-06	3.42E-06	3.71E-06	3.75E-06	3.56E-06	4.20E-01	4.84E-02	1.73E-02	2.34E-01	3.65E-06	1.31E-02
	ME	4.35E-06	4.80E-06	4.32E-06	4.37E-06	3.86E-06	4.20E-01	4.90E-02	3.69E-03	3.12E-01	5.61E-06	4.65E-03
	AFE	43060.21	20222.8	28283.24	20048.82	45347.49	99402.5	167536	45484	198326	21681.84	43496.55
	SR	100	100	100	100	100	2	23	92	100	100	88
f_8	SD	2.54E-06	2.02E-06	2.39E-06	1.91E-06	2.43E-06	6.29E-02	2.90E-02	6.82E-07	5.68E-02	9.54E-07	2.22E-06
	ME	2.78E-06	8.33E-06	7.72E-06	7.83E-06	6.97E-06	2.51E-02	5.92E-03	9.14E-06	2.22E-02	9.17E-06	7.84E-06
	AFE	638.34	13632.1	35006.99	15420.5	32039	49744	30339	15464.5	63043.5	23565.56	17862.88
	SR	100	100	100	100	100	85	96	100	88	100	100
f_9	SD	2.91E-06	1.85E-06	1.92E-06	1.53E-06	1.96E-06	6.08E-07	7.39E-07	8.54E-07	6.15E-07	6.83E-07	1.73E-06
	ME	3.07E-06	8.15E-06	8.00E-06	8.18E-06	7.74E-06	9.32E-06	8.99E-06	9.10E-06	9.33E-06	9.30E-06	8.16E-06
	AFE	508.21	7160.7	17656.91	11875	18678.5	28182.5	17018	11765	28227.5	9987.18	14205.4
	SR	100	100	100	100	100	100	100	100	100	100	100
f_{10}	SD	2.53E-03	1.61E+01	1.84E+01	1.58E+01	1.22E+01	1.63E+00	5.20E-04	8.13E-04	1.80E-02	5.03E-04	1.57E+01
	ME	5.33E-03	6.11E+01	1.01E+02	9.76E+01	8.38E+01	2.60E+00	9.47E-03	9.23E-03	2.20E-02	9.56E-03	1.13E+02
	AFE	2821.42	200025.72	200000.31	200000.01	200000	100050	68154.5	171519.5	196434	100752.87	200040
	SR	100	0	0	0	0	0	100	100	100	99	0
f_{11}	SD	2.99E-06	2.19E-06	2.31E-06	1.87E-06	2.46E-06	7.36E-07	8.77E-07	8.15E-07	6.96E-07	1.22E-06	1.65E-06
	ME	3.15E-06	7.96E-06	7.55E-06	7.83E-06	7.24E-06	9.27E-06	8.89E-06	9.11E-06	9.29E-06	8.90E-06	8.84E-06
	AFE	1138.09	43029.9	61286.27	23043	62034.5	68942.5	39664.5	27123.5	69125.5	47579.82	24546.79
	SR	100	100	100	100	100	100	100	100	100	100	100
f_{12}	SD	2.94E-06	1.60E-06	2.06E-06	1.97E-06	1.99E-06	6.08E-07	9.48E-07	7.40E-07	6.26E-07	9.02E-07	1.58E-06
	ME	3.41E-06	8.36E-06	8.04E-06	7.86E-06	7.73E-06	9.23E-06	8.94E-06	9.09E-06	9.24E-06	9.12E-06	8.55E-06
	AFE	634.36	14830.27	23739.99	14076	31207.5	35048	22003.5	15034	35048.5	20632.76	16111.3
	SR	100	100	100	100	100	100	100	100	100	100	100

to be cont'd on next page

Table 2: Comparison of the results of benchmark test problems (Cont.)

Test Problem	Measure	GHABC	ABC	BHABC	GABC	BSFABC	PSO-2011	DE	GbestDE	SMO	MeABC	LFABC
f_{13}	SD	2.66E-06	6.60E-07	8.78E-07	7.84E-07	1.45E-06	4.08E-07	5.43E-07	4.53E-07	3.09E-07	3.43E-07	8.04E-07
	ME	5.20E-06	9.45E-06	9.24E-06	9.22E-06	8.68E-06	9.56E-06	9.32E-06	9.48E-06	9.61E-06	9.63E-06	9.34E-06
	AFE	1201.97	48054.88	122848.9	27693	53027.5	70901.5	45017	26715	70794.5	54120.92	30994.7
	SR	100	100	100	100	100	100	100	100	100	100	100
f_{14}	SD	2.32E-01	8.66E-02	4.45E-02	3.35E-02	6.82E-02	8.01E-02	9.95E-03	1.40E-02	5.53E-02	3.82E-02	4.45E-02
	ME	6.76E-01	9.75E-01	9.33E-01	9.33E-01	9.56E-01	3.98E-01	2.01E-01	2.02E-01	2.88E-01	9.22E-01	9.39E-01
	AFE	346.93	139235.63	94411.64	85618.12	186319.67	100003	58843	104523.5	200050	23006.5	101452.43
	SR	100	57	97	95	73	1	99	97	13	100	87
f_{15}	SD	2.95E-06	1.96E-06	2.33E-06	1.97E-06	2.32E-06	7.15E-07	8.56E-07	7.07E-07	6.37E-07	8.73E-07	1.70E-06
	ME	3.47E-06	8.08E-06	7.94E-06	8.01E-06	7.13E-06	9.24E-06	9.00E-06	9.11E-06	9.33E-06	9.15E-06	8.43E-06
	AFE	738.76	19417.41	25433.47	15925	36685.5	43706	25889	17655	44374.5	25677.84	18093.08
	SR	100	100	100	100	100	100	100	100	100	100	100
f_{16}	SD	3.12E-06	2.83E-06	2.40E-06	2.60E-06	2.72E-06	9.32E-02	2.20E-06	1.93E-06	1.38E-06	3.82E-01	3.13E-06
	ME	2.84E-06	4.90E-06	6.55E-06	6.12E-06	5.84E-06	2.75E+00	7.15E-06	7.43E-06	8.48E-06	3.23E+00	5.86E-06
	AFE	359.31	19776.19	7104.52	9392.5	14434	100050	7795.5	5704	9897	200024.31	7523.66
	SR	100	100	100	100	100	0	100	100	100	0	100
f_{17}	SD	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	1.41E-06	3.32E-01	0.00E+00	0.00E+00	3.07E-06	0.00E+00
	ME	0.00E+00	0.00E+00	0.00E+00	0.00E+00	0.00E+00	8.33E-06	1.00E-01	0.00E+00	0.00E+00	5.13E-06	0.00E+00
	AFE	474.54	7089.94	9030.52	8951	36988	9786.5	33846.5	10470	35050	7240.66	10863.2
	SR	100	100	100	100	100	100	91	100	100	100	100
f_{18}	SD	2.71E-06	7.33E-02	2.33E-06	2.39E-06	1.99E-01	5.56E-01	6.30E-01	1.56E-06	6.06E-01	4.54E-01	4.27E-05
	ME	2.93E-06	1.06E-02	7.20E-06	6.83E-06	6.09E-02	1.04E+01	8.93E-01	8.24E-06	1.40E+00	9.31E+00	1.27E-05
	AFE	659.72	114061.2	70078.06	47688.66	123141.06	100050	176110.5	47035	198670	200010.81	42442.21
	SR	100	84	100	100	85	0	17	100	100	0	99
f_{19}	SD	1.52E-01	3.12E+00	4.96E-02	1.19E+00	5.34E+00	6.05E-01	1.41E-06	4.89E-02	4.30E-07	2.71E-04	6.84E-02
	ME	1.14E-01	3.20E+00	1.00E-01	1.21E+00	4.16E+00	1.58E+00	8.25E-06	9.02E-03	9.57E-06	3.52E-05	1.07E-01
	AFE	121685.11	200052.16	110353.66	196770.68	199822.64	99659.5	17251	165772	67426.5	49036.82	39650.81
	SR	99	0	96	5	1	2	100	65	85	99	95
f_{20}	SD	3.03E-06	2.14E-06	2.56E-06	2.01E-06	2.29E-06	6.74E-07	8.63E-07	8.21E-07	8.13E-07	1.40E-02	1.99E-06
	ME	3.23E-06	7.84E-06	6.97E-06	7.74E-06	7.29E-06	9.46E-06	8.90E-06	9.08E-06	9.20E-06	8.56E-02	8.51E-06
	AFE	948.06	31029.82	28894.12	19477	49425	65973.5	32927	22365	56547	36913.97	21192.18
	SR	100	100	100	100	100	100	100	100	100	100	100
f_{21}	SD	1.99E-06	2.85E-06	1.75E-06	2.13E-06	2.52E-06	6.40E-07	7.33E-07	7.93E-07	1.77E-02	1.15E-06	1.75E-06
	ME	7.75E-06	6.93E-06	7.88E-06	7.84E-06	6.72E-06	9.34E-06	9.15E-06	9.07E-06	3.12E-03	8.95E-06	8.27E-06
	AFE	21739.1	10355.32	12163.02	13128.5	26570.5	56626.5	19941	14373	37764.5	36702.66	15263.63
	SR	100	100	100	100	100	100	100	100	100	100	100
f_{22}	SD	1.81E-06	2.38E-06	2.33E-06	1.81E-06	2.63E-06	4.03E-03	7.39E-07	7.81E-07	5.56E-07	6.71E-07	1.97E-06
	ME	7.83E-06	7.44E-06	7.71E-06	7.93E-06	7.11E-06	1.77E-03	9.07E-06	9.12E-06	9.33E-06	9.33E-06	8.07E-06
	AFE	32466.2	23156.54	23504.76	16625.5	40983.5	46168	27209	17831.5	44306	19093.96	18653.59
	SR	100	100	100	100	100	84	100	100	100	100	100
f_{23}	SD	2.96E-06	2.73E-06	2.95E-06	2.93E-06	1.69E-05	7.58E-07	2.91E-06	3.15E-06	2.81E-06	1.21E-06	2.84E-06
	ME	5.34E-06	7.24E-06	5.49E-06	5.33E-06	1.28E-05	9.19E-06	4.95E-06	5.23E-06	4.96E-06	9.01E-06	7.52E-06
	AFE	14849.56	34002.38	7259.59	8701.35	49064.36	44060.5	1413	4454.5	2753.5	29262.18	3746.11
	SR	100	100	100	100	92	100	100	100	100	100	100
f_{24}	SD	5.74E-03	1.07E-01	1.89E-03	1.42E-02	3.19E-02	2.83E-06	3.41E-01	2.65E-03	2.24E-04	2.87E-06	1.29E-03
	ME	1.22E-02	1.56E-01	8.26E-03	1.63E-02	2.62E-02	4.83E-06	4.62E-02	1.75E-03	8.13E-04	4.24E-06	9.19E-03
	AFE	138652.39	200085.97	63024.92	159243.54	153739	2715	22950	105190.5	48776.5	5358.3	65107.64
	SR	57	0	99	42	44	100	91	70	100	100	100

to be cont'd on next page

Table 2: Comparison of the results of benchmark test problems (Cont.)

Test Problem	Measure	GHABC	ABC	BHABC	GABC	BSFABC	PSO-2011	DE	GbestDE	SMO	MeABC	LFABC
f_{25}	SD	2.04E-05	7.16E-05	1.65E-05	2.31E-05	7.91E-05	3.26E-05	3.64E-04	1.80E-04	1.18E-05	7.30E-06	1.79E-04
	ME	8.01E-05	1.69E-04	8.71E-05	8.76E-05	1.53E-04	4.52E-05	2.82E-04	1.92E-04	8.97E-05	6.59E-06	1.37E-04
	AFE	6549.21	182713.19	62584.17	99509.94	150752.09	2347.5	63860	154021	35865	41316.66	61386.26
	SR	100	21	100	92	45	100	70	40	100	80	95
f_{26}	SD	3.12E-05	2.73E-05	2.54E-05	2.29E-05	1.98E-04	1.13E-05	1.40E-01	1.92E-02	2.71E-01	3.19E-05	2.37E-01
	ME	4.58E-05	5.87E-05	6.29E-05	6.36E-05	8.56E-05	9.03E-05	2.00E-02	1.93E-03	8.01E-02	9.05E-05	6.01E-02
	AFE	4467.44	18836.4	11497.4	8726.06	6208.54	37771	8249.5	41654	29745.5	96233.48	17885.73
	SR	100	100	100	100	99	100	98	97	100	86	94
f_{27}	SD	2.87E-01	9.28E-01	3.66E+00	4.72E-02	3.95E+00	3.67E-01	2.25E+00	3.57E+00	1.08E+01	2.45E-05	7.64E-01
	ME	1.90E-01	5.36E-01	6.67E-01	9.33E-02	2.32E+00	1.64E-01	2.25E+00	3.69E+00	2.92E+00	6.11E-05	2.53E-01
	AFE	137809.2	175910.33	120130.27	104982	191465.22	29248	186125	181873	187162.5	9471.7	66632.89
	SR	72	24	80	94	11	83	8	14	42	100	95
f_{28}	SD	2.04E-06	2.57E-06	2.28E-06	2.13E-06	2.49E-06	2.51E+01	1.52E-06	1.62E-06	1.50E-06	1.47E+00	2.36E-06
	ME	7.13E-06	6.93E-06	7.22E-06	7.07E-06	6.94E-06	8.38E+00	8.06E-06	7.93E-06	8.29E-06	7.76E-01	7.27E-06
	AFE	9552.14	8922.32	8838.87	5577.5	18117	98430.5	10364.5	7751	15785.5	148560.23	6203.32
	SR	100	100	100	100	100	3	100	100	100	39	100
f_{29}	SD	1.42E-03	3.02E-03	2.39E-03	7.35E-04	6.18E-03	4.61E+03	1.38E-02	1.62E-02	2.87E-02	3.60E+03	7.35E-04
	ME	2.62E-04	1.16E-03	8.61E-04	7.90E-05	4.58E-03	2.17E+03	1.37E-02	1.71E-03	4.05E-02	1.29E+04	8.01E-05
	AFE	67282.34	87111.25	91174.8	42366.85	118467.79	100050	153524	68046	197491	200018.88	40382.88
	SR	95	85	88	99	58	0	30	98	81	0	99
f_{30}	SD	1.52E-06	1.85E-06	2.05E-06	1.28E-06	1.93E-06	5.61E-02	8.90E-07	9.32E-07	1.05E-06	3.13E-06	1.34E-06
	ME	8.30E-06	8.09E-06	7.68E-06	8.64E-06	8.13E-06	6.59E-02	8.90E-06	8.91E-06	8.93E-06	5.98E-06	8.66E-06
	AFE	15737.84	23391.88	71048.93	9321	31326.5	100050	15453.5	11739.5	24630	35602.37	10934.63
	SR	100	100	100	100	100	0	100	100	100	100	100
f_{31}	SD	3.06E-14	3.46E-05	2.97E-14	2.81E-14	3.00E-14	2.96E-14	3.02E-14	2.98E-14	2.92E-14	1.35E-07	3.28E-14
	ME	4.27E-14	9.53E-06	4.97E-14	4.29E-14	3.88E-14	5.35E-14	4.79E-14	4.59E-14	4.82E-14	2.03E-08	5.60E-14
	AFE	11240.27	188862.04	86195.05	48895.67	4677.1	9773.5	4815	11289	9796.5	84658.38	14065.55
	SR	100	13	100	100	100	100	100	100	100	82	100
f_{32}	SD	5.52E-03	5.52E-03	5.77E-03	5.37E-03	5.28E-03	5.42E-03	5.14E-03	5.03E-03	5.55E-03	1.39E-05	5.68E-03
	ME	4.88E-01	4.89E-01	4.91E-01	4.90E-01	4.91E-01	4.91E-01	4.90E-01	4.91E-01	4.92E-01	1.91E-05	4.91E-01
	AFE	885.19	3145.86	946.24	775	2800.72	4966.5	2154.5	2550.5	5050	120379.15	687.8
	SR	100	100	100	100	100	100	100	100	100	40	100
f_{33}	SD	6.60E-06	6.95E-06	6.65E-06	6.43E-06	6.44E-06	6.61E-06	6.52E-06	6.75E-06	6.86E-06	5.61E-03	6.96E-06
	ME	8.82E-05	8.80E-05	8.83E-05	8.85E-05	8.71E-05	8.80E-05	8.80E-05	8.86E-05	8.84E-05	4.89E-01	9.04E-05
	AFE	922.61	1772.87	800.09	602.5	1013.58	1487	998	1710	1445	1555.31	587.42
	SR	100	100	100	100	100	100	100	100	100	100	100
f_{34}	SD	2.88E-06	2.97E-06	3.07E-06	2.95E-06	2.64E-06	3.12E-06	1.62E-05	2.88E-06	2.93E-06	6.57E-06	3.10E-06
	ME	1.95E-03	1.94E-03	1.95E-03	1.94E-03	1.95E-03	1.95E-03	1.95E-03	1.95E-03	1.95E-03	6.40E-06	1.95E-03
	AFE	21458.41	29064.93	4761.02	5094.92	17641.71	3262	3927	3341.5	3092	36397.37	3418.07
	SR	100	100	100	100	100	100	99	100	100	82	100
f_{35}	SD	5.62E-06	5.58E-06	5.78E-06	5.76E-06	5.76E-06	2.49E-03	5.31E-06	5.62E-06	1.37E-03	6.87E-06	5.83E-06
	ME	5.16E-06	4.93E-06	5.10E-06	5.16E-06	5.11E-06	7.10E-04	4.62E-06	5.22E-06	3.12E-04	8.73E-05	5.16E-06
	AFE	3855.44	9968.97	7917.47	2468.49	9396.07	46715	8414.5	9121	90199	807.76	1619.34
	SR	100	100	100	100	100	67	100	100	100	100	100
f_{36}	SD	3.08E-03	1.75E-03	1.82E-03	2.42E-03	1.90E-03	3.47E-01	2.36E-01	8.27E-03	2.94E-01	2.90E-06	1.67E-03
	ME	5.18E-03	7.71E-03	7.78E-03	7.53E-03	7.91E-03	7.13E-01	5.57E-01	1.18E-02	4.39E-01	1.95E-03	8.35E-03
	AFE	25081.83	62307.88	42023.26	49473.76	63543.15	96757	198935	145212.5	181097.5	9907.84	22030.31
	SR	100	100	100	99	100	9	2	68	63	100	100
f_{37}	SD	2.72E-16	1.28E-16	5.52E-11	1.20E-16	2.40E-16	2.16E-14	8.74E-17	8.86E-17	6.12E-17	5.27E-06	1.09E-16
	ME	3.04E-16	8.31E-16	1.84E-11	8.43E-16	7.12E-16	1.42E-14	9.01E-16	9.10E-16	9.29E-16	4.50E-06	8.75E-16
	AFE	1672.4	85143.78	200024.42	38559.5	71183.5	100022	59418	40955	104872.5	5404.81	44903
	SR	100	100	0	100	100	4	100	100	100	100	100

Table 3: Comparison based on Acceleration Rate (AR)

TP:Test Problem

TP	GHABC Vs ABC	GHABC Vs BHABC	GHABC Vs GABC	GHABC Vs BS-FABC	GHABC Vs PSO-2011	GHABC Vs DE	GHABC Vs GbestDE	GHABC Vs SMO	GHABC Vs MeABC	GHABC Vs LFABC
f_1	21.740	34.726	22.337	46.804	59.699	34.942	23.844	59.319	30.608	26.052
f_2	11.689	18.038	17.417	50.922	67.362	43.312	26.304	67.682	12.692	19.842
f_3	68.888	45.631	31.537	65.172	76.240	66.311	30.640	117.534	44.786	42.169
f_4	47.128	52.316	41.025	144.699	117.931	235.802	223.831	235.802	68.000	47.909
f_5	52.311	85.310	25.590	60.622	64.646	35.768	24.222	64.796	54.015	30.144
f_6	73.378	57.285	53.062	138.105	87.429	59.195	50.016	90.318	101.422	82.739
f_7	0.470	0.657	0.466	1.053	2.308	3.891	1.056	4.606	0.504	1.010
f_8	21.356	54.841	24.157	50.191	77.927	47.528	24.226	98.762	36.917	27.983
f_9	14.090	34.743	23.366	36.754	55.454	33.486	23.150	55.543	19.652	27.952
f_{10}	70.895	70.886	70.886	70.886	35.461	24.156	60.792	69.622	35.710	70.900
f_{11}	37.809	53.850	20.247	54.508	60.577	34.852	23.832	60.738	41.807	21.568
f_{12}	23.378	37.424	22.189	49.195	55.249	34.686	23.699	55.250	32.525	25.398
f_{13}	39.980	102.206	23.040	44.117	58.988	37.453	22.226	58.899	45.027	25.787
f_{14}	401.336	272.135	246.788	537.053	288.251	169.611	301.281	576.629	66.315	292.429
f_{15}	26.284	34.427	21.556	49.658	59.161	35.044	23.898	60.066	34.758	24.491
f_{16}	55.039	19.773	26.140	40.171	278.450	21.696	15.875	27.544	556.690	20.939
f_{17}	14.941	19.030	18.862	77.945	20.623	71.325	22.063	73.861	15.258	22.892
f_{18}	172.893	106.224	72.286	186.657	151.655	266.947	71.295	301.143	303.175	64.334
f_{19}	1.644	0.907	1.617	1.642	0.819	0.142	1.362	0.554	0.403	0.326
f_{20}	32.730	30.477	20.544	52.133	69.588	34.731	23.590	59.645	38.936	22.353
f_{21}	0.476	0.559	0.604	1.222	2.605	0.917	0.661	1.737	1.688	0.702
f_{22}	0.713	0.724	0.512	1.262	1.422	0.838	0.549	1.365	0.588	0.575
f_{23}	2.290	0.489	0.586	3.304	2.967	0.095	0.300	0.185	1.971	0.252
f_{24}	1.443	0.455	1.149	1.109	0.020	0.166	0.759	0.352	0.039	0.470
f_{25}	27.899	9.556	15.194	23.018	0.358	9.751	23.517	5.476	6.309	9.373
f_{26}	4.216	2.574	1.953	1.390	8.455	1.847	9.324	6.658	21.541	4.004
f_{27}	1.276	0.872	0.762	1.389	0.212	1.351	1.320	1.358	0.069	0.484
f_{28}	0.934	0.925	0.584	1.897	10.305	1.085	0.811	1.653	15.553	0.649
f_{29}	1.295	1.355	0.630	1.761	1.487	2.282	1.011	2.935	2.973	0.600
f_{30}	1.486	4.515	0.592	1.991	6.357	0.982	0.746	1.565	2.262	0.695
f_{31}	16.802	7.668	4.350	0.416	0.870	0.428	1.004	0.872	7.532	1.251
f_{32}	3.554	1.069	0.876	3.164	5.611	2.434	2.881	5.705	135.992	0.777
f_{33}	1.922	0.867	0.653	1.099	1.612	1.082	1.853	1.566	1.686	0.637
f_{34}	1.354	0.222	0.237	0.822	0.152	0.183	0.156	0.144	1.696	0.159
f_{35}	2.586	2.054	0.640	2.437	12.117	2.183	2.366	23.395	0.210	0.420
f_{36}	2.484	1.675	1.972	2.533	3.858	7.931	5.790	7.220	0.395	0.878
f_{37}	50.911	119.603	23.056	42.564	59.807	35.529	24.489	62.708	3.232	26.849

Table 4: Comparison based on Mann-Whitney U rank sum test at significant level $\alpha = 0.05$ and average number of function evaluations

TP: Test Problem

TP	**GHABC Vs ABC**	**GHABC Vs BHABC**	**GHABC Vs GABC**	**GHABC Vs BS-FABC**	**GHABC Vs PSO-2011**	**GHABC Vs DE**	**GHABC Vs GbestDE**	**GHABC Vs SMO**	**GHABC Vs MeABC**	**GHABC Vs LFABC**
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}	+	+	+	+	+	+	+	+	+	+
f_{16}	+	+	+	+	+	+	+	+	+	+
f_{17}	+	+	+	+	+	+	+	+	+	+
f_{18}	+	+	+	+	+	+	+	+	+	+
f_{19}	+	-	+	+	-	-	+	-	-	-
f_{20}	+	+	+	+	+	+	+	+	+	+
f_{21}	-	-	-	+	+	-	-	+	+	-
f_{22}	-	-	-	+	+	-	-	+	-	-
f_{23}	+	-	-	+	+	-	-	-	+	-
f_{24}	+	-	+	+	-	-	-	-	-	-
f_{25}	+	+	+	+	-	+	+	+	+	+
f_{26}	+	+	+	+	+	+	+	+	+	+
f_{27}	+	-	-	+	-	+	+	+	-	-
f_{28}	-	-	-	+	+	+	-	+	+	-
f_{29}	+	+	-	+	+	+	+	+	+	-
f_{30}	+	+	-	+	+	-	-	+	+	-
f_{31}	+	+	+	-	-	-	+	-	+	+
f_{32}	+	+	-	+	+	+	+	+	+	-
f_{33}	+	-	-	+	+	+	+	+	+	-
f_{34}	+	-	-	-	-	-	-	-	+	-
f_{35}	+	+	-	+	+	+	+	+	-	-
f_{36}	+	+	+	+	+	+	+	+	-	-
f_{37}	+	+	+	+	+	+	+	+	+	+
Total no. of + signs	33	27	25	35	31	29	30	32	30	23

The boxplots examination has likewise been performed for a correlation with respect to solidified performance of all considered algorithms. The boxplots investigation represents the graphical distribution of empirical data in an efficient manner. The boxplots for GHABC and other considered algorithms are depicted in Fig. 4. It is clear from the Fig. 4, that GHABC performs better than other considered algorithms as median and interquartile range is quite low.

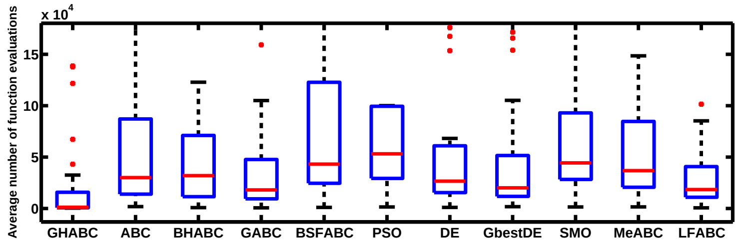

Figure 4: Boxplots graphs for average number of function evaluations

Further, comparison is also performed in terms of success performance (SP) (Qu et al., 2014). The SP is defined by the following Eq. 8 :

$$SP = \frac{AFE_{ALGO}}{SR_{ALGO}} \tag{8}$$

Here, AFE_{ALGO} is the AFEs and SR_{ALGO} is SR for the considered algorithm. An algorithm that is consuming less number of function evaluations and yielding higher SR is considered better. Hence smaller values of SP are desirable. The value of SP is not defined when SR is 0. The SP for proposed GHABC and other considered algorithms are calculated and the obtained values are listed in the Table 5. The results show that SP of GHABC is better than the other considered algorithms. In Table 5, SP is not defined for the functions, f_4, f_{10}, f_{16}, f_{18}, f_{19}, f_{24}, f_{29}, f_{30}, and f_{37} as the value of SR is equal to 0 in any of the algorithm.

Table 5: Comparison based on Success Performance (SP)

TP:Test Problem

TP	**GHABC**	**ABC**	**BHABC**	**GABC**	**BSFABC**	**PSO-2011**	**DE**	**GbestDE**	**SMO**	**MeABC**	**LFABC**
f_1	6.423	139.638	223.049	143.475	300.630	383.460	224.440	153.155	381.015	196.599	167.339
f_2	4.816	56.294	86.871	83.880	245.245	324.420	208.595	126.685	325.965	61.128	95.561
f_3	9.657	692.971	454.289	304.553	629.361	1168.643	790.574	295.890	1351.220	432.497	407.225
f_5	11.938	624.471	1018.401	305.490	723.685	771.725	426.990	289.160	773.520	644.818	359.850
f_6	10.302	755.945	590.160	546.655	1482.052	1250.972	609.830	515.270	949.454	1044.858	869.780
f_7	430.602	202.228	282.832	200.488	453.475	49701.250	7284.174	494.391	1983.260	216.818	494.279
f_8	6.383	136.321	350.070	154.205	320.390	585.224	316.031	154.645	716.403	235.656	178.629
f_9	5.082	71.607	176.569	118.750	186.785	281.825	170.180	117.650	282.275	99.872	142.054
f_{11}	11.381	430.299	612.863	230.430	620.345	689.425	396.645	271.235	691.255	475.798	245.468
f_{12}	6.344	148.303	237.400	140.760	312.075	350.480	220.035	150.340	350.485	206.328	161.113
f_{13}	12.020	480.549	1228.489	276.930	530.275	709.015	450.170	267.150	707.945	541.209	309.947
f_{14}	3.469	2442.730	973.316	901.243	2552.324	100003.000	594.374	1077.562	15388.462	230.065	1166.120
f_{15}	7.388	194.174	254.335	159.250	366.855	437.060	258.890	176.550	443.745	256.778	180.931
f_{17}	4.745	70.899	90.305	89.510	369.880	97.865	371.940	104.700	350.500	72.407	108.632
f_{20}	9.481	310.298	288.941	194.770	494.250	659.735	329.270	223.650	565.470	369.140	211.922
f_{21}	217.391	103.553	121.630	131.285	265.705	566.265	199.410	143.730	377.645	367.027	152.636
f_{22}	324.662	231.565	235.048	166.255	409.835	549.619	272.090	178.315	443.060	190.940	186.536
f_{23}	148.496	340.024	72.596	87.014	533.308	440.605	14.130	44.545	27.535	292.622	37.461
f_{25}	65.492	8700.628	625.842	1081.630	3350.046	23.475	912.286	3850.525	358.650	516.458	646.171
f_{26}	44.674	188.364	114.974	87.261	62.713	377.710	84.179	429.423	297.455	1118.994	190.274
f_{27}	1914.017	7329.597	1501.628	1116.830	17405.929	352.386	23265.625	12990.929	4456.250	94.717	701.399
f_{28}	95.521	89.223	88.389	55.775	181.170	32810.167	103.645	77.510	157.855	3809.237	62.033
f_{31}	112.403	14527.849	861.951	488.957	46.771	97.735	48.150	112.890	97.965	1032.419	140.656
f_{32}	8.852	31.459	9.462	7.750	28.007	49.665	21.545	25.505	50.500	3009.479	6.878
f_{33}	9.226	17.729	8.001	6.025	10.136	14.870	9.980	17.100	14.450	15.553	5.874
f_{34}	214.584	290.649	47.610	50.949	176.417	32.620	39.667	33.415	30.920	443.870	34.181
f_{35}	38.554	99.690	79.175	24.685	93.961	697.239	84.145	91.210	901.990	8.078	16.193
f_{36}	250.818	623.079	420.233	499.735	635.432	10750.778	99467.500	2135.478	2874.563	99.078	220.303

6. Conclusion and future works

This article proposes a local search technique based upon the grasshopper jumping mechanism, namely grasshopper local search (GHLS). The proposed GHLS strategy is incorporated into artificial bee colony (ABC) algorithm to improve the exploitation capability and convergence speed of the algorithm. Thus modified strategy is named as grasshopper inspired ABC (GHABC). To validate the performance, the proposed GHABC has been assessed by standard benchmark test problems and compared with other state-of-art algorithms. The numerical experiments and analyses depict the validity of the proposed approach. It can be concluded that GHABC algorithm is a good choice to find solution of numerical optimization problems.

References

Akay, B., & Karaboga, D. (2012). A modified artificial bee colony algorithm for real-parameter optimization. *Information Sciences*, *192*, 120–142.

Banharnsakun, A., Achalakul, T., & Sirinaovakul, B. (2011). The best-so-far selection in artificial bee colony algorithm. *Applied Soft Computing*, *11*(2), 2888–2901.

Bansal, J. C., Sharma, H., Arya, K., & Nagar, A. (2013). Memetic search in artificial bee colony algorithm. *Soft Computing*, *17*(10), 1911–1928.

Bansal, J. C., Sharma, H., & Jadon, S. S. (2013). Artificial bee colony algorithm: a survey. *International Journal of Advanced Intelligence Paradigms*, *5*(1), 123–159.

Bansal, J. C., Sharma, H., Jadon, S. S., & Clerc, M. (2014). Spider monkey optimization algorithm for numerical optimization. *Memetic computing*, *6*(1), 31–47.

Bennet-Clark, H. (1975). The energetics of the jump of the locust schistocerca gregaria. *Journal of Experimental Biology*, *63*(1), 53–83.

Clerc, M., & Kennedy, J. (2011). Standard pso 2011. *Particle Swarm Central Site [online] http://www. particleswarm. info*.

Hall, A. R. (1996). *Isaac newton: adventurer in thought* (Vol. 4). Cambridge University Press.

Heitler, W. (1974). locust jump. specialisations of the metathoracic femoral-tibial joint. *Journal of comparative physiology*.

Karaboga, D. (2005). *An idea based on honey bee swarm for numerical optimization* (Tech. Rep.). Technical report-tr06, Erciyes university, engineering faculty, computer engineering department.

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

Karaboga, D., & Akay, B. (2011). A modified artificial bee colony (abc) algorithm for constrained optimization problems. *Applied Soft Computing*, *11*(3), 3021–3031.

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

Karaboga, D., Gorkemli, B., Ozturk, C., & Karaboga, N. (2014). A comprehensive survey: artificial bee colony (abc) algorithm and applications. *Artificial Intelligence Review*, *42*(1), 21–57.

Kennedy, J. (2011). Particle swarm optimization. In *Encyclopedia of machine learning* (pp. 760–766). Springer.

Mokan, M., Sharma, K., Sharma, H., & Verma, C. (2014). Gbest guided differential evolution. In *Industrial and information systems (iciis), 2014 9th international conference on* (pp. 1–6).

Offenbacher, E. L. (1970). Physics and the vertical jump. *American Journal of Physics*, *38*(7), 829–836.

Qu, B., Liang, J., Suganthan, P., & Chen, Q. (2014). Problem definitions and evaluation criteria for the cec 2015 competition on single objective multi-niche optimization. *Computational Intelligence Laboratory, Zhengzhou University. Zhengzhou, China: Tech. Rep.*

Sharma, A., Sharma, H., Bhargava, A., & Sharma, N. (2016a). Optimal power flow analysis using lvy flight spider monkey optimisation algorithm. *International Journal of Artificial Intelligence and Soft Computing*, *5*(4), 320-352. doi:

Sharma, A., Sharma, H., Bhargava, A., & Sharma, N. (2016b). Power law-based local search in spider monkey optimisation for lower order system modelling. *International Journal of Systems Science*, 1–11.

Sharma, H., Bansal, J. C., Arya, K., & Yang, X.-S. (2016). Lévy flight artificial bee colony algorithm. *International Journal of Systems Science*, *47*(11), 2652–2670.

Sharma, N., Sharma, H., Sharma, A., & Bansal, J. C. (2015). Black hole artificial bee colony algorithm. In *International conference on swarm, evolutionary, and memetic computing* (pp. 214–221).

Storn, R., & Price, K. (1997). Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces. *Journal of global optimization*, *11*(4), 341–359.

Suganthan, P. N., Hansen, N., Liang, J. J., Deb, K., Chen, Y.-P., Auger, A., & Tiwari, S. (2005). Problem definitions and evaluation criteria for the cec 2005 special session on real-parameter optimization. *KanGAL report*, *2005005*, 2005.

Yang, X.-S. (2014). *Nature-inspired optimization algorithms*. Elsevier.

Zhu, G., & Kwong, S. (2010). Gbest-guided artificial bee colony algorithm for numerical function optimization. *Applied mathematics and computation*, *217*(7), 3166–3173.