Alternated Superior Chaotic Biogeography-Based Algorithm for Optimization Problems

Deepak Kumar, Central University of Rajasthan, India*

Mamta Rani, Central University of Rajasthan, India

ABSTRACT

In this study, the authors consider a switching strategy that yields a stable desirable dynamic behaviour when it is applied alternatively between two undesirable dynamical systems. Over the last few years, dynamical systems employed “chaos₁ + chaos₂ = order” and “order₁ + order₂ = chaos” (vice-versa) to control and anti control of chaotic situations respectively. To find parameter values for these kinds of alternating situations, comparison is being made between bifurcation diagrams of a map and its alternate version, which, on their own, means independent of one another, yield chaotic orbits. However, the parameter values yield a stable periodic orbit, when alternating strategy is employed upon them. It is interesting to note that we look for stabilization of chaotic trajectories in nonlinear dynamics, with the assumption that such chaotic behaviour is not desirable for a particular situation. The method described in this paper is based on the Parrondo’s paradox, where two losing games can be alternated, yielding a winning game, in a superior orbit.

KEYWORDS

Alternated Logistic Map, ASCBBO, BBO, CBBO, SCBBO, Superior Iterations

1. INTRODUCTION

Optimization problems deal with the situations in which it is mandatory to search for the most appropriate solution among all the available solutions of a particular problem in a reasonable amount of time. These large scale optimization problems often suffer from the problems of multi-modality, non-continuous, dimensionality, non-convex and so-on. So, to tackle these real world complicated problems, efficient optimization algorithms are urgently required. Therefore, various evolutionary techniques have been developed and applied in recent years which include Genetic Algorithms (GAs), Particle Swarm Optimization algorithm (PSO), Differential Evolution algorithm (DE), Ant Colony Optimization (ACO), Artificial Bee Colony Strategy (ABC), and BBO (Simon, 2008).

From the last decade or so, the recent advances in theories and applications of nonlinear dynamics especially chaotic maps have drawn much attention in many fields of optimization in replacing certain algorithm dependent parameters (Jalili, Hosseinzadeh & Kaveh, 2014; Li-Jiang & Tian-Lun, 2002; Talatahari, Azar, Sheikholeslami & Gandomi, 2012). Chaotic maps have shown increase in population diversity and high level of mixing capability. Therefore, replacing a fixed parameter with
the chaotic map may provide solutions with higher mobility and greater diversity. Many chaotic maps have been used by these meta-heuristic algorithms to improve upon the results of these algorithms through proper balance between exploration and exploitation activities (Li-Jiang & Tian-Lun, 2002; Talatihari, Azar, Sheikholeslami & Gandomi, 2012; Yang, Li & Cheng, 2007).

Mingjun & Huanwen (2004) presented a novel algorithm by replacing the Gaussian distribution of simulated annealing with chaotic initialization and chaotic sequences. The proposed algorithm has been validated on typical complex function optimization problems. Alatas, Akin & Ozer, (2009) have presented twelve chaos-embedded PSO methods with the use of eight chaotic maps and analysed them on the benchmark functions. The simulation results demonstrated the robustness of the proposed methods with increased solution quality, i.e., in some cases they improved the global searching capability by escaping the local solutions. Alatas (2010a) presented two new ABC algorithms in combination with seven chaotic maps for parameter adaptation for improved convergence characteristics and to prevent the ABC from plunging into local solutions.

Alatas (2010b) presented seven new harmony search algorithms which employ chaotic maps for better convergence characteristics. In this research work, chaotic number generators are employed whenever there is a need for it by the classical harmony search algorithm. It has been demonstrated that results obtained from these coupling of various areas, like those of harmony search and complex dynamics, can significantly improve the quality of results in some optimization problems. Gharooni-fard et al. (2010) introduced a novel chaos based genetic based algorithm. The proposed approach, when applied to both balanced and unbalanced workflow structures, have validated its usage. Basically, the proposed approach scatters the solutions among the whole search space by employing the positive characteristics of the chaotic variables which together with avoiding premature convergence of the solutions also generates superior results within a shorter time.

Talatahari et al. (2012) proposed improved imperialist competitive algorithm using chaotic maps. Particularly, the random coefficient vector has been replaced by different chaotic systems and the random parameter in the orthogonal vector. Logistic and Sinusoidal maps performed better than the other chaotic maps used in this study. Gandomi et al. (2013) presented an upgraded variant of firefly algorithm by embedding 12 chaotic maps to tune the attractiveness and absorption coefficients. The proposed algorithm applied on the global optimization problems clearly demonstrated that some chaotic maps have phenomenally outperformed the results of the original firefly algorithm. Alatas (2013) proposed to solve the economic load dispatch problem with the application of chaotic firefly algorithm. In this paper, chaotic tent map was used to enhance two key parameters of firefly algorithm, i.e., randomization and attractiveness. The proposed algorithm demonstrated good convergence attributes on all considered economic load dispatch test cases in comparison to all the other soft computing techniques employed in the paper.

Fister et al. (2014) presented a randomized firefly algorithm in collaboration with different probability distributions and chaotic maps. The experimental results showed improved performance of the randomized firefly algorithm when used with probability distributions (e.g., uniform, Gaussian and l’evy flights) and chaotic maps (logistic and tent). Wang et al. (2014) presented a hybridized version of chaos theory with Krill Herd algorithm for solving optimization problems. Different chaotic maps were utilized to regulate the key parameter of Krill Herd algorithm. The experimental results on different chaotic Krill Herd variants established the superior performance of the singer map in forming the best chaotic Krill Herd. Taking clue from success of these metaheuristic algorithms, another popular algorithm in the series of nature inspired algorithms, BBO was introduced.

BBO was an evolutionary optimization algorithm which was given by American scientist, Dan Simon in 2008 (Simon, 2008). He invented a new population-based search technique which was inspired from the theory of island biogeography known as Biogeography-Based Optimization algorithm. The main characteristics of BBO algorithm are migration, speciation and extinction of species in a given geographical location. It is comparable to other evolutionary algorithms in solving complex optimization problems and afterwards, a lot of improvements have been given in
the literature especially when chaos was incorporated in it (Lesmoir-Gordon & Rood, 2014; Saremi & Mirjalili, 2013).

So, chaotic migration and chaotic mutation operators are used to increase the population diversity to avoid entrapment of the candidate solutions in local optima (Liu et al., 2005). Saremi and Mirjalili (2013) used three chaotic maps with four benchmark functions to improve the weaknesses of the BBO algorithm. The integration of chaotic maps with the BBO algorithm is another method in improving the results of the BBO algorithm. Sine map have successfully improved the results out of all the other chaotic maps. Afterwards, they used ten chaotic maps with ten test functions in further expansion of their work (Saremi, Mirjalili & Lewis, 2014). They used this technique of integration in five different ways. Selection, migration and mutation operators are defined with chaotic maps at first, then combination of selection and migration and at the end they employed combination of selection migration and mutation strategies. Zhu, Luo and Zhu (2014) proposed improved genetic algorithm with four local search operators which are inspired from Dijkstra’s algorithm and carried out when the topology changes to generate local shortest path trees which in turn are used to promote the performance of the individual in the population for dynamic shortest path problems. The experimental results obtained when applied on CEC 2014 test suite adapt rapidly to new environments and produce high quality solutions after environmental adjustments. Later on, Guo-ping et al. (2016) used chaotic maps with BBO algorithm in finding parameters of discrete chaotic systems with minimal time series data and control chaos using constant feedback method. Giri et al. (2017) used chaotic maps in improving local and global parameters and have shown increased convergence over the non-chaotic approach.

Jalili et al. (2014) used chaotic migration and chaotic mutation operators to solve the problem of truss structures with natural frequency constraints which are nonlinear dynamical optimization problem with several local optima. Later, Heidari, Mirvahabi & Homayouni (2015) used this technique in predicting earthquake-originated slope displacements (EIDS). They used chaotic BBO in combination with SVR (Support Vector Regression) to investigate the best possible values of SVR parameters. Wang et al. (2016) used this combination of chaos with BBO in centroid based clustering methods. They used three types of simulation data in proving the superiority of their approach. Wang and Song (2017) used chaotic mapping strategy in combination with BBO optimal migration model which is close to the natural law in achieving overall increased convergence velocity and higher optimization precision accuracy. To know more about the BBO algorithm, its modifications and its combination with other meta-heuristic algorithms, one may go for a comprehensive survey from the last ten years prepared by Ma et al. (2017). Some of the other researchers who have contributed to the chaos based metaheuristic algorithms have been given in Table 1.

A number of different chaotic maps have been used in the BBO algorithm in the previous researches (Saremi & Mirjalili, 2013; Saremi, Mirjalili & Lewis, 2014). In this paper, we propose to use the chaotic sequence that is generated by alternating two ordered logistic maps together in the superior orbit and then evaluate their performance to ascertain how they behave in increased solution space. The paper has been structured as follows. Section 2 explains the motivation and thought behind this paper. Section 3 discusses the preliminaries of the BBO algorithm, Parrondo’s paradox and its applications in superior orbit. Section 4 describes about the proposed approach of combining alternating strategy in superior orbit with chaotic BBO and also the theoretical time complexity of the proposed approaches. Section 5 gives the simulation experiments and results analysis followed by concluding remarks which are discussed in Section 6.

2. MOTIVATION

The quality of random sequences generated remarkably affects the global optimal solutions of metaheuristic algorithms. Previous studies have shown that these random sequences with an advanced amount and uniform structure are vital to achieving the globally optimal results with enhanced
| Name                      | Method/Algorithm                                                                 | Chaotic maps and Benchmarks/ Datasets Used                                                                 | Parameters Validated                                                                 | Applications                        |
|---------------------------|---------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------|-------------------------------------|
| Yakubu, & Aboiyar (2018) | New confusion-diffusion cryptosystem which makes use of chaos rich Shimizu-Morioku system to shuffle the image | Shimizu-Morioka system and a standard test digital colour image of size 256x256, stored with TIFF file format (Lena_colour.tif) | Histogram uniformity analysis, the correlation coefficient analysis, and the number of pixel changing intensity | Image segmentation problems          |
| Linglong, Yehui, & Changkai (2018) | Conventional fuzzy clustering algorithms with global search capability of the PSO swarm algorithm along with chaotic sequences | Logistic map and three test images in tiff format are used for image segmentation | Time consumption and classification accuracy | Image classification problems          |
| Wang et al. (2018)       | Chaotic starling PSO which is inspired from the collective responses of the starling birds | Logistic map and benchmark functions are Sphere, Griewank, Rastrigin, and Rosenbrock and datasets are Data_3_2, Data_5_2, Data_10_2, Data_4_2, Iris, Wine, Glass, and CMC. | Robustness and effectiveness | Optimization problems          |
| Bejinariu et al. (2019)  | PSO, multi swarm optimization (MSO), cuckoo search algorithm (CSA) and black hole algorithm (BHA) are combined with nine chaotic maps | Chebyshev, Circle, Gauss, Iterative, Piecewise, Sine, Singer, Sinusoidal and Tent maps and Medical dataset | Precision of clusters | Clustering problems |
| Gálvez, Cuervas, Becerra & Avalos (2019) | Cluster chaotic optimization | ICMC map and 30 benchmark functions and 4 engineering design optimization problems | Robustness and accuracy | Optimization problems          |
| Lu et al. (2019)         | Dynamic swarm firefly algorithm in combination with chaos theory and max-min distance algorithm | Tent map and Iris, Wine, Seed, Glass, Hayes-Roth, and New-Thyroid. | Fast convergence, accuracy of clustering results and avoidance of local solutions | Optimization problems          |
| Arslan & Toz (2019)      | Whale optimization algorithm along with chaotic maps using an adaptive normalization method and fuzzy c-means clustering algorithm | Chebyshev, Circle, Gauss, Iterative, Logistic, Piecewise, Sine, Singer, Sinusoidal, and Tent maps and 13 benchmark functions and Iris, Balance Scale, User Modeling, Breast Cancer, Seeds, and Fertility database | Clustering performance | Optimization problems          |
| Bouyer & Farajzadeh (2015)| A hybrid of k-harmonic means clustering algorithm and a modified version of PSO algorithm along with Cuckoo Search Levy Flight algorithm | ArtSet1, ArtSet2, Iris, Wine, Wisconsin breast cancer, Ripley’s glass, CMC, Thyroid gland, Vowel, Ecoli | Convergence rate, efficiency and local optima entrapment | Clustering optimization problems |
| Dhanusha, & Kumar (2021) | Unsupervised nature inspired crow search learning model | Logistic map and “CASAS” and “OASIS” datasets | Efficiency of the proposed algorithm in handling the noisy data and indeterminacy behaviour of the dataset | Alzheimer disease detection          |
| Zhu, Liu, & Wang (2020)  | Chaotic crow search algorithm and improved fuzzy c-means clustering algorithm | Chebyshev map and Synthetic and non-destructing images | Cluster density and noise reduction | Image segmentation          |
| Kaur, Pal & Singh (2020) | Flower pollination algorithm with chaos | Logistic map, Sine map, Dyadic map, Chebyshev map, and Circle map and Iris, Wine, Breast_Cancer, Glass, Balance, Dermatology, Haberman, Ecoli, Heart, Tae, Spambase, ILPQ, Leaf, Libras, Qualitative_Bankruptcy, Synthetic are the datasets employed. | Execution time, stability and convergence speed | Optimization clustering problems |
| Singh (2020)             | Harris hawk optimization algorithm in relation with chaotic sequences | Logistic map and Shape datasets are Aggregation, Compound Path based, Spiral, Flame, Jain, R15, D31 and UCI datasets are Glass, Iris, Wine, Yeast | Accuracy of cluster indices | Clustering applications          |

*continued on next page*
accuracy. From a few years, it has been seen that the use of chaotic sequences instead of random numbers have been quite instrumental in substantially increasing the performance of the various metaheuristic algorithms. This suggests that the use of chaos in metaheuristic algorithms is an area of great interest among many researchers from disciplines of varying fields (Pecora & Carroll, 1990; dos Santos Coelho & Mariani, 2008; Strogatz, 2018; Jin, Lin & Zhang, 2021).

In the context of ecological modelling, from the last decade or so, the idea of switching strategies has been reconsidered in the form of Parrondo’s paradox, where two losing games when combined together in a deterministic or random order give a winning game (Danca, Fečkan & Romera, 2014). Logistic map, over the years have served as a medium for development and understanding of nonlinear dynamics. Therefore, the idea of using alternate discrete dynamics on logistic map may yield favourable situations. One of the situations may be when two ordered logistic maps are alternated together, it may result in chaotic behaviour which is an ideal condition for achieving higher rates of optimization values in optimization problems, i. e., “order\(_1\) + order\(_2\) = chaos”. For more details, one may refer to (Danca, Fečkan & Romera, 2014; Danca & Tang, 2016; Levinsohn, Mendoza & Peacock-López, 2012; Maier & Peacock-López, 2010; Peacock-López, 2011). Later on, Rani and Yadav (Rani & Yadav, 2016; Yadav & Rani, 2015) extended this idea in studying logistic map and its variants in superior orbit and given examples of “chaos\(_1\) + chaos\(_2\) = order” and “order\(_1\) + order\(_2\) = chaos” (vice-versa).

3. PRELIMINARIES

3.1. Biogeography Based Optimization

The term BBO suggests that it has its roots in the biogeography discipline which concerns with the relationships between different species (habitants) living in ecologically distributed habitats in a given ecosystem. The evolvement of the species happens in terms of immigration, emigration and mutation activities. The drifting of the species to neighbouring habitats takes place through various means like air, water, and many other different pathways (Wallace, 2005).

Additionally, in relation to movement of species between different habitats, each habitat is having its own set of survival indicators, which is called as Habitat Suitability Index (HSI). In fact, BBO being an optimization algorithm in which a habitat is taken as a possible candidate solution. The variables that characterize habitats are called Suitability Index Variables (SIVs). The fitness value of each habitat is calculated using HSI (MacArthur & Wilson, 2016).
High HSI habitats tend to have higher number of species, while lower HSI habitats attract smaller number of species. The HSI of poor habitats can be improved with new features derived from more attractive habitats in the evolution process (Simon, 2011). In this approach, the information between poor and good habitats is shared through the migration operator. This migration operator is responsible for emigration and immigration. This adaptive information sharing between habitats happens due to immigration rate $\lambda$ and emigration rate $\mu$ of each habitat which in turn are functions of the number of species present in the habitat. These can be calculated according to the following equations as (Du, Simon & Ergezer, 2009; Giri et al., 2017):

$$
\lambda_k = I \left(1 - \frac{K}{S_{\text{max}}} \right)
$$

$$
\mu_k = E \left(\frac{K}{S_{\text{max}}} \right)
$$

where $I$ is representing the maximum possible immigration rate; $E$ is showing the maximum possible emigration rate; the number of species in the $k^{th}$ habitat is represented by $K$ and the maximum number of species supported by the habitat is represented by $S_{\text{max}}$ (Guo-ping et al., 2016; Heidari, Mirvahabi & Homayouni, 2015). The pseudo code of the BBO algorithm is following.

**Algorithm 1.** Pseudo code of BBO algorithm (Saremi, Mirjalili & Lewis, 2014)

**Step 1.** Initialize the parameters of BBO algorithm.
**Step 2.** While the termination condition is not satisfied.
**Step 3.** Initialize the Fitness Function for the Habitats.
**Step 4.** Calculate the Habitat Fitness Index (HSI) and sort them.
**Step 5.** Update the $S$, $\lambda$ and $\mu$ of each habitat.
**Step 6.** For i=1 to maximum number of habitants do
   If rand < $\lambda_i$ then
      For j=1 to maximum number of habitants do
         If rand < $\mu_j$
            Select a random habitant in $x_i$ and replace it with $x_j$
         End if
      End for
   End if
   If rand < mutation probability
      Mutate a random number of habitats
   End if
   Elitism
End while

Fig. 1 illustrates the linear migration model of the BBO algorithm. As it is clear from the figure that the emigration rate is zero when there are no species present in the habitat. The emigration rate attains the maximum value $E$ when the species reach to its maximum capacity $S_{\text{max}}$. In the similar manner, the immigration rate achieves the maximum possible value $I$ when the number of species
is zero. Also, when the number of species becomes $S_{\text{max}}$, then the immigration rate declines and becomes zero. $S_0$ is the equilibrium point, which is achieved when the emigration rate $\mu$ becomes equal to immigration rate $\lambda$ (Simon, 2008; Simon, 2011).

For each individual habitat $H_k$, the associated probabilistic rate which determines whether to immigrate or not is $\lambda_k$. Based on the emigration rate $\mu_j$, the emigrating solution $H_j$ is selected probabilistically when immigration is selected. During replacement of a copy of SIV $\sigma$ from individual habitat $H_j$ to $H_k$, it is said that $\sigma$ has immigrated to $H_k$ and emigrated from $H_j$; that is $H_k(\sigma) \leftarrow H_j(\sigma)$. Thus migration operator is able to efficiently enhance the global convergence of the algorithm by sharing information among individual habitats (Wallace, 2005).

After migration activity, mutation operator is used to further increase the diversity of the available population. It is a probabilistic operator which is used to modify the SIV of a randomly selected habitat which is habitat’s priori probability of existence and is computed as follows (Heidari, Mirvahabi & Homayouni, 2015; Jalili, Hosseinzadeh & Kaveh, 2014).
\[ m_k = m_{\text{max}} \left( 1 - \frac{P_k}{P_{\text{max}}} \right) \]  

(3)

where \( m_{\text{max}} \) represents a user-defined parameter and \( P_{\text{max}} = \max\{P_k\} \), \( k = 1, 2, 3, \ldots, N \) and \( P_k \) shows the probability that the habitat has exactly \( k \) number of species.

Another feature of BBO is that the habitats having higher HSI are kept as elites and moved from the previous generation to the next generation. It is meant therefore that the new habitats of current iteration are combined with some elites of the prior generation. After combination, the higher HSI are selected for creation of newer generation of population. In this study, the probability of mutation rate is set to 0.005 (MacArthur & Wilson, 2016; Simon, 2008; Simon, 2011).

### 3.2. Parrondo’s Paradox

It is a paradox in game theory given by famous physicist Juan Parrondo in the year 1996. As per this paradox, when two simple games with negative gains are played together alternatively may produce a game with positive gains in a different deterministic or random manner (Danca, Fečkan & Romera, 2014). In 2001, it was introduced to the combination of two unstable systems \( A_1 \) and \( A_2 \) to study the dramatic change in the properties of the systems when they are combined. Initially, the authors considered the traditional idea of Parrondo’s paradox of “\( \text{losing}_1 + \text{losing}_2 = \text{winning} \)” and applied it on different kinds of linear systems to show the “\( \text{instability}_1 + \text{instability}_2 = \text{stability} \)” (Danca & Tang, 2016; Mendoza et al., 2018).

Two different discrete dynamics \( A_1 \) and \( A_2 \) are considered and the alternation of combination of the dynamical systems \( A_1 \) and \( A_2 \) is discussed as follows:

\[
x_0^{A_0} x_1^{A_1} x_2^{A_2} x_3^{A_3}, 
\]

where \( H \) is describing a deterministic or random law which allocates a value of 1 or 2 to every member of the sequence \( \{0, 1, 2, \ldots\} \), and \( \{x_0, x_1, x_2, \ldots\} \) are values given to a variable \( x \) which is representing the physical system. The two individual dynamics \( A_1 \) and \( A_2 \) may be chaotic but when combined periodically in an alternated way \( A_1 A_2 A_1 A_2 A_1 A_2 \ldots = (A_1, A_2) \), may produce an ordered sequence and vice versa. The phenomenon thus created can be stated in terms of “\( \text{chaos}_1 + \text{chaos}_2 = \text{order} \)” and “\( \text{order}_1 + \text{order}_2 = \text{chaos} \)” (vice-versa) (Levinsohn, Mendoza & Peacock-López, 2012).

**Definition: Alternated System:** Let us consider two different dynamics \( A_1 \) and \( A_2 \), where \( A_1: x_{n+1} = x_n^2 + c_1 \), \( A_2: x_{n+1} = x_n^2 + c_2 \), and the alternation of combination of two dynamics \( A_1 \) and \( A_2 \) is defined as:

\[
x_{n+1} = \begin{cases} 
   x_n^2 + c_1, & \text{when } n \text{ is odd,} \\
   x_n^2 + c_2, & \text{when } n \text{ is even.} 
\end{cases} 
\]

(4)

where \( x, c, c_1, c_2 \) represent the real numbers. As it is a well-known fact that \( x_{n+1} = x_n^2 + c \) is topologically conjugate to the logistic map \( x_{n+1} = r x_n \left( 1 - x_n \right) \), \( x_n \in [0,1] \), it is derived that there may be a situation that shows “\( \text{chaos}_1 + \text{chaos}_2 = \text{order} \)” and also “\( \text{order}_1 + \text{order}_2 = \text{chaos} \)” which may arise in the logistic map. For more details on the literature on Parrondo’s paradox, one may refer to (Danca, Fečkan & Romera, 2014; Danca & Tang, 2016; Levinsohn, Mendoza & Peacock-López, 2012).
3.3. Superior Orbit (SO)

Feedback processes discovered by Isaac Newton and Gottfried W. Leibniz have found tremendous applications in nonlinear systems in the form of dynamical laws (Ashish, 2014). The feedback process is the process in which the output of the first iteration is given to the second iteration and the process is repeated until some given number of iterations. It is simple in principle as the same process is repeated again and again. Mostly, two types of feedback machines are used.

3.3.1. One-step Feedback Machine (Picard Orbit)

Definition: (Picard Orbit): Let \( A \) be a non-empty set of numbers and \( f : A \to A \). For a point, \( x_0 \) in \( A \), the Picard orbit (generally called orbit of \( f \)) is the set of all iterates of the point \( x_0 \), that is,

\[
O(f,x_0) = \{ x_n : x_n = f(x_{n-1}), n = 1,2,\ldots \}.
\]

(5)

The orbit \( O(f,x_0) \) of \( f \) at the initial point \( x_0 \) is the sequence \( \{f(x_0)\} \) (Ashish, 2014; Goel, 2011; Negi & Rani, 2008a and 2008b).

3.3.2. Two-step Feedback Machine (Superior Orbit)

Two-step feedback machine was introduced by Rani in fractal and chaotic models until recently (Negi & Rani, 2008). It became the reason for generation of superior fractals. In this approach, a new number is generated after the insertion of two numbers, and the formula for computing the new number is

\[
x_{n+1} = g(x_n, x_{n-1}).
\]

Definition: (Superior Iterates): Let \( A \) be a subset of real numbers and \( f : A \to A \). For \( x_0 \in A \), construct a sequence \( \{x_n\} \) in the following manner:

\[
x_1 = \beta_1 f(x_0) + (1-\beta_1) x_0,
\]

\[
x_2 = \beta_2 f(x_1) + (1-\beta_2) x_1
\]

\[
x_n = \beta_n f(x_{n-1}) + (1-\beta_n) x_{n-1}
\]

(6)

where \( 0 < \beta_n \leq 1 \) and \( \{\beta_n\} \) is convergent away from 0 (c.f. Negi & Rani, 2008a).

The subset \( A \) may also be taken as a subset of complex numbers without loss of generality as depicted in the above definition. The sequence constructed above \( \{x_n\} \) is a superior sequence of iterates or superior orbit, denoted as \( SO(f,x_0,\beta_n) \). Superior orbit at \( \beta = 1 \) reduces to \( O(f,x_0) \) (see above definition of Picard orbit). The definition is originally given by W. R. Mann (1953). M. A. Krasnosel’skii (1955) gave superior iterates for \( \beta_n = 0.5 \). Because of the superset of solutions that are generated as compared to Picard iterates by Mann iterations, Rani and Kumar renamed them as superior iterates (Ashish, 2014; Goel, 2011; Singh, Mishra & Sinkala, 2012).

A number of superior fractal structures have been developed by Rani along with other researchers for \( \beta_n = \beta, n = 1,2,\ldots \) for various values of \( \beta \) (Negi & Rani, 2008a; Negi & Rani, 2008b; Negi, Rani & Mahanti, 2008).
4. ALTERNATED SUPERIOR CHAOTIC BBO (ASCBBO) AND SUPERIOR CHAOTIC BBO (SCBBO)

The alternate superior chaotic mapping strategy is employed to realize the mutation operator in BBO algorithm and we call the proposed algorithm as alternate superior chaotic BBO, abbreviated as ASCBBO algorithm. The integration of alternated chaotic BBO in superior orbit with mutation can help in improving the detection capability (exploitation) by the increased solution set with these kinds of combinations. The ASCBBO is used with different standard test functions which are Sphere, Schwefel, Rosenbrock, Quartic, Penalty #1, Penalty #2, Griewank, Fletcher, Ackley and Rastrigin (Saremi, Mirjalili & Lewis, 2014). A habitat for migration is selected on probability which is defined by selection operator (λ) and after the selection of a habitat; emigration is performed with emigration probability (μ) as can be seen from Fig. 1. The chaotic sequence \( C(x) \) is generated when we iterate the logistic map \( f(x) = r \cdot x \cdot (1 - x) \) in an iterative manner by taking the values of \( r \) as 4.76 and 4.8034 in a superior orbit with \( \beta = 0.7 \) in odd and even iterations respectively. These values have been obtained from the work of Yadav (Yadav & Rani, 2015). The logistic map shows ordered (non-chaotic) behaviour when these values are iterated individually. However, when these values of ‘\( r \)’ are iterated alternatively, then the map produces chaotic oscillations, i.e., \( \text{order}_1 + \text{order}_2 = \text{chaos} \). In case of superior chaotic BBO (SCBBO), the value of \( r \) is taken as 4.1 and \( \beta = 0.9 \) in a superior orbit as given by Rani and Agarwal (2009). The mapping of chaotic sequences to mutation operators is described further as follows.

4.1. Chaotic Mapping of Mutation Operator

When the selection and emigration of the habitat is done, then the next task is to mutate the inhabitants so that the stagnation of the species is removed by altering certain parameters of the species to further diversify the population in order to get more areas of favourable solution space for enhancement of the required procedure. The chaotic values are used to describe this mutation probability as described below.

\[
\text{for } i = 1 \text{ to number of habitants at } k\text{-th habitat} \\
\text{if } C(x) < \text{Mutation_rate } (k) \text{ then} \\
\text{Mutate } i\text{-th habitant} \\
\text{end if} \\
\text{end for}
\]

Here, \( C(x) \) is representing the chaotic values of the map at \( i\text{th} \) iteration and \( \text{Mutation_rate } (k) \) is illustrating the mutation probability of \( k\text{th} \) habitat. The chaotic mapping gives the values for the emigration operators which are in the range [-1, 1] and then are normalised in the range of [0, 1] (Saremi, Mirjalili & Lewis, 2014). The investigation of BBO and ASCBBO under the influence of chaotic maps is explained in the following section. The initial value of ASCBBO is kept at 0.7 as in case of CBBO (Saremi, Mirjalili & Lewis, 2014) as it holds great significance in case of nonlinear chaotic situations.

The comparative study of all the graphs given in Fig. 2 clearly indicates that in case of superior logistic map, the chaotic data points are densely populated and more uniformly distributed which may help in finding more global optimal values as the solution space is increased for the candidate particles. In case of alternated superior chaotic logistic map, two chaotic maps are used in an alternate manner in the increased solution domain of finding a possible global optimal point which further increases the possibility of avoidance of the candidate solutions in local optimal points as the increased complexity in the alternated chaotic maps further reduces the biasness in input data.

4.2. Theoretical Time Complexity of BBO, CBBO, ASCBBO, and SCBBO

As compared with other metaheuristic algorithms, BBO works by sharing information among candidate solutions which makes it suitable for similar kind of solving problems that the other algorithms are
used for, such as when applied on high dimensional data. Hence, the computational cost of BBO and other similar algorithms will be the same as they heavily rely on the evaluation of the objective function. BBO uses tournament selection for the selection operator which usually demands $O(N)$ time complexity, where $N$ is the number of the habitats. For migration operation, a habitat having $D$ number of SIVs demands $O(ND + O(f))$ time complexity where $O(f)$ is the time complexity for computing the fitness function $f$. Therefore, each generation needs $O(N(ND + O(f)))$ time for its computation. Constant time is required when the neighbourhood HSI is selected. Hence, when $M$ number of iterations are used in an experiment, then the required time becomes $O(NM(ND + O(f)))$.

Following observations are made for the general time complexity of BBO algorithm.

1. $O(f)$ is considered much less as compared to $N^2$, then the overall time complexity of the BBO algorithm becomes $O(MN^2D)$.
2. When $D$ is insignificant compared to $N$, then the complexity of BBO takes the form as $O(MN^2)$.

In essence, whenever a habitant in a habitat immigrates, then the emigration vector is computed which depends upon the local best (local population) and global best (whole population), based on the rate of migration (Giri et al., 2017). Also in case of CBBO, SCBBO and ASCBBO, different chaotic sources are used for the selection, migration and mutation of the habitants which also require constant time as the case with the random. Hence, the time complexity of CBBO, SCBBO, and
ASCBBO is equivalent to BBO. Thus, for small number of SIVs, the overall time complexity of all the versions of BBO is $O(MN^2)$.

5. SIMULATION EXPERIMENTS AND RESULTS ANALYSIS

Ten standard benchmark functions and CEC 2014 test suite (in 50 dimensional space) have been used to test the performance of the ASCBBO and SCBBO algorithms. These test functions have been categorized into two groups namely unimodal and multimodal test functions. As the name specifies, unimodal functions have single optima in them which makes them best suited for the exploitation related activities. On the other side, multimodal functions have multiple optimal points which create challenges in finding the most appropriate optima in these kinds of test functions. Because one of them is a global optima and rest all are the local ones. The avoidance of local optimal points is the characteristic property of any metaheuristic algorithm in finding global values. Hence, the multimodal functions are given the task of exploring more region(s) in order to find the global optimal points. Thus, this piece of research work is applicable to both single objective and multi-objective test problems.

It is to be noted that all the test functions used in this study have the minimal value 0 except for Schwefel function which is having a minimal value as -12569.5. Table 2 shown below is describing the various properties of the different unimodal and multimodal functions. The dimension of these functions is shown as Dim which gives the count of various parameters used in the function. Range is depicting the boundary of the search space of the test function (Saremi, Mirjalili & Lewis, 2014). Various initial parameters used for the BBO, CBBO, SCBBO, ASCBBO, Grey Wolf Optimizer (GWO), Sine Cosine Algorithm (SCA), Ant Lion Optimizer (ALO), Genetic Algorithm (GA), Differential Evolution (DE), Ant Colony Optimization (ACO), Gravitational Search Algorithm (GSA) approaches are listed in Table 3.

5.1. Performance Analysis of ASCBBO and SCBBO

All the simulation experiments have been done in Matlab R2016a. The proposed algorithms are run ten times and the average is computed. Each time 500 iterations are being performed for the mutation operator on simple BBO and ASCBBO. In case of CEC 2014 benchmark functions, 1000 iterations have been carried out on each and every algorithm. The best values for average (mean) and standard deviation obtained in the last iteration for both the operators are observed and depicted in Tables 4 and 5. Notice that the ASCBBO shows much improvement in mean optimal values in both the cases as listed in Tables 4 and 5 when compared to CBBO (Saremi, Mirjalili & Lewis, 2014). Also the standard deviation values are showing the same trend.

In case of plain BBO operator as shown in Table 4, Penalty #2 test function gave the best values as compared to other benchmark functions. The other test functions Penalty #1, Fletcher, Rosenbrock, Schwefel, Sphere, Rastrigin, Griewank, Ackley and Quartic follow the descending sequence in terms of mean values. Same trend is also repeated by standard deviation values except for Griewank and Penalty #2 test functions. The decreasing order of standard deviation values is Penalty #1, Fletcher, Rosenbrock, Schwefel, Rastrigin, Sphere, Quartic, and finally Ackley test function.

The mutation operator shown in Table 5 follows the descending sequence of mean values as Penalty #1, Penalty #2, Fletcher, Rosenbrock, Schwefel, Griewank, Sphere, Quatric and Ackley test functions except in case of Rastrigin. In the same manner, Penalty #2 predicted the best standard deviation value in case of mutation probability followed by Penalty #1, Fletcher, Schwefel, Rosenbrock, Griewank, Sphere, Quartic and in the end Ackley test function except Rastrigin test function.

In case of CEC 2014 test functions also, when comparison has been made with state of the art algorithms like Grey Wolf Optimizer (GWO), Sine Cosine Algorithm (SCA), Ant Lion Optimizer (ALO), Genetic Algorithm (GA), Differential Evolution (DE), Ant Colony Optimization (ACO), Gravitational Search Algorithm (GSA), our methods (ASCBBO and SCBBO) have performed
phenomenally well and outperformed all the compared algorithms with much less mean and standard deviation values as given in Tables 6 and 7.

Thus from above discussion, it is quite clear that ASCBBO and SCBBO have given much improved results as compared to CBBO (Saremi, Mirjalili & Lewis, 2014). These methods can also be employed on parameter optimization for the prediction of a number of practical problems like software testing, software fault prediction, Glaucoma, Covid-19, glucose, mammography, cardiac problems, and plant leaf disease etc. (Khanna, Chauhan & Sharma, 2019; Khanna et al., 2019; Singh, Khanna & Garg, 2020; Thawkar, Singh & Khanna, 2021; Singh, Garg & Khanna, 2021)

5.2. Qualitative Analysis

Line graphs have also been plotted in the two cases of BBO and its mutation probability for computing the qualitative analysis of ASCBBO algorithm as given in Figs. 3 and 4. In each of the graphs shown in both the figures, the mean values are plotted against the number of iterations with respect to the particular benchmark function. All the graphs show higher rates of convergence in both the cases as compared to CBBO with respect to given test functions once again proving the superior performance of ASCBBO algorithm. Also, the convergence plots and box plots (Anova test) given in Figs. 5 and 6 clearly indicate the superior performance of our techniques on CEC 2014 test functions as they have been able to avoid local optimal solutions with high speed and accuracy of solutions.

5.3. Statistical Testing

Statistical tests should be conducted on the meta-heuristic algorithms to test their performance as proposed by Derrac et al. (2011). Mere computing the mean and standard deviation values are not enough in terms of overall inductiveness of the validity of the performance of these algorithms. Therefore, a nonparametric statistical test, which is Wilcoxon’s rank sum test, (Wilcoxon, 1992) should be carried out to test the validity of the metaheuristic algorithms separately. The significance level of the test is kept at 5%. It is a general practice that the $p$ values less than 0.05 are considered to be sufficient enough against the theory of null hypothesis. Also, the $p$ values depicted in Tables 3 and 4 in both the cases have proved the validity of the ASCBBO and SCBBO algorithms.

6. CONCLUDING REMARKS

In this paper, the use of chaotic sequence in an alternated manner in superior orbit is used as a source of population initialization. These have been used in selection, emigration, mutation and their combination stages. Chaotic maps in superior orbit produce more uniformly distributed chaotic sequences than used in chaotic BBO. Use of two chaotic attractors alternatively increase the complexity in chaotic sequences. Both the ways reduce biasness in the input chaotic data which helps the candidate solutions to arrive at global optimal points without being fallen in local optima and stuck there with much improved speed and precision of resultant solutions.

In all the cases, ASCBBO and SCBBO have been able to achieve much higher levels of optimization in comparison to CBBO. All the state of the art algorithms used in case of CEC 2014 test functions namely GWO, SCA, ALO, GA, DE, ACO and GSA are compared on the parameters: mean, standard deviation and $p$ value test. In some cases, the dip in values of optimal points in case of ASCBBO and SCBBO approaches are multi time than the compared algorithms. Thus, both theoretically and statistically, we have proved categorically that the methods suggested in this paper have been able to avoid premature convergence and achieved higher solution accuracy along with enhanced performance against some of the well-known metaheuristic algorithms which are trending recently (GWO, SCA, ALO, GSA) and also the most popular previous algorithms (GA, DE, ACO).

Future Work: In future, it is going to be an interesting phenomenon to deploy this technique on real world engineering optimization problems and other combinatorial optimization problems which are inherently complex in nature. Also, this technique could be applied on other metaheuristic algorithms
like Cuckoo Search, Grasshopper Optimization Algorithm, Whale optimization Algorithm, Salp Swarm Optimization Algorithm, Elephant Herding Optimization Algorithm and the like ones.

Table 2. Various benchmark functions used in the study

| Sr. No. | Benchmark Functions | Function Formula | Dim | Range | Optimal Value \( (f_{\text{opt}}) \) | Features |
|---------|---------------------|------------------|-----|-------|----------------------------------|----------|
| F1      | Sphere              | \( f(x) = \sum_{i=1}^{n} x_i^2 \) | 30  | [-100, 100] | 0 | Unimodal, Separable, Regular |
| F2      | Schwefel            | \( f(x) = \sum_{i=1}^{n} -x_i \sin(\sqrt{|x_i|}) \) | 30  | [-65.536, 64.536] | -12569.5 | Unimodal, Non-Separable, Regular |
| F3      | Rosenbrock          | \( f(x) = \sum_{i=1}^{n-1} \left[ 100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2 \right] \) | 30  | [-2.048, 2.048] | 0 | Unimodal, Non-Separable, Regular |
| F4      | Quartic             | \( f(x) = \sum_{i=1}^{n} |ix_i| \) | 30  | [-1.28, 1.28] | 0 | Unimodal, Separable, Regular |
| F5      | Penalty 1           | \( f(x) = \sum_{i=1}^{n} \left[ x_i - a \right]^2 \) | 30  | [-50, 50] | 0 | Multimodal, Non-Separable, Regular |
| F6      | Penalty 2           | \( f(x) = \sum_{i=1}^{n} \left[ x_i - a \right]^2 \) | 30  | [-500, 500] | 0 | Multimodal, Non-Separable, Regular |
| F7      | Griewank            | \( f(x) = \frac{1}{4000} \sum_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos \left( \frac{x_i}{\sqrt{i}} \right) + 1 \) | 30  | [-600, 600] | 0 | Multimodal, Non-Separable, Regular |
| F8      | Fletcher            | \( f(x) = \sum_{i=1}^{n} (A_i - B_i)^2 \) | 30  | [-\Pi, \Pi] | 0 | Multimodal, Non-Separable, Irregular |
| F9      | Ackley              | \( f(x) = -20 \exp \left(-0.5 \sum_{i=1}^{n} x_i^2 \right) - \exp \left( \sum_{i=1}^{n} \cos (2\pi x_i) \right) + 20 + e \) | 30  | [-50, 50] | 0 | Multimodal, Non-Separable, Regular |
| F10     | Rastrigin           | \( f(x) = \sum_{i=1}^{n} x_i^2 - 10\cos(2\pi x_i) + 10 \) | 30  | [-5.12, 5.12] | 0 | Multimodal, Separable, Regular |
Table 3. Initialization of parameter values for BBO, CBBO, SCBBO, ASCBBO, GWO, SCA, GSA, ACO, GA, DE and ALO

| Parameter initializations for BBO, CBBO, SCBBO and ASCBBO | Value |
|----------------------------------------------------------|-------|
| Size of population                                       | 30 (50 in CEC 2014 test problems) |
| Habitat modification probability                         | 1     |
| Immigration probability bounds per gene(inhabitant)      | [0, 1]|
| Step size for numerical integration of probabilities     | 1     |
| Maximum immigration (I) and Maximum Emigration (E)       | 1     |
| Probability of Mutated inhabitants                      | 0.005 |

| Parameter initializations for GWO | Value |
|----------------------------------|-------|
| a (Area Vector)                  | 2     |
| rᵢ₁, rᵢ₂ (Random Vectors)       | [0,1] |
| Size of population               | 50    |

| Parameter initializations for SCA | Value |
|----------------------------------|-------|
| Size of population               | 50    |
| a (Constant)                     | 2     |

| Parameter initializations for GSA | Value |
|----------------------------------|-------|
| Elitist Check (No. of fittest agents after stopping criterion) | 1     |
| Rpower (Exponent of distance between agents)               | 1     |
| Min_flag (1: minimum ; 0: maximum)                         | 1     |
| Size of population                                           | 50    |

| Parameter initializations for ACO | Value |
|----------------------------------|-------|
| Pheromone update constant        | 1     |
| Initial pheromone                | 10    |
| Pheromone sensitivity            | 0.3   |
| Visibility sensitivity           | 0.1   |
| Size of population               | 50    |

| Parameter initializations for GA | Value |
|----------------------------------|-------|
| Size of population               | 50    |
| Pc (Crossover probability)       | 0.95  |
| Pm(Mutation probability)         | 0.001 |
| Er (Elitism)                     | 0.2   |

| Parameter initializations for DE | Value |
|----------------------------------|-------|
| Size of population               | 50    |
| Lower bound of scaling factor    | 0.2   |
| Upper bound of scaling factor    | 0.8   |
| PCR (Crossover probability)      | 0.8   |

| Parameter initializations for ALO | Value |
|----------------------------------|-------|
| Size of population               | 50    |
Table 4. Performance comparison of ASCBBO on simple BBO operator

| Name of the Function | Criteria                  | CBBO (Saremi, Mirjalili & Lewis, 2014) | ASCBBO     |
|----------------------|---------------------------|----------------------------------------|------------|
| Sphere               | Mean                      | 50.19608, 17.88657, 0.427355           | 37.17486136, 10.25978, 0 |
|                      | Standard Deviation (SD)   |                                        |            |
|                      | P-Value                   | 0.427355                               | 0          |
| Ackley               | Mean                      | 16.93064, 1.259177, 0.212294           | 15.75670808, 0.844094, 0 |
|                      | Standard Deviation (SD)   |                                        |            |
|                      | P-Value                   | 0.212294                               | 0          |
| Griewank             | Mean                      | 142.9973, 28.83283, 0.57075            | 140.2411749, 29.65174, 0.208754 |
|                      | Standard Deviation (SD)   |                                        |            |
|                      | P-Value                   | 0.57075                                | 0          |
| Fletcher             | Mean                      | 828.365.4, 189.001, 0.001315            | 603882.8685, 145286.3, 3.2819E-205 |
|                      | Standard Deviation (SD)   |                                        |            |
|                      | P-Value                   | 0.001315                               | 0          |
| Schwefel             | Mean                      | 5614.696, 583.4996, 0.241322            | 5468.771204, 446.447, 0 |
|                      | Standard Deviation (SD)   |                                        |            |
|                      | P-Value                   | 0.241322                               | 0          |
| Penalty #1           | Mean                      | 18595.594, 13784.289, 0.73373           | 11690.87979, 8870459, 3.81152E-73 |
|                      | Standard Deviation (SD)   |                                        |            |
|                      | P-Value                   | 0.73373                                | 0          |
| Penalty #2           | Mean                      | 63937.965, 24464.971, 0.307489          | 55712.72373, 25818.606, 1.02433E-91 |
|                      | Standard Deviation (SD)   |                                        |            |
|                      | P-Value                   | 0.307489                               | 0          |
| Rosenbrock           | Mean                      | 1331.466, 691.616, 0.307489            | 1062.02699, 508.3196, 1.1419E-203 |
|                      | Standard Deviation (SD)   |                                        |            |
|                      | P-Value                   | 0.307489                               | 0          |
| Quartic              | Mean                      | 9618467, 6949806, 0.57075              | 8716411017, 5086379, 9.6298E-180 |
|                      | Standard Deviation (SD)   |                                        |            |
|                      | P-Value                   | 0.57075                                | 0          |
| Rastrigin            | Mean                      | 135.3346, 29.70814, 0.000246            | 131.0232107, 17.31392, 0 |
|                      | Standard Deviation (SD)   |                                        |            |
|                      | P-Value                   | 0.000246                               | 0          |
Table 5. Performance comparison of ASCBBO on mutation operator

| Name of the Function | Criteria            | CBBO (Saremi, Mirjalili & Lewis, 2014) | ASCBBO       |
|----------------------|---------------------|----------------------------------------|--------------|
|                      | Mean                | 57.38914                               | 50.1449926   |
|                      | Standard Deviation (SD) | 18.31544                               | 17.44709     |
|                      | P-Value             | 0.088973                               | 2.7128E-119  |
| Sphere               | Ackley              | 16.71692                               | 16.4889305   |
|                      | Mean                | 0.909707                               | 0.444036     |
|                      | Standard Deviation (SD) | 0.520523                               | 0            |
| Ackley               | Griewank            | 177.019                                | 160.480161   |
|                      | Mean                | 48.90111                               | 41.94935     |
|                      | Standard Deviation (SD) | 0.037635                               | 3.6301E-228  |
|                      | P-Value             | 0.520523                               | 0            |
| Griewank             | Fletcher            | 802.456                                | 684040.4607  |
|                      | Mean                | 263.524.6                              | 143144       |
|                      | Standard Deviation (SD) | 0.021134                              | 2.083E-118   |
|                      | P-Value             | 0.021134                               | 0            |
| Fletcher             | Schwefel            | 5,792.544                              | 5686.935398  |
|                      | Mean                | 677.7967                               | 400.0152     |
|                      | Standard Deviation (SD) | 0.140465                              | 3.8178E-291  |
|                      | P-Value             | 0.140465                               | 0            |
| Schwefel             | Penalty #1          | 22.986.907                             | 13,494,022.35|
|                      | Mean                | 10,639.030                             | 6080759      |
|                      | Standard Deviation (SD) | 0.121225                              | 3.09833E-61  |
|                      | P-Value             | 0.121225                               | 0            |
| Penalty #1           | Penalty #2          | 64,692,549                             | 59643824.68  |
|                      | Mean                | 30,294,479                             | 21092195     |
|                      | Standard Deviation (SD) | 0.57075                              | 4.50373E-98  |
|                      | P-Value             | 0.57075                                | 0            |
| Penalty #2           | Rosenbrock          | 1,378.675                              | 1021.611419  |
|                      | Mean                | 655.9377                               | 403.1622     |
|                      | Standard Deviation (SD) | 0.161972                              | 9.083E-137   |
|                      | P-Value             | 0.161972                               | 0            |
| Rosenbrock           | Quartic             | 11.30964                               | 10.26638274  |
|                      | Mean                | 4.400924                               | 3.569843     |
|                      | Standard Deviation (SD) | 0.241322                              | 6.6395E-267  |
|                      | P-Value             | 0.241322                               | 0            |
| Quartic              | Rastrigin           | 77.96833                               | 149.3794566  |
|                      | Mean                | 12.21388                               | 21.35098     |
|                      | Standard Deviation (SD) | 0.57075                              | 0            |
|                      | P-Value             | 0.57075                                | 0            |
### Table 6. Performance comparison of GWO, SCA, ALO, GA and DE algorithms on CEC 2014 test suite

| Criteria | F’s | GWO | SCA | ALO | GA | DE |
|----------|-----|-----|-----|-----|----|----|
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |
| Mean SD | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value | Mean | P-Value | Best Value |

continued on next page
| Criteria | F\(_n\)'s | GWO | SCA | ALO | GA | DE |
|----------|-----------|------|-----|-----|----|----|
| Mean SD  | F16       | 92068497.45 | 64261783.93 | 2.294E-108 | 0 | 352841422.8 |
| P-Value  |          | 47285093.5    | 12099437.6    | 0 | 30601401.58 |
| Best Value |         | 42074531.3     | 10059383.9    | 7.59078E-28 |
| Mean SD  | F17       | 90978158.14 | 97170804.05 | 1.4422E-99 | 19445735.05 |
| P-Value  |          | 41101836.0    | 13310699.04  | 78293325.3    |
| Best Value |         | 37104770.9     | 13157303.95   | 31477798.12   |
| Mean SD  | F18       | 53189723.43 | 30839187.83 | 4.37321E-80 | 18026943.81 |
| P-Value  |          | 34672206.47   | 35221688.93    | 0 | 36276800.12 |
| Best Value |         | 40473098.1     | 69236836.84   | 1.1286E-14   |
| Mean SD  | F19       | 46218374.45 | 23070217.38 | 5.336E-104 | 20338178.15 |
| P-Value  |          | 32667617.6    | 58379900.03    | 0 | 21384086.82 |
| Best Value |         | 21810384.95    | 75243571.75   | 32616075.92   |
| Mean SD  | F20       | 4026830.76   | 31831306.07 | 3.1582E-209 | 10090118.93 |
| P-Value  |          | 30947385.48   | 10848766.13    | 0 | 23482499.95 |
| Best Value |         | 35068791.78    | 6258060.15     | 5.04237E-19 |
| Mean SD  | F21       | 61553764.42 | 54376747.32 | 3.7072E-93 | 1677128.53 |
| P-Value  |          | 32014500.1    | 10776699.63    | 0 | 19411013.6 |
| Best Value |         | 28319199.72    | 10092657.41    | 2.9063E-107 |
| Mean SD  | F22       | 71336044.44 | 22436897.93 | 1.9706E-140 | 40198262.59 |
| P-Value  |          | 31970599.3    | 56362489.25    | 263814081.7     |
| Best Value |         | 28354293.3     | 12519602.23    | 4.66907E-16 |
| Mean SD  | F23       | 43449516.13 | 23071013.75 | 3.72553E-51 | 23601062.25 |
| P-Value  |          | 34672544.8    | 10436839.3    | 11514391.47    |
| Best Value |         | 20798911.7     | 15093623.77    | 8.91914E-14 |
| Mean SD  | F24       | 85533779.63 | 65486951.33 | 5.3342E-104 | 25454062.19 |
| P-Value  |          | 299214208.9   | 7780468.38    | 26127053.97     |
| Best Value |         | 27920556.2     | 66687490.19    | 1.01313E-12 |
| Mean SD  | F25       | 64321517.6   | 40593000.26 | 1.97154E-75 | 35617295.44 |
| P-Value  |          | 34465716.2    | 11387341.1    | 31802081.84    |
| Best Value |         | 28986051.6     | 16411084.82    | 1.14293E-08 |
| Mean SD  | F26       | 31417895.49 | 19583195.34 | 6.2892E-75 | 18085676.12 |
| P-Value  |          | 41650311.4    | 11339866.6    | 28835069.13    |
| Best Value |         | 28002375.9     | 69979375.8     | 5.9935E-14    |
| Mean SD  | F27       | 63646048.72 | 23781235.77 | 1.02524E-72 | 46874273.37 |
| P-Value  |          | 34550298.8    | 35536502.07    | 4056858.96    |
| Best Value |         | 41778621.4     | 4661796.07     | 1.13666E-09 |
| Mean SD  | F28       | 89838418.95 | 65031200.03 | 1.17784E-50 | 28290051.43 |
| P-Value  |          | 55386522.7    | 7069665.01    | 5.4566E-27    |
| Best Value |         | 38487134.2     | 18140760.01    | 1.9872E-15 |
| Mean SD  | F29       | 9457813.96  | 933258708.08 | 4.3532E-67 | 30226702.53 |
| P-Value  |          | 434428559.3    | 10052892.97    | 0 | 23827450.95 |
| Best Value |         | 39735746.2     | 3594877.86     | 5.09478E-08 |
| Mean SD  | F30       | 5366263.85  | 38030606.00 | 1.1722E-136 | 19445735.05 |
| P-Value  |          | 33812120.4    | 40423146.59    | 0 | 28293325.3 |
| Best Value |         | 33966766.9     | 4804798.902    | 2.08155E-12   | 23602812.12    |

International Journal of Applied Metaheuristic Computing
Volume 13 • Issue 1
| Criteria          | F(n)'s   | ACO | GSA | ASCBBO | SCBBO |
|------------------|----------|-----|-----|--------|-------|
| Mean SD P-Value  | Mean     | 1.118153230 | 11937855723 | 15545748.44 | 17331986.44 |
| Best Value       | 0.8481579213 | 0.5 | 1.1184092470 | 5668303.702 | 2506811.25 |
|                  | Mean     | 1.187796885 | 10793061835 | 6.0614561.61 | 1.952826E-29 |
|                  | 0.7.2024E-272 | 0 | 1.2166416-63 | 11437663.9 | 15559403.21 |
|                  | Mean     | 1.119138207 | 10404035657 | 27167185.9 | 10776227.62 |
|                  | 172434492.2 | 10793061835 | 0 | 1.2166416-63 | 1.0776227.62 |
|                  | Mean     | 1.131905424 | 8746848523  | 12361964.27 | 10040023.07 |
|                  | 0.2.29594E-94 | 0 | 0 | 0 | 5.72176E-28 |
|                  | Mean     | 1.041310155 | 8373429232  | 2033128.54 | 1.9080371.54 |
|                  | 299966089.1 | 0.4925713646 | 4.40135E-07 | 2052117.8 | 2635470.768 |
|                  | Mean     | 1.123473446 | 14981779398 | 2400923.51 | 1.9424200.97 |
|                  | 160120928.8 | 7840977.61 | 1.34467E+10 | 8176468.668 | 0.026136442 |
|                  | Mean     | 1.135757852 | 5552327878  | 2871029.04 | 1.5688832.81 |
|                  | 93937754.32 | 665585370.6 | 5.226335.476 | 7189161.509 | 3.84619E-37 |
|                  | Mean     | 1.103982121 | 5459806640  | 22362167.6 | 1.7446416.77 |
|                  | 210993048.1 | 2231386040 | 3.474443.513 | 426801.357 | 8361832.406 |
|                  | Mean     | 1.104012581 | 11135659989 | 13006886.26 | 21499023.79 |
|                  | 327294797.8 | 11272493449 | 3.0649648.60 | 3708891961.6 | 1.1693373.56 |
|                  | Mean     | 1.132710922 | 6944785680  | 27084362.19 | 4.54954E-101 |
|                  | 2835658315 | 8291781.135 | 1.7937E-129 | 47916451.381 | 6.806335E-58 |
|                  | Mean     | 1.103623077 | 48404947332 | 22877731.91 | 16592818.7 |
|                  | 325751804.6 | 218629469 | 4.2591453.854 | 13940025.022 | 6.8361832.406 |
|                  | Mean     | 1.113503783 | 4833788231  | 1547030.93 | 1.9563545.37 |
|                  | 191833605 | 8376207874 | 1.9514E-113 | 2351427.99 | 1.61057E-75 |
|                  | Mean     | 1.112054168 | 6962503268 | 24546729.88 | 9565335.47 |
|                  | 110289283.3 | 4942727716 | 3.7482026.599 | 21542729.99 | 7.1654904.56 |
|                  | Mean     | 1.116447282 | 7286903474 | 20040842.53 | 15062632.29 |
|                  | 115845116.1 | 3263910120 | 1.0610253.4 | 2155051.677 | 1.120742.12 |
|                  | Mean     | 1.144280218 | 11089860883 | 23703059.66 | 1.5416082.59 |
|                  | 247289417.7 | 4746103295 | 8.2574675.73 | 1.0412530.212 | 3.174520.212 |
|                  | Mean     | 1.108963548 | 5659847532  | 3.8674689.6 | 6.64759E-59 |
|                  | 0.108963548 | 5659847532 | 1.5493787.77 | 11578302.03 | 5.1578302.03 |

continued on next page
Table 7. Continued

| Criteria | F's | ACO | GSA | ASCBBO | SCBBO |
|----------|-----|-----|-----|--------|-------|
| Mean SD  | F16 | 909911166.7 | 67973333.4 | 0 | 823134475.2 |
| P-Value  | 2872015683 | 1079099605 | 0.5 | 1941805257 | 20803500.83 |
| Best Value | 5238569.75 | 5.849E-117 | 16781720.34 | 17183643.71 |
| Mean SD  | F17 | 228898666.1 | 228898666.1 | 0 | 9907111257 |
| P-Value  | 7572499057 | 5507820061 | 0.5 | 4094171147 | 18542494.59 |
| Best Value | 6691233.91 | 1.50302E-60 | 11062020.9 | 14371660.71 |
| Mean SD  | F18 | 939526969.2 | 128324765.8 | 0 | 8481759213 |
| P-Value  | 11349802839 | 657039205 | 0.5 | 4211930063 | 21369095.08 |
| Best Value | 3219630.92 | 1.65496E-95 | 19467352.85 | 15238567.08 |
| Mean SD  | F19 | 1068101483 | 454910874.3 | 9.9066E-27 | 547780806 |
| P-Value  | 8094412370 | 2805912796 | 0.5 | 5704859901 | 14635316.4 |
| Best Value | 3082837687 | 2.43432E-75 | 11537652.86 | 15213426.44 |
| Mean SD  | F20 | 1177704644 | 180057372.5 | 0 | 9729961812 |
| P-Value  | 12436575625 | 10846182772 | 0.5 | 2563591480 | 24515493.19 |
| Best Value | 9.801E-152 | 16553179.82 | 19151820.18 |
| Mean SD  | F21 | 1239247896 | 248797093.3 | 6.4725E-25 | 884228853.4 |
| P-Value  | 4874128863 | 1462905377 | 0.5 | 2836801818 | 27067496.26 |
| Best Value | 3252107575 | 2.688E-125 | 22131779.92 | 14206380.27 |
| Mean SD  | F22 | 1086890900 | 219423319.1 | 0 | 838827867.9 |
| P-Value  | 5508995837 | 151661129 | 0.5 | 4376160057 | 23123900.17 |
| Best Value | 4293607.72 | 4.7676E-91 | 19765319.23 | 13282159.71 |
| Mean SD  | F23 | 1155823850 | 200089979.9 | 1.656E-182 | 925161638.5 |
| P-Value  | 7315309905 | 1076356288 | 0.5 | 171333967.8 | 15710323.17 |
| Best Value | 5.4872E-89 | 165703913.2 | 9153785.45 |
| Mean SD  | F24 | 1120054168 | 110289283.3 | 0 | 103718174 |
| P-Value  | 6962503268 | 4942727716 | 0.5 | 1485128199 | 24547629.88 |
| Best Value | 7482026.59 | 5.6959E-120 | 11532808.18 | 13386421.46 |
| Mean SD  | F25 | 1142869513 | 1844030116 | 0 | 1012476900 |
| P-Value  | 9242949253 | 1174421274 | 0.5 | 8412508006 | 21013548.88 |
| Best Value | 10500970.43 | 6.26541E-48 | 15606241.49 | 14309906.27 |
| Mean SD  | F26 | 1153929247 | 7276512711 | 0 | 1089635478 |
| P-Value  | 5404029074 | 2770176740 | 0.5 | 2567208055 | 21872189.81 |
| Best Value | 8.2594E-85 | 9170735.69 | 16847999.59 |
| Mean SD  | F27 | 1611131561 | 42278866.12 | 1.296E-141 | 1581235888 |
| P-Value  | 1415617231 | 4335144765 | 0.5 | 11090746971 | 26630796.9 |
| Best Value | 10743625.75 | 3.24213E-56 | 1903906.28 | 15597713.09 |
| Mean SD  | F28 | 1088075001 | 331137329.4 | 0 | 823134475.2 |
| P-Value  | 21646843273 | 15745337898 | 0.5 | 1038821537 | 16552678.33 |
| Best Value | 6.94976.1777 | 1549787.77 | 1712952.124 |
| Mean SD  | F29 | 1001523702 | 125852410.5 | 0 | 881426422.7 |
| P-Value  | 11442504152 | 10011927316 | 0.5 | 4700578956 | 20470523.58 |
| Best Value | 8288707.65 | 5.90716E-75 | 11062020.9 | 18043935.43 |
| Mean SD  | F30 | 1136321458 | 286507355 | 1.6532E-278 | 8841646428.8 |
| P-Value  | 8483968696 | 7501652274 | 0.5 | 4094171147 | 21768371.46 |
| Best Value | 1.75406E-96 | 19467352.85 | 13443503.74 |

International Journal of Applied Metaheuristic Computing
Volume 13 • Issue 1
Figure 3a. Convergence curves for simple BBO operator
Figure 3b. Convergence curves for simple BBO operator

Figure 4a. Convergence curves for mutation operator on CBBO and ASCBBO
Figure 4b.

Figure 5a. Convergence curves for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO
Figure 5b. Convergence curves for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO
Figure 5c. Convergence curves for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO.

F9

F10

F11

F12
Figure 5d. Convergence curves for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO
Figure 5e. Convergence curves for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO.
Figure 5f. Convergence curves for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO
Figure 5g. Convergence curves for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO

Figure 6a. Box plot graphs for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO
Figure 6b. Box plot graphs for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO
Figure 6c. Box plot graphs for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO
Figure 6d. Box plot graphs for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO
Figure 6e. Box plot graphs for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO
Figure 6f. Box plot graphs for selection and migration operator combined on GWO, SCA, ALO, GA, DE, ACO, GSA, ASCBBO and SCBBO
REFERENCES

Alatas, B. (2010a). Chaotic bee colony algorithms for global numerical optimization. *Expert Systems with Applications, 37*(8), 5682–5687. doi:10.1016/j.eswa.2010.02.042

Alatas, B. (2010b). Chaotic harmony search algorithms. *Applied Mathematics and Computation, 216*(9), 2687–2699. doi:10.1016/j.amc.2010.03.114

Alatas, B., Akin, E., & Ozer, A. (2009). Chaos embedded particle swarm optimization algorithms. *Chaos, Solitons, and Fractals, 40*(4), 1715–1734. doi:10.1016/j.chaos.2007.09.063

Arslan, H., & Toz, M. (2019). Data clustering based on fuzzy c-means and chaotic whale optimization algorithms. *Sigma, 37*(4), 1107–1128.

Arul, R., Ravi, G., & Velusami, S. (2013). An improved harmony search algorithm to solve economic load dispatch problems with generator constraints. *Electrical Engineering, 96*(1), 55–63. doi:10.1007/s00202-012-0276-0

Ashish. (2014). *Fixed-point Iterative Procedure in Fractal and Chaos and the Stability of Functional Equations* (PhD). Maharishi Dayanand University, Rohtak, India.

Bejnariu, S. I., Costin, H., Rotaru, F., & Luca, R. (2019). Data clustering by nature-inspired algorithms and chaotic maps. In *2019 E-Health and Bioengineering Conference (EHB)*. IEEE.

Bouyer, A., & Farajzadeh, N. (2015). An optimized K-harmonic means algorithm combined with modified particle swarm optimization and Cuckoo search algorithm. *Journal of Intelligent Systems, 29*(1), 1–18. doi:10.1515/jisyss-2015-0009

Danca, M., Fečkan, M., & Romera, M. (2014). Generalized form of Parrondo’s paradoxical game with applications to chaos control. *International Journal of Bifurcation and Chaos in Applied Sciences and Engineering, 24*(01), 1450008. doi:10.1142/S0218127414500084

Danca, M., & Tang, W. (2016). Parrondo’s paradox for chaos control and anticontrol of fractional-order systems. *Chinese Physics B, 25*(1), 010505. doi:10.1088/1674-1056/25/1/010505

Derrac, J., García, S., Molina, D., & Herrera, F. (2011). A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm and Evolutionary Computation, 1*(1), 3–18. doi:10.1016/j.swevo.2011.02.002

Dhanusha, C., & Kumar, A. S. (2021). Deep recurrent Q reinforcement learning model to predict the alzheimer disease using smart home sensor data. *IOP Conference Series. Materials Science and Engineering, 1074*(1), 012014. doi:10.1088/1757-899X/1074/1/012014

dos Santos Coelho, L., & Mariani, V. C. (2008). Use of chaotic sequences in a biologically inspired algorithm for engineering design optimization. *Expert Systems with Applications, 34*(3), 1905–1913. doi:10.1016/j.eswa.2007.02.002

Du, D., Simon, D., & Ergezer, M. (2009). Biogeography-based optimization combined with evolutionary strategy and immigration refusal. In *2009 IEEE International Conference on Systems, Man and Cybernetics* (pp. 997-1002). IEEE. doi:10.1109/ICSMC.2009.5346055

Fang, U., Li, J., Lu, X., Ali, M., Gao, L., & Xiang, Y. (2021). Chaotic-to-fine clustering for unlabeled plant disease images. arXiv preprint arXiv:2101.06820.

Fister, I., Yang, X. S., & Brest, J. (2014). On the randomized firefly algorithm. In *Cuckoo search and firefly algorithm* (pp. 27–48). Springer. doi:10.1007/978-3-319-02141-6_2

Gálvez, J., Cuevas, E., Becerra, H., & Avalos, O. (2019). A hybrid optimization approach based on clustering and chaotic sequences. *International Journal of Machine Learning and Cybernetics, 11*(2), 359–401. doi:10.1007/s13042-019-00979-6

Gandomi, A., Yang, X., Talatahari, S., & Alavi, A. (2013). Firefly algorithm with chaos. *Communications in Nonlinear Science and Numerical Simulation, 18*(1), 89–98. doi:10.1016/j.cnsns.2012.06.009

Gharooni-fard, G., Moein-darbari, F., Deldari, H., & Morvaridi, A. (2010). Scheduling of scientific workflows using a chaos-genetic algorithm. *Procedia Computer Science, 1*(1), 1445–1454. doi:10.1016/j.procs.2010.04.160
Giri, P. K., De, S. S., Dehuri, S., & Cho, S. B. (2017). Locally and globally tuned chaotic biogeography-based optimization algorithm. In 2017 International Conference on Information Technology (ICIT) (pp. 152-158). IEEE. doi:10.1109/ICIT.2017.44

Goel, S. (2011). Generation of New Fractal Models and Study of Stability of Iterative Procedures (Ph.D). Gautam Buddh Technical University, Lucknow, India.

Guo-ping, Y., San-yang, L., Jian-ke, Z., & Hai-lin, F. (2016). Parameter estimation for discrete chaotic systems with biogeography-based optimization. Journal of Lanzhou University: Natural Sciences, 52(6), 832–837.

Han, Y., Ding, J., Du, L., & Lei, Y. (2021). Control and anti-control of chaos based on the moving largest Lyapunov exponent using reinforcement learning. 10.21203/rs.3.rs-176868/v1

Heidari, A., Mirvahabi, S., & Homayouni, S. (2015). An effective hybrid support vector regression with chaos-embedded biogeography-based optimization strategy for prediction of earthquake-triggered slope deformations. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences, XL-1(W5), 301–305. doi:10.5194/isprsarchives-XL-1-W5-301-2015

Jalili, S., Hosseinizadeh, Y., & Kaveh, A. (2014). Chaotic biogeography algorithm for size and shape optimization of truss structures with frequency constraints. Periodica Polytechnica. Civil Engineering, 58(4), 397–422. doi:10.3311/PPci.7466

Jin, Q., Lin, N., & Zhang, Y. (2021). K-means clustering algorithm based on chaotic adaptive artificial bee colony. Algorithms, 14(2), 53.

Kaur, A., Pal, S. K., & Singh, A. P. (2020). Hybridization of chaos and flower pollination algorithm over K-means for data clustering. Applied Soft Computing, 97, 105523. doi:10.1016/j.asoc.2019.105523

Khanna, M., Chaudhary, A., Toofani, A., & Pawar, A. (2019). Performance comparison of multi-objective algorithms for test case prioritization during web application testing. Arabian Journal for Science and Engineering, 44(11), 9599–9625. doi:10.1007/s13369-019-03817-7

Khanna, M., Chauhan, N., & Sharma, D. K. (2019). Search for prioritized test cases during web application testing. International Journal of Applied Metaheuristic Computing, 10(2), 1–26. doi:10.4018/IJAMC.2019040101

Krasnosel’skiı, M. A. (1955). Two remarks on the method of successive approximations. Uspekhi Matematicheskhi Nauk, 10, 123–127.

Lesmoir-Gordon, N., & Rood, W. (2014). Introducing fractals: A graphic guide. Icon Books Ltd.

Levinsohn, E. A., Mendoza, S. A., & Peacock-López, E. (2012). Switching induced complex dynamics in an extended logistic map. Chaos, Solitons, and Fractals, 45(4), 426–432. doi:10.1016/j.chaos.2011.12.020

Li-Jiang, Y., & Tian-Lun, C. (2002). Application of chaos in genetic algorithms. Communications in Theoretical Physics, 38(2), 168–172. doi:10.1088/0253-6102/38/2/168

Linglong, T., Yehui, C., & Changkai, L. (2018). Research on image segmentation optimization algorithm based on chaotic particle swarm optimization and fuzzy clustering. In Proceedings of the 2018 7th International Conference on Software and Computer Applications (pp. 178-182). doi:10.1145/3185089.3185139

Liu, B., Wang, L., Jin, Y. H., Tang, F., & Huang, D. X. (2005). Improved particle swarm optimization combined with chaos. Chaos, Solitons, and Fractals, 25(5), 1261–1271. doi:10.1016/j.chaos.2004.11.095

Lu, X., Wu, Q., Zhou, Y., Ma, Y., Song, C., & Ma, C. (2019). A dynamic swarm firefly algorithm based on chaos theory and max-min distance algorithm. Traitement du Signal, 36(3), 227–231. doi:10.18280/ts.360304

Ma, H., Simon, D., Siarry, P., Yang, Z., & Fei, M. (2017). Biogeography-based optimization: A 10-year review. IEEE Transactions on Emerging Topics in Computational Intelligence, 1(5), 391–407. doi:10.1109/TETCI.2017.2739124

MacArthur, R. H. (2010). The Theory of Island Biogeography Revisited (J. B. Losos & R. E. Ricklefs, Eds.). Princeton University Press.

Maier, M. P., & Peacock-López, E. (2010). Switching induced oscillations in the logistic map. Physics Letters. [Part A], 374(8), 1028–1032. doi:10.1016/j.physleta.2009.12.039
Mann, W. R. (1953). Mean value methods in iteration. *Proceedings of the American Mathematical Society, 4*(3), 506–510. doi:10.1090/S0002-9939-1953-0054846-3

Mendoza, S. A., Matt, E. W., Guimarães-Blandón, D. R., & Peacock-López, E. (2018). Parrondo’s paradox or chaos control in discrete two-dimensional dynamic systems. *Chaos, Solitons, and Fractals, 106*, 86–93. doi:10.1016/j.chaos.2017.11.011

Mingjun, J., & Huanwen, T. (2004). Application of chaos in simulated annealing. *Chaos, Solitons, and Fractals, 21*(4), 933–941. doi:10.1016/j.chaos.2003.12.032

Negi, A. & Rani, M. (2008a). A new approach to dynamic noise on superior Mandelbrot set. *Chaos, Solitons & Fractals, 36*(4), 1089-1096.

Negi, A. & Rani, M. (2008b). Midgets of superior Mandelbrot set. *Chaos, Solitons & Fractals, 36*(2), 237-245.

Negi, A., Rani, M., & Mahanti, P. K. (2008). Computer simulation of the behaviour of Julia sets using switching processes. *Chaos, Solitons & Fractals, 37*(4), 1187-1192.

Peacock-López, E. (2011). Seasonality as a Parrondian game. *Physics Letters. [Part A], 375*(35), 3124–3129. doi:10.1016/j.physleta.2011.06.063

Pecora, L. M., & Carroll, T. L. (1990). Synchronization in chaotic systems. *Physical Review Letters, 64*(8), 821–825. doi:10.1103/PhysRevLett.64.821 PMID:10042089

Rani, M., & Agarwal, R. (2009). A new experimental approach to study the stability of logistic map. *Chaos, Solitons, and Fractals, 41*(4), 2062–2066. doi:10.1016/j.chaos.2008.08.022

Rani, M., & Yadav, A. (2016). Parrondo’s paradox in the superior logistic map. *Int. J Tech. Res, 1*(2), 1–8.

Saremi, S., & Mirjalili, S. (2013). Integrating chaos to biogeography-based optimization algorithm. *International Journal of Computer and Communication Engineering, 2*(6), 655–658. doi:10.7763/IJCCCE.2013.V2.268

Saremi, S., Mirjalili, S., & Lewis, A. (2014). Biogeography-based optimisation with chaos. *Neural Computing & Applications, 25*(5), 1077–1097. doi:10.1007/s00521-014-1597-x

Simon, D. (2008). Biogeography-based optimization. *IEEE Transactions on Evolutionary Computation, 12*(6), 702–713. doi:10.1109/TEVC.2008.919004

Simon, D. (2011). A probabilistic analysis of a simplified biogeography-based optimization algorithm. *Evolutionary Computation, 19*(2), 167–188. doi:10.1162/EVCO_a_00018 PMID:20807078

Singh, L. K., Garg, H., & Khanna, M. (2021). An artificial intelligence-based smart system for early glaucoma recognition using OCT images. *International Journal of E-Health and Medical Communications, 12*(4), 32–59. doi:10.4018/IJEHMC.20210701.oa3

Singh, L. K., Khanna, M., & Garg, H. (2020). Multimodal biometric based on fusion of ridge features with minutiae features and face features. *International Journal of Information System Modeling and Design, 11*(1), 37–57. doi:10.4018/IJISMD.2020010103

Singh, S. L., Mishra, S. N., & Sinkala, W. (2012). A new iterative approach to fractal models. *Communications in Nonlinear Science and Numerical Simulation, 17*(2), 521–529. doi:10.1016/j.cnsns.2011.06.014

Singh, T. (2020). A chaotic sequence-guided Harris hawks optimizer for data clustering. *Neural Computing & Applications, 32*(23), 17789–17803. doi:10.1007/s00521-020-04951-2

Strogatz, S. H. (2018). *Nonlinear dynamics and chaos with student solutions manual: With applications to physics, biology, chemistry, and engineering*. CRC Press. doi:10.1201/9780429399640

Talatahari, S., Azar, B. F., Sheikholeslami, R., & Gandomi, A. H. (2012). Imperialist competitive algorithm combined with chaos for global optimization. *Communications in Nonlinear Science and Numerical Simulation, 17*(3), 1312–1319. doi:10.1016/j.cnsns.2011.08.021

Thawkar, S., Singh, L. K., & Khanna, M. (2021). Multi-objective techniques for feature selection and classification in digital mammography. *Intelligent Decision Technologies, 15*(1), 115–125. doi:10.3233/IDT-200049

Wallace, A. (2005). *The geographical distribution of animals (two volumes)*. Adamant Media Corporation.
Deepak Kumar is a research scholar in the department of computer science and has interests in computational intelligence, nature inspired algorithms, fractals and chaos, data mining etc. Mamta Rani is an associate professor and HOD in Dept. of Computer Science in the Central University of Rajasthan, Kishangarh, Rajasthan.