Hindawi
Modelling and Simulation in Engineering
Volume 2018, Article ID 4945157, 14 pages
https://doi.org/10.1155/2018/4945157

Research Article

Application and Development of Enhanced Chaotic Grasshopper Optimization Algorithms

Akash Saxena[1], **Shalini Shekhawat**[1] **and Rajesh Kumar**[2]

[1] *Swami Keshvanand Institute of Technology, Jaipur 302017, India*
[2] *Malaviya National Institute of Technology, Jaipur 302017, India*

Correspondence should be addressed to Akash Saxena; aakash.saxena@hotmail.com

Received 30 December 2017; Revised 23 March 2018; Accepted 8 April 2018; Published 23 May 2018

Academic Editor: Gaetano Sequenzia

In recent years, metaheuristic algorithms have revolutionized the world with their better problem solving capacity. Any metaheuristic algorithm has two phases: exploration and exploitation. The ability of the algorithm to solve a difficult optimization problem depends upon the efficacy of these two phases. These two phases are tied with a bridging mechanism, which plays an important role. This paper presents an application of chaotic maps to improve the bridging mechanism of Grasshopper Optimisation Algorithm (GOA) by embedding 10 different maps. This experiment evolves 10 different chaotic variants of GOA, and they are named as Enhanced Chaotic Grasshopper Optimization Algorithms (ECGOAs). The performance of these variants is tested over ten shifted and biased unimodal and multimodal benchmark functions. Further, the applications of these variants have been evaluated on three-bar truss design problem and frequency-modulated sound synthesis parameter estimation problem. Results reveal that the chaotic mechanism enhances the performance of GOA. Further, the results of the Wilcoxon rank sum test also establish the efficacy of the proposed variants.

1. Introduction

Optimization is a term which refers to the selection of the best option amongst the given set of alternatives. Examples of optimization processes are everywhere such as in business, human resource management, challenging engineering design problems, transportation, profit making propositions, and industrial applications. Optimization can be done for the maximization of any proposition or minimization of any proposition. In engineering problems particularly, the use of maximization is for efficiency maximization, classification accuracy maximization, and revenue or profit maximization, and on the other hand, minimization can be performed for cost, loss, risk, and execution time of any engineering process. Apart from these classifications of optimization, another classification of the optimization problem can be done on the basis of constraints. An optimization problem without any constraints is called unconstrained optimization; similarly another type is constrained optimization with linear and nonlinear constraints. Another classification can be done on the basis of the objective of the optimization; when an optimization problem aims towards a single objective, it is called the single objective optimization problem, and similarly when it aims towards multiobjectives, the same is called the multiobjective optimization problem [1]. A recent trend is to employ metaheuristic optimization algorithms to solve challenging problems of the real world. The term metaheuristic refers to problem-independent higher level heuristic mechanism [2]. In recent years, applications of metaheuristic algorithms in engineering problems have been reported. The successful and effective implementation of these algorithms on real applications has attracted the attention of researchers to work in this direction.

The metaheuristic optimization approaches can be subdivided into three categories:

(1) Evolutionary computing-based algorithms [3–5]

(2) Physics law-based algorithms [6–8]

(3) Swarm intelligence-based algorithms [9–16]

Evolutionary-based algorithms are based on natural evolution due to environmental pressure [2]. These algorithms employ selection or mixing criterion to generate an optimal solution set which possesses higher fitness values. Basic virtues of these algorithms are of stochastic nature, incorporating crossover and termination operators for hybridizing the solutions and enhancing the fitness value. A few examples of these algorithms are Genetic Algorithm [4] and Evolution Strategy and Evolutionary Programming [5]. Another class of algorithms is the algorithms which are inspired from physics and based on the laws of fundamental physics. A few examples of these algorithms are Gravitational Search Algorithm [14], Big Bang-Big Crunch Algorithm [7], and Black hole Algorithm [8].

The third category is based on swarm intelligence methods, where the cognitive and social behavior of the natural swarms like birds and school of fish is mimicked in the form of simulation. The most famous algorithm in this category is Particle Swarm Optimization (PSO), which works on the philosophy "Follow the Leader" [9]. Other examples of these algorithms are Bat Algorithm [10], Firefly Algorithm [11], and Cuckoo Search Algorithm [12]. A recently published swarm algorithm, which became popular nowadays, is Grey Wolf Optimizer (GWO), the algorithm that mimics the hunting behavior of grey wolves and is a fine example of the compliance of the social hierarchy of the wolf pack during searching, attacking, and hunting phases. A novel algorithm based on crow behavior named as Crow Search Algorithm (CSA) has been proposed [14]. CSA mimics the behavior of crow to store their excess food in hiding places and retrieve it when it is needed. Similarly, Ant Lion Optimizer Algorithm [15] and Grasshopper Optimisation Algorithm [16] are also a good example of social mimicry of the natural swarms.

The applications of swarm algorithms are very well reported in the literature and in many design problems, namely, Automatic Generation Control [17], Unit Commitment [18, 19], Feature Selection [20], and Ambient Air Quality Classification [21].

Many optimization algorithms have employed chaotic sequences over the random walk (random numbers generation) due to the fact that the random walk not always implements the global search well. Thus in some cases, the algorithm development is based on chaotic variables instead of random variables, and these algorithms are called chaotic algorithms [22–26]. A chaotic Firefly Algorithm was proposed in [22]. In this work, attractive movement of fireflies was simulated with ten chaotic maps. Chaotic sequences are used for parameters a and A in the Chaotic GWO approach [27]. Chaos enhanced Accelerated Particle Swarm Optimization (CAPSO) which was proposed by Gandomi et al. [25]. An attraction parameter was tuned with normalized chaotic maps in that work. Chaotic Bat Algorithm was proposed in [23], and the tuning of the crucial parameter of this algorithm was done with the help of chaotic maps. Different ten chaotic maps were employed in gravitational search algorithm in [28].

Two mechanisms: diversification and intensification are essential parts of any swarm algorithm. The initial phase of any swarm algorithm started with random search; usually this process swifts and holds responsibility to search every possible direction of the search space, and thus, the process is random in nature. On the other hand, the intensification process is strategic. The outcome of this process is specific and is treated as the solution of the problem. It is empirical to say that speed of these processes is different. Every algorithm employs a bridging mechanism to maintain a good amount of trade-off between these two processes. Some algorithms use different operators and different models of random walks in different phases, and in short, these operators/mechanisms help the algorithm to maintain a fair balance between these two processes. This paper investigates the impact of different chaotic sequences on the bridging mechanism of the GOA, by evaluating the performance of the proposed variants on standard benchmark functions and real applications. 10 different chaotic sequences are embedded with the parameter c, and the careful observation is presented. Following research objectives are framed for this work:

(1) To employ 10 different chaotic maps through normalized function to propose chaotic variants of GOA. These variants are developed on the basis of adaptive, chaotic, and monotonically decreasing parameter c.
(2) To conduct a nonparametric Wilcoxon rank sum test for observing the efficacy of the chaotic variants with the GOA by observing p values.
(3) To apply these variants on three-bar truss design and parameter estimation for frequency-modulated sound waves and compare the performance with the other contemporary algorithms.

Remaining part of the paper is organized as follows: in Section 2, brief details of different chaotic maps are incorporated. In Section 3, an overview of GOA is presented. The development of chaotic variants is explained in Section 4. Simulation results on benchmark problems and engineering optimization problems are presented in Section 5. Last but not the least, major conclusions of this study have been presented in Conclusion.

2. Chaotic Map

In this section, the definitions of different chaotic maps are presented. Table 1 shows the definition range and names of the chaotic maps. These maps have also been studied in the approaches [22, 28]. The shape of these maps with starting point $x_k = 0.7$ is shown in Figure 1.

3. Grasshopper Optimisation Algorithm: An Overview

Grasshopper Optimisation Algorithm (GOA) [16] is a recently proposed naturally inspired algorithm, which is based on one of the largest swarms of all creatures. As grasshoppers are herbivores, they cause severe damage to crops. The swarming behavior of a grasshopper depends on both

Table 1: Definition of chaotic maps [28].

Name of map	Equation	Range
Chebyshev	$x_{k+1} = \cos\left(k \cos^{-1}\left(x_k\right)\right)$	$(-1, 1)$
Circle	$x_{k+1} = \mathrm{mod}\{x_k + b - (a/2\pi)\ \sin\left(2\pi x_k, 1\right)\}, \quad a = 0.5,\ b = 0.2$	$(0, 1)$
Gauss	$x_{k+1} = \begin{cases} 1, & \text{if } x_k = 0 \\ \left(1/\mathrm{mod}\left(x_k, 1\right)\right), & \text{otherwise} \end{cases}$	$(0, 1)$
Iterative	$x_{k+1} = \sin\left(a\pi/x_k\right), \quad a = 0.7\ \ (\pi \approx 3.14)$	$(-1, 1)$
Logistic	$x_{k+1} = a x_k\left(1 - x_k\right), \quad a = 4$	$(0, 1)$
Piecewise	$x_{k+1} = \begin{cases} \left(x_k/P\right), & 0 \le x_k < P \\ \left(\left(x_k - P\right)/\left(0.5 - P\right)\right), & P \le x_k < 0.5 \\ \left(\left(1 - P - x_k\right)/\left(0.5 - P\right)\right), & 0.5 \le x_k \le 1 - P \\ \left(\left(1 - x_k\right)/P\right), & 1 - P \le x_k < 1 \end{cases} \quad P = 0.4$	$(0, 1)$
Sine	$x_{k+1} = (a/4)\sin\left(\pi x_k\right), \quad a = 4$	$(0, 1)$
Singer	$x_{k+1} = \mu\left(7.86 x_k - 23.31 x_k^2 + 28.75 x_k^3 - 13.302875 x_k^4\right), \quad \mu = 2.3$	$(0, 1)$
Sinusoidal	$x_{k+1} = a x_k^2 \sin\left(\pi x_k\right), \quad a = 2.3$	$(0, 1)$
Tent	$x_{k+1} = \begin{cases} \left(x_k/0.7\right), & x_k < 0.7 \\ (10/3)\left(1 - x_k\right), & x_k \ge 0.7 \end{cases}$	$(0, 1)$

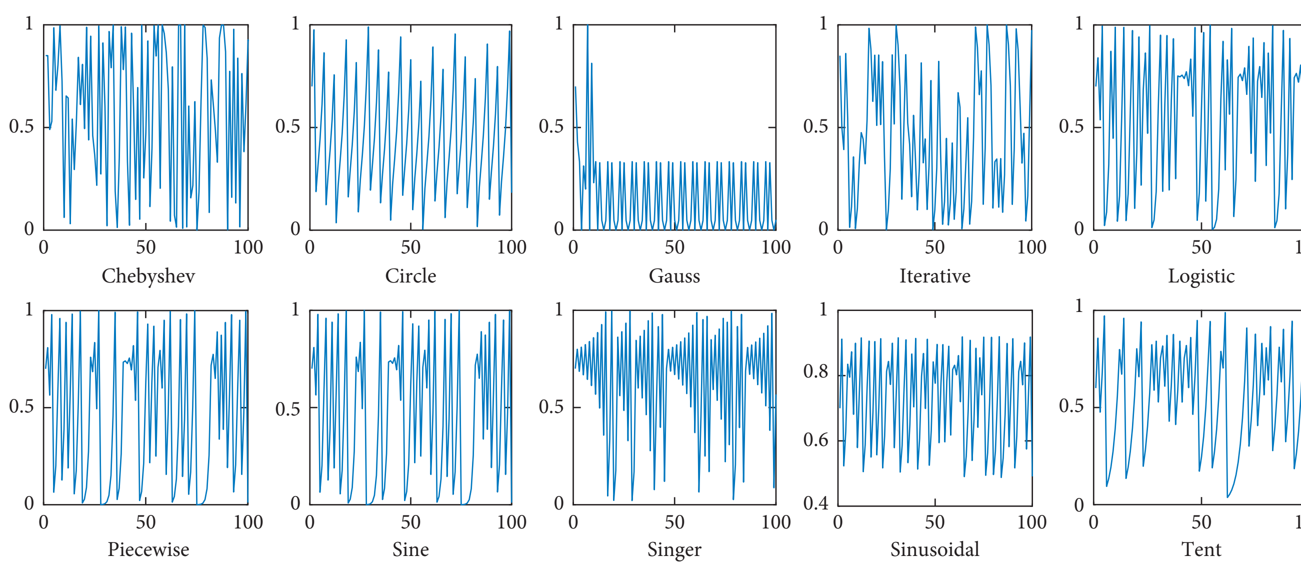

Figure 1: Chaotic maps.

nymphs and adults. The nymph moves on rolling on the ground and feeds on succulents and soft plants. An adult grasshopper can jump high in search of food and therefore have a larger area to explore. As a result, both type of movements are observed, that is, slow movement and abrupt movement of large range which represents exploration and exploitation. The mathematical frame work presented in [16] has been presented here. The swarming behavior of the grasshopper is represented mathematically as

$$X_i = S_i + G_i + A_i, \tag{1}$$

where X_i is the position of the ith grasshopper, S_i is the social interaction, G_i is the gravity force in the ith grasshopper, and A_i is the wind advection.

The social interaction S_i is given as

$$S_i = \sum_{j=1, j \ne i}^{N} s\left(d_{ij}\right) \widehat{d}_{ij}, \tag{2}$$

where $d_{ij} = |x_j - x_i|$ is the distance between the ith and jth grasshopper and $\widehat{d}_{ij} = (x_j - x_i)/(d_{ij})$ is a unit vector from the ith grasshopper to the jth grasshopper. Function s implies the social forces which can be given mathematically as

$$s(r) = f e^{(-r/l)} - e^{-r}, \tag{3}$$

where f is the intensity of attraction and l is the attractive length scale. In the search of food, grasshoppers create three types of regions in terms of social interaction known as the comfort zone, repulsion region, and attraction region. When the distance is larger between grasshoppers, then function

"s" is not able to apply strong forces. To resolve this, the G component in (1) is given as

$$G_i = -g\widehat{e}_g, \tag{4}$$

where g is the gravitational constant and $\widehat{e}_g$ represents a unity vector towards the center of Earth. The A component is calculated as

$$A_i = u\widehat{e}_w, \tag{5}$$

where u is the constant drift and $\widehat{e}_w$ is a unity vector in the direction of wind. Substituting values of s, G, and A in (1), we get

$$X_i = \sum_{j=1, j\neq i}^{N} s\left(\left|x_j - x_i\right|\right)\frac{x_j - x_i}{d_{ij}} - g\widehat{e}_g + u\widehat{e}_w, \tag{6}$$

where $s(r)$ is given by (3) and N is the number of grasshoppers. A revised form of this formula can be used to solve optimization problem:

$$X_i^d = c\left(\sum_{j=1, j\neq i}^{N} c\frac{ub_d - lb_d}{2} s\left(\left|x_j^d - x_i^d\right|\right)\frac{x_j - x_i}{d_{ij}}\right) + \widehat{T}_d, \tag{7}$$

where ub_d is the upper bound in the Dth dimension, lb_d is the lower bound in the Dth dimension, $\widehat{T}_d$ is the value of the Dth dimension in the target, and c is the decreasing coefficient to shrink the comfort zone, repulsive zone, and attraction zone. It is assumed that the wind direction is always towards a target. In the process of searching food, nymphs move on rolling on the ground and adults move on jumping in the air, creating both the cases exploration and exploitation. One can balance both of these two by decreasing the parameter c in (8) proportionally to the number of iteration. This can be calculated as

$$c = c_{\max} - l\left[\frac{c_{\max} - c_{\min}}{L}\right], \tag{8}$$

where $c_{\max}$ is the maximum value, $c_{\min}$ is the minimum value, l indicates the current iteration, and L is the maximum number of iterations. The application of GOA has been observed in many engineering optimization problems [29, 30].

4. Development of Enhanced Chaotic Grasshopper Optimization Algorithms

This section presents philosophy and chronological development of ECGOAs. In this work, we have obeyed the philosophy of GOA and decreased the parameter (c) in due course of iterations. However, with the inculcation of the different chaotic sequences in the comfort zone reduction parameter "c," the diversification virtue of the GOA enhances till the last iteration. To develop the variants, a normalization function is employed to distribute the sequences between maximum and minimum bias before it can be biased with the parameter c. The mathematical expression for this function at any iteration l can be given as

$$N_m(l) = N_m^{\max} - \left(\frac{N_m^{\max} - N_m^{\min}}{L}\right) * l, \tag{9}$$

where L denotes the maximum iteration. The normalized chaotic sequence can be given as per following equation:

$$C(l) = N_m(l) * x_l, \tag{10}$$

where x_l is the value of chaotic sequence computed as per Table 1.

The instantaneous value of the chaotic sequence embedded parameter for ECGOA will be given as per the following equation:

$$c^{\text{ECGOA}}(l) = c^{\text{GOA}}(l) + C(l). \tag{11}$$

By using (8)–(10), one can easily get the value of the chaotic sequence for any of the chaotic maps given in Table 1. For example, we briefly present here (12)–(15) for piecewise map (ECGOA6). Further, in this paper, 10 different chaotic sequences are embedded with the mapping through a normalized function to the parameter c in GOA. In classical GOA, this parameter act as a bridging mechanism for the exploration and exploitation phase over the whole course of iterations. In the initial phase, the search agents take large steps to explore the search space in effective manner, and in later case, these steps are reduced with the help of linear decrement in the parameter c. In this work, the focus is on this linear variation with different chaotic sequences embedded through a normalized function. The major motivation to perform this experiment is to seek the possibility of better exploration and exploitation by introducing the chaotic sequences in each iteration. In GOA, this parameter decreases linearly, which means that algorithm either performs diversification (exploration) or intensification (exploitation). In this work, the authors have changed the parameter c chaotically so that the exploration virtue can be kept alive in the final steps of iterations.

For justification, an implementation of a logistic chaotic map with the abovementioned procedure is shown in Figure 2. On the basis of this mathematical procedure, 10 different variants of chaotic algorithms are proposed here which are named as Enhanced Chaotic Grasshopper Optimization Algorithms (ECGOAs): the different variants are ECGOA1 with Chebyshev map, ECGOA2 with Circle map, ECGOA3 with Gauss map, ECGOA4 with Iterative map, ECGOA5 with Logistic map, ECGOA6 with Piecewise map, ECGOA7 with Sine map, ECGOA8 with Singer map, ECGOA9 with Sinusoidal map, and ECGOA10 with Tent map.

Case 1: if $0 \le x_l < P$,

$$\begin{aligned} c^{\text{ECGOA}(6)}(l) &= \left(c_{\max} + N_m^{\max}\frac{x_l}{P}\right)\left(1 - \frac{l}{L}\right) \\ &\quad + \frac{l}{L}\left(c_{\min} + N_m^{\min}\frac{x_l}{P}\right). \end{aligned} \tag{12}$$

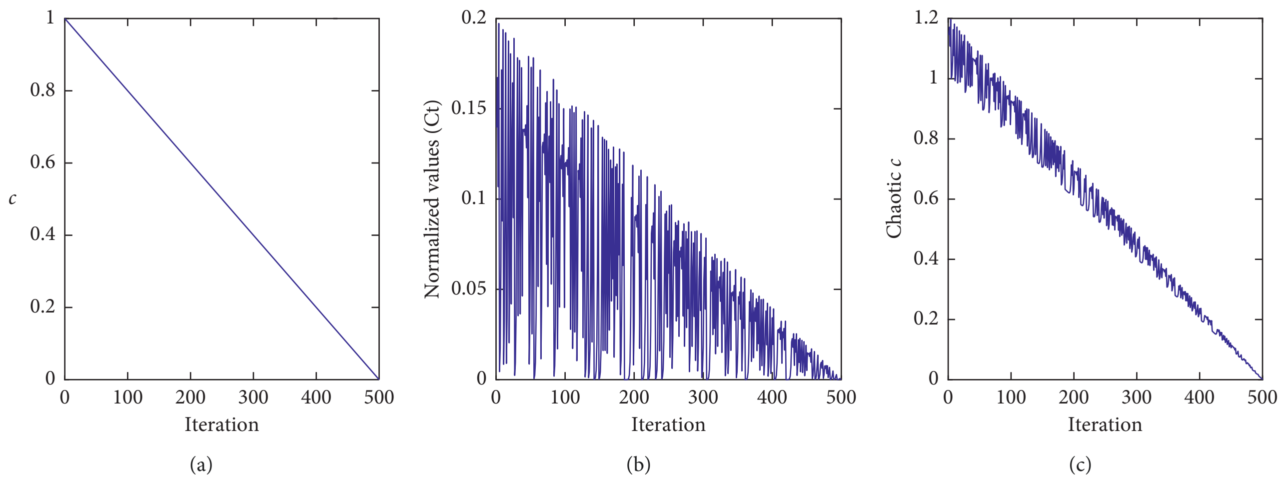

Figure 2: Development of chaotic bridging mechanism.

Case 2: if $P \le x_l < 0.5$,

$$c^{\mathrm{ECGOA}(6)}(l) = \left[c_{\max} + N_m^{\max}\left(\frac{x_l - P}{0.5 - P}\right)\right]\left(1 - \frac{l}{L}\right) + \frac{l}{L}\left[c_{\min} + N_m^{\min}\left(\frac{x_l - P}{0.5 - P}\right)\right]. \quad (13)$$

Case 3: if $0.5 \le x_l < 1 - P$,

$$c^{\mathrm{ECGOA}(6)}(l) = \left[c_{\max} + N_m^{\max}\left(\frac{1 - P - x_l}{0.5 - P}\right)\right]\left(1 - \frac{l}{L}\right) + \frac{l}{L}\left[c_{\min} + N_m^{\min}\left(\frac{1 - P - x_l}{0.5 - P}\right)\right]. \quad (14)$$

Case 4: if $1 - P \le x_l < 1$,

$$c^{\mathrm{ECGOA}(6)}(l) = \left[c_{\max} + N_m^{\max}\frac{(1 - x_k)}{P}\right]\left(1 - \frac{l}{L}\right) + \left(\frac{l}{L}\right)\left[c_{\min} + N_m^{\min}\frac{(1 - x_k)}{P}\right]. \quad (15)$$

Parameter "c" is an important parameter of GOA and used twice in (7), and the inner "c" contributes to shrink the attraction and repulsion zones between grasshoppers. This effect is analogous to the exploitation phase mechanism. However, with the increment in the iteration counter, outer c reduces the search and helps algorithm to converge. For balancing the intensification and diversification processes in GOA, parameter c decreases linearly with every passing iteration. The comfort of grasshoppers is reduced with every iteration by varying the parameter c from 1 to zero linearly. However, in the proposed ECGOAs, chaotic sequence changes the boundary of the comfort zone randomly in monotonically decreasing trend. This mechanism assists the search agents to release themselves from the local minima trap. The transition from the diversification phase to the intensification phase can be achieved slowly with the employment of a different chaotic sequences-enabled adaptive approach. This change makes parameter "c" adaptive and random concurrently. The values for $N_m^{\min}$ and $N_m^{\max}$ are considered as $1e-10$ and 0.2, respectively. In the following section, benchmarking of these variants and application of these variants on two engineering problems are investigated.

5. Simulation and Results

The testing of the optimization problem on some known functions is the best way to showcase the efficacy of the algorithm. Some of the essential characteristics of these functions are that the functions should be multimodal or unimodal in nature, the function should be nonseparable, and moreover, the functions should lag in the global structure. By keeping these virtues in consideration, benchmarking of the variants is done on five unimodal and five multimodal shifted and biased benchmark functions [31–33]. In the standard benchmark functions, the minima lies at zero; however, in multimodal functions, multioptima (local) can exist. To make the problem harder, shift and bias have been provided to the functions so that the robustness of variants can be tested. Figure 3 shows the 2D version of these functions and definition, and other relevant details of these functions are given in Table 2.

The results of the proposed variants on the unimodal functions are shown in Tables 3 and 4 for 30 dimensions and 50 dimensions; similarly the results on the multimodal benchmark problems are shown in Tables 5 and 6 for 30 and 50 dimensions, respectively. For making the analysis meaningful, four different statistical parameters, namely, standard deviation (SD), maximum value (Max), minimum value (Min), and Mean value parameters are calculated. The stopping criterion for these variants along with GOA is the maximum iteration which is set to 500. Each variant is tested on all the ten benchmark functions, and the results

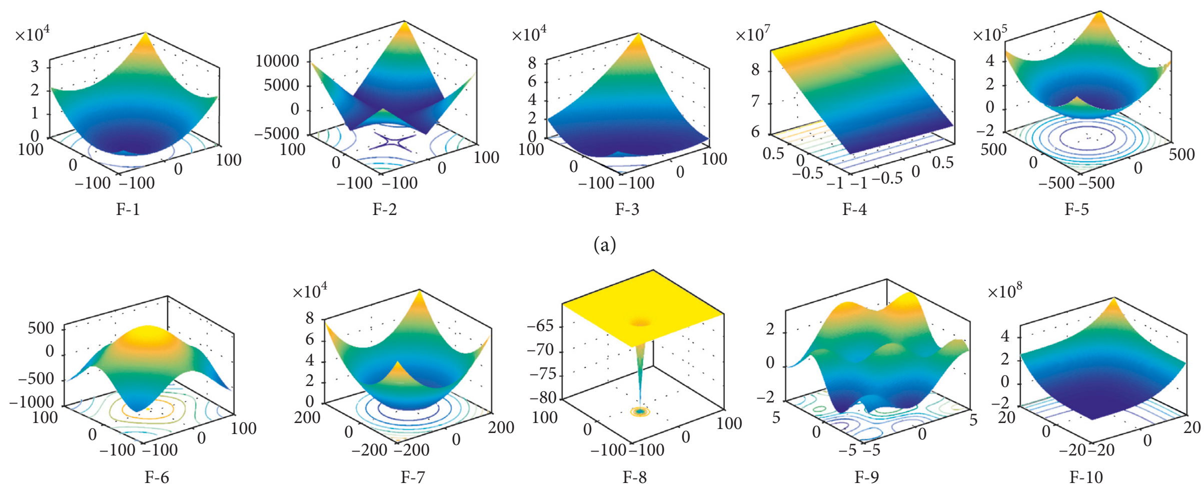

FIGURE 3: Shifted and biased benchmark functions. (a) Unimodal and (b) multimodal test functions.

TABLE 2: Benchmark functions [28–31].

Function	Dimension	Range	Minimum value
Unimodal benchmark function			
$F_1(x) = \sum_{i=1}^{n} (x_i + 30)^2 - 50$	30	[−100, 100]	−50
$F_2(x) = \sum_{i=1}^{n} \lvert x_i + 10\rvert + \prod_{i=1}^{n} \lvert x_i + 10\rvert - 50$	30	[−10, 10]	−50
$F_3(x) = \sum_{i=1}^{n} (\sum_{j-1}^{n} x_j + 30)^2 - 50$	30	[−100, 100]	−50
$F_4(x) = \sum_{i=1}^{n-1} [100((x_{i+1} + 60) - (x_i + 60)^2)^2 + ((x_i + 60) - 1)^2] - 50$	30	[−30, 30]	−50
$F_5(x) = \sum_{i=1}^{n-1} ([(x_i + 60) + 0.5])^2 - 80$	30	[−100, 100]	−80
Multimodal benchmark function			
$F_6(x) = \sum_{i=1}^{n} -(x_i + 300)\sin(\sqrt{\lvert(x_i + 300)\rvert})$	30	[−500, 500]	−418.9829 × (32)
$F_7(x) = \sum_{i=1}^{n} [(x_i + 2)^2 - 10\cos(2\pi((x_i + 20) + 2) + 10)] - 50$	30	[5.12, 5.12]	−50
$F_8(x) = -20\exp(-0.2\sqrt{(1/n)\sum_{i=1}^{n}(x_i + 20)^2}) - \exp((1/n)\sum_{i=1}^{n}\cos(2\pi(x_i + 20))) + 20 + e - 80$	30	[−32, 32]	−80
$F_9(x) = (1/4000)\sum_{i=1}^{n} (x_i + 400)^2 - \prod_{i=1}^{n} \cos((x_i + 400)/(\sqrt{i})) + 1 - 80$	30	[−600, 600]	−80
$F_{10}(x) = (\pi/n)\{10\sin(\pi y_1) + \sum_{i=1}^{n} (y_i - 1)^2[1 + 10\sin^2(\pi y_{i+1})] + (y_n - 1)^2\} + \sum_{i=1}^{n} u((x_i + 30), 10, 100, 4) - 80$	30	[−50, 50]	−80
where $y_i = 1 + (((x_i + 30) + 1)/(4))$, $u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m, & x_i > a \\ 0, & -a < x_i < a \\ k(-x_i - a)^m, & x_i < -a \end{cases}$			

are averaged over 20 independent runs. The following subsection presents the results of unimodal benchmark functions.

5.1. Qualitative Results and Discussions of Unimodal Benchmark Problems. Unimodal functions are the functions which have no local minima, or in other words, they possess only one minima. These functions are suitable for benchmarking the exploitation quality and convergence speed of any algorithm. This section presents the results on unimodal benchmark problems for 30 dimensions and 50 dimensions.

5.1.1. Simulation Results of 30-D Unimodal Benchmark Problems. The chaotic variants are benchmarked for the exploitation of quality and convergence properties on unimodal benchmark functions. The results for 30 and 50 dimensions are shown in Tables 3 and 4, respectively. For the unimodal functions, that is, F_1, F_2, and F_3, it is observed that, as per maximum values obtained for each variant, the lowest maximum value of function 1 is for ECGOA8 ($1.68E+03$), and the mean value and the standard deviation value for this variant are also lowest, that is, $3.60E+02$ and 480.17. The convergence properties of this variant for function 1 is shown in Figure 4. It is observed that convergence properties of this variant is superior to others. Similarly, for functions 2

Table 3: Results and comparison of ECGOAs with GOA (30-D) on unimodal functions.

Algorithm	Statistical parameters	F_1	F_2	F_3	F_4	F_5
ECGOA1	Max	$4.72E+03$	7.441252	32384.27	5997.771	11547.22
	SD	1189.661	20.48785	8687.739	1448.595	2648.273
	Mean	$9.54E+02$	−34.0542	13561.55	519.9767	1515.979
	Min	−33.212	−50	3212.748	**−23.8636**	−49.6495
ECGOA2	Max	$3.67E+03$	6.983289	27021.89	**774.8997**	7411.605
	SD	1070.123	15.60652	7021.802	**235.0562**	2190.903
	Mean	$8.65E+02$	−33.8791	14337.61	**67.54067**	1770.953
	Min	−49.6441	−50	3949.875	−22.9774	−46.6215
ECGOA3	Max	$2.64E+03$	−10	23806.04	359978.7	4481.247
	SD	806.0854	14.47655	6705.574	80425.93	1249.42
	Mean	$5.10E+02$	−32.2934	11977.49	18315.61	891.0015
	Min	−49.7058	−50	**1076.156**	−23.1032	−45.3564
ECGOA4	Max	$1.70E+03$	7.013903	**22406.58**	5997.795	3592.651
	SD	536.6934	15.66702	**5003.504**	2352.145	855.4461
	Mean	$5.19E+02$	−30.1622	11631.74	1257.859	479.4171
	Min	−44.7163	−50	4640.106	−22.0574	−44.2489
ECGOA5	Max	$5.67E+03$	−10.9219	48674.12	359978.5	6140.426
	SD	1268.831	14.3114	11345.15	80389.73	1753.316
	Mean	$8.38E+02$	−38.4507	15310.28	18486.14	1587.872
	Min	−40.7524	−50	4158.884	−22.8343	**−49.772**
ECGOA6	Max	$4.56E+03$	27.11865	34182.77	359978.7	9207.069
	SD	1504.872	19.65367	7717.328	80359.56	2300.953
	Mean	$1.26E+03$	−35.3059	14001.46	18656.76	1582.863
	Min	**−49.5997**	−50	3692.072	−22.1018	−49.5038
ECGOA7	Max	$4.63E+03$	**−30**	25403.45	774.9623	9922.265
	SD	1378.628	**8.3955**	5469.37	333.3362	2460.44
	Mean	$1.21E+03$	**−44.166**	13289.92	179.7019	1421.978
	Min	−45.0917	**−50**	5563.239	−22.918	−49.0028
ECGOA8	Max	**$1.68E+03$**	5.974824	29394.74	359978.8	**1992.826**
	SD	**480.1784**	16.12038	5952.928	80354.38	**486.393**
	Mean	**$3.60E+02$**	−34.3295	12586.33	18679.3	**411.1577**
	Min	−49.5353	−50	4302.044	−23.1203	−49.105
ECGOA9	Max	$6.40E+03$	27.15447	35257.01	263688.6	8310.101
	SD	1522.377	20.47155	8866.673	58719	1997.682
	Mean	956.9384	−34.7665	15903.13	14423.59	1573.284
	Min	−29.154	−50	2960.342	−22.7203	7.090624
ECGOA10	Max	$7.43E+03$	6.999618	28524.88	3813.648	4410.677
	SD	1720.647	15.64801	6605.129	1176.846	1183.068
	Mean	1042.276	−35.9717	**10828.61**	360.635	1071.854
	Min	−30.8355	−50	1853.837	−23.0794	41.99444
GOA [16]	Max	$6.52E+03$	−10	23228.38	5997.962	3775.516
	SD	1578.947	11.67419	5972.024	1329.84	1181.988
	Mean	1276.143	−38.4728	12189.96	554.6474	898.7971
	Min	−43.7533	−50	1178	−22.9119	−49.7339

and 3, the algorithms ECGOA7 and ECGOA4 have the minimum standard deviation values. The differences in the minimum, maximum, standard deviation, and mean values are very marginal with other variants. Similarly, for function 4, ECGOA2 performs better than other variants as the values of three statistical parameters out of four are minimum. For function 5, again ECGOA8 provides better results as per the obtained Max, SD, and Mean values. The solution of function 1 by ECGOA8 is shown in Figure 5.

5.1.2. Simulation Results of 50-D Unimodal Benchmark Problems. Further the analysis is carried out on 50-D unimodal functions. The results of all the developed variants with GOA on unimodal benchmark functions are shown in Table 4. From the careful inspection of the results, it is observed that for function 1, ECGOA8 possesses minimum values of statistical parameter Min. However, other parameters, namely, Max and Mean are low for ECGOA3. For function 2, the values of SD and Mean are optimal for ECGOA8. Hence, it can be concluded that this variant outperforms others for this particular function. For functions 3, 4, and 5, ECGOA9, ECGOA1, and ECGOA2 possess optimal mean values. From this analysis, it can be concluded that the exploitation capability of GOA has been

Table 4: Results and comparison of ECGOAs with GOA (50-D) on unimodal functions.

Algorithm	Statistical parameters	F_1	F_2	F_3	F_4	F_5
ECGOA1	Max	$2.91E+04$	**32.18483**	111797.5	3033130912	$3.24E+04$
	SD	4580.993499	24.20682	20978.46	**3033130912**	**4894.9252**
	Mean	$2.03E+04$	6.261021	62956.04	**3033130912**	21270.264
	Min	12525.67662	−31.5255	30301.64	−1.53368951	13898.103
ECGOA2	Max	$3.48E+04$	106.5421	103711.7	2154795033	**29074.924**
	SD	5707.583776	45.64063	18934.69	486469119.6	5100.4422
	Mean	$2.26E+04$	14.83906	68327.38	138349178.8	**19007.027**
	Min	15420.5153	**−50**	38549.12	−1.24711837	10868.143
ECGOA3	Max	$\mathbf{2.85E+04}$	69.54283	113569.7	**1206254731**	31566.385
	SD	4356.900319	31.4644	18027.51	393145232.4	5251.3516
	Mean	$\mathbf{2.00E+04}$	14.19818	63156.77	189714077.2	23529.016
	Min	14758.56528	−32.9833	40865.46	**−4.86134464**	15436.59
ECGOA4	Max	$3.22E+04$	90	104517.8	2346293923	35774.413
	SD	**4196.200051**	33.22862	20597.45	669971386.6	5791.7485
	Mean	$2.10E+04$	7.25466	59656.25	261272209.6	22008.399
	Min	15427.28384	**−50**	**24785.01**	−1.42685917	15822.568
ECGOA5	Max	$3.15E+04$	106.4788	109516.7	2513564142	36989.27
	SD	5903.244549	32.80914	23233.04	857624011.7	6199.7488
	Mean	$2.13E+04$	8.691083	64561.99	558000246.9	23826.641
	Min	14246.34036	**−50**	30868.1	−2.94514106	13771.658
ECGOA6	Max	$2.87E+04$	70	85934.37	2670344909	36744.052
	SD	5121.771356	27.96234	15565.56	872997453.1	5968.3631
	Mean	$2.04E+04$	16.29936	57717.67	397263587.7	24972.077
	Min	10043.07676	**−50**	25295.89	−1.43401992	16032.898
ECGOA7	Max	$3.47E+04$	48.67325	95502.4	2284274161	32038.771
	SD	6057.849926	27.39903	17070.54	829809398.2	4873.0274
	Mean	$2.29E+04$	4.364473	60573.92	417129473.9	21394.938
	Min	9880.378913	**−50**	30157.06	−2.83394271	11814.998
ECGOA8	Max	$3.15E+04$	47.85565	137403.3	2664314422	37842.546
	SD	5462.667635	**22.65879**	26818.08	895484803.7	6337.3582
	Mean	$2.16E+04$	**2.030997**	67721.54	410479727.9	23106.754
	Min	**8900.984581**	−31.5151	32203.72	−1.36190421	14299.857
ECGOA9	Max	$3.45E+04$	106.6678	102734.8	2841946390	30316.017
	SD	5138.922636	38.41419	21197.83	765285734.9	4717.0682
	Mean	21536.60375	7.865827	56321.14	356879478.5	23646.126
	Min	11155.93754	**−50**	26036.76	−1.46623175	13613.975
ECGOA10	Max	$3.03E+04$	50	122011.5	2270930776	33079.78
	SD	5296.153654	24.70026	18910.63	618827002.9	4896.2693
	Mean	21341.72611	9.529229	69260.53	288246084.2	22333.503
	Min	14745.05282	−30	36566.91	−1.44162177	13425.011
GOA [16]	Max	$3.55E+04$	86.94913	**84051.88**	3011759319	29750.024
	SD	5820.921429	37.35558	**14837.63**	925046677.7	4977.283
	Mean	22165.10941	10.92968	56877.2	525116848.9	21473.229
	Min	13433.88239	−46.8306	37391.19	−1.44722832	**9884.174**

substantially improved by the chaotic comfort zone function adaptation.

5.2. Qualitative Results and Discussions on Multimodal Benchmark Problems. In this section, experiments are carried out on multimodal functions. The results for both 30-D and 50-D problems are shown in Tables 5 and 6. Multimodal functions are those functions which possess one global minima and can have several local minima. The nature of these functions is used to benchmark the exploration quality of the proposed variants. This benchmarking exhibits the ability of the variants to search a global optimum, in such a challenging environment, where the probability of getting trapped in a local optimum is high. The bias and shift in conventional multimodal benchmark problems make the functions more complex and suitable for benchmarking the variants for the real-world engineering problems.

5.2.1. Simulation Results of 30-D Multimodal Benchmark Problems. Inspecting the results of the multimodal functions in Table 5, it is observed that ECGOA8 performs better as compared to other variants for function 6, as the value of the SD is low. It is evident to say that the values of these statistical parameters can be a meaningful indicator

Table 5: Results and comparison of ECGOAs with GOA (30-D) for multimodal functions.

Algorithm	Statistical parameters	F_6	F_7	F_8	F_9	F_{10}
ECGOA1	Max	−8904.24	271.0372	−66.5449	−29.2767	3279130
	SD	1254.676	51.23868	4.119399	13.23356	884472.8
	Mean	−11029.5	146.0832	−78.6616	−51.4071	1238399
	Min	−13826.5	66.54498	−80	−69.8688	−66.7927
ECGOA2	Max	−8832.74	229.4997	−66.7126	−28.9994	3958975
	SD	1142.206	52.12368	2.975594	11.40269	1072203
	Mean	−10745.6	145.3066	−79.171	−47.2001	1923464
	Min	−13627.7	**15.85951**	−80	−69.4254	−68.3447
ECGOA3	Max	−8376	**222.1649**	−64.3491	**−39.1735**	**2999948**
	SD	1527.962	47.29322	3.493484	**10.75939**	1038006
	Mean	−11293.5	164.0275	−78.9705	**−60.5568**	1442602
	Min	−14534.5	56.96255	−80	**−78.9487**	−66.9631
ECGOA4	Max	−8307.93	256.214	−78.3538	−19.0129	3999958
	SD	1109.739	57.38026	**0.506681**	12.7704	1292593
	Mean	−11006.9	**141.5465**	−79.6353	−51.0195	1278308
	Min	−12818.6	53.7245	−80	−69.8415	−72.994
ECGOA5	Max	−8540.51	243.5495	−60.3452	−28.1305	4922217
	SD	1220.617	48.63998	5.114423	13.8308	1398789
	Mean	−11059.9	146.0103	−78.1109	−47.6149	1688204
	Min	−13079.1	54.99126	−80	−69.8507	**−75.9006**
ECGOA6	Max	−8426.57	249.5214	−60.0117	−39.1139	2999975
	SD	1345.302	44.48447	5.512268	11.35779	913600.2
	Mean	−10663.5	147.7425	−78.1364	−54.092	**758313.7**
	Min	−13323.2	72.3563	−80	−78.4344	−72.602
ECGOA7	Max	−7639.37	244.7659	−60.0035	−9.16557	3742472
	SD	1637.129	40.17972	4.45385	15.04767	1044970
	Mean	−10769.7	144.7726	−78.7532	−51.0586	1140919
	Min	**−15549.9**	79.90368	−80	−69.5569	−73.8642
ECGOA8	Max	−8815.14	238.5191	**−78.3538**	−29.0018	3999953
	SD	**972.5037**	**39.69233**	0.675579	11.76663	1258635
	Mean	−11237	163.4965	**−79.6707**	−50.8158	1575727
	Min	−12712	90.96168	**−80**	−69.9456	−68.8842
ECGOA9	Max	−7711.69	250.473	−78.3538	−19.8782	2999949
	SD	1198.213	40.70405	0.675578	12.726	**778483.5**
	Mean	−10681.1	167.9705	−79.6707	−41.31	1095574
	Min	−12474.2	96.95393	−80	−69.4142	−64.8591
ECGOA10	Max	−8746.85	269.3226	−60.1722	−19.3277	5249683
	SD	1190.82	41.32749	4.41648	12.70165	1582795
	Mean	−10661.7	161.4352	−78.7616	−43.6748	1394908
	Min	−13545.7	84.36102	−80	−69.0206	−61.9252
GOA [16]	Max	**−9225.8**	258.3774	−60.0393	−20.1744	3000918
	SD	1292.871	50.35495	8.248127	12.85493	975411.2
	Mean	**−11237.6**	161.2432	−75.4165	−57.6119	1122615
	Min	−14022.9	76.48267	−80	−69.8943	−69.4996

to judge the performance of variants. For function 6, the standard deviation value for ECGOA8 is 972.50 as compared with other variants. The standard deviation of GOA for this function is 1292.87. For function 7, this variant also performs better as compared to others as the standard deviation value for this variant is the lowest, that is, 39.69, and for GOA, it is 50.35. From the results, it can be concluded that ECGOA8 (Singer map-enabled chaotic mechanism) provides better results for first three multimodal functions. The convergence characteristics for function 7 are plotted in Figure 6. For function 8, again this variant shows promising results as the three out of 4 parameters are low as compared to other variants. For function 9, ECGOA3 performs better as the parameters associated with the judgement attain lower values. For function 10, ECGOA6 has least mean values. Hence, it can be concluded that for most of the functions these variants outperform GOA.

5.2.2. Simulation Results of 50-D Multimodal Benchmark Problems. The results of this experiment are shown in

Table 6: Results and comparison of ECGOAs with GOA (50-D) for multimodal functions.

Algorithm	Statistical parameters	F_6	F_7	F_8	F_9	F_{10}
ECGOA1	Max	−13284.9	525.3938	−59.4202	1142.297	$5.99E+08$
	SD	1468.401	61.31428	0.128399	376.3986	179942439
	Mean	−16012.5	441.7274	−59.6647	474.3385	130982138.6
	Min	−18346	296.6225	−59.8477	80.21982	9945376.637
ECGOA2	Max	−13222.8	568.819	−59.4939	1225.204	518183195.7
	SD	1284.353	59.19765	0.114828	380.9211	161215153.4
	Mean	−15729.6	431.9621	−59.682	526.118	111311331.7
	Min	−17918.7	321.537	−59.9695	92.68036	8548635.153
ECGOA3	Max	−12054.8	615.8589	−59.4566	1189.236	717878837.1
	SD	1475.55	70.58729	0.119339	325.6841	234394118.5
	Mean	−15601.9	455.3608	−59.6446	488.7296	203256596.5
	Min	−18405	313.2195	−59.8967	109.8307	8852551.366
ECGOA4	Max	−12155.6	**486.4432**	−59.4112	1128.317	802793748.5
	SD	**1175.43**	50.69295	0.12958	343.5583	260492546.8
	Mean	−15259.2	**410.6172**	−59.6654	417.0988	194442446.2
	Min	−16864.5	316.0931	−59.8891	77.3793	7577441.415
ECGOA5	Max	−13473.6	536.6068	−59.4432	1395.387	506974142.9
	SD	1503.746	**49.9885**	0.14537	383.249	146280518.5
	Mean	−15499.8	429.7536	−59.7203	523.472	101913643.5
	Min	−18340.6	**353.849**	−60.0317	96.59476	8061427.882
ECGOA6	Max	−12379	561.8097	−59.4954	1562.137	622102603.5
	SD	1635.44	75.40271	0.115495	390.2075	201123569.1
	Mean	−15574.9	431.8302	−59.6764	381.6959	152537664.3
	Min	−18784	278.6123	−59.9099	**72.10621**	**5058058.625**
ECGOA7	Max	−13522.3	611.5379	**−59.5858**	**924.5556**	1019814442
	SD	1291.999	81.91863	**0.083408**	**253.7802**	296529381.7
	Mean	**−16657.1**	442.2678	−59.7265	**360.7756**	207773478.8
	Min	**−19647.9**	283.6656	−59.8738	112.8999	6189273.462
ECGOA8	Max	−12906.6	522.0567	−59.4512	1035.483	469635197.1
	SD	1467.799	53.46532	0.15148	291.7955	156859406.9
	Mean	−15172.3	422.7514	−59.6563	401.8186	111764679.8
	Min	−18254.9	290.0732	−60.062	96.79589	5845757.794
ECGOA9	Max	−13639.7	543.6966	−59.3504	1138.901	**434575957.9**
	SD	1417.122	80.27476	0.119305	367.9628	173595240.7
	Mean	−15904.4	426.5996	−59.6111	457.5753	130924283.2
	Min	−18462.9	262.1963	−59.8748	94.65694	6641433.378
ECGOA10	Max	−13343.5	580.2786	−59.3751	940.1151	601504311
	SD	1218.772	56.89642	0.144239	286.1847	172390624.5
	Mean	−15770	460.4834	**−59.7354**	336.8918	119948547.1
	Min	−18108	365.7742	**−60.151**	96.49957	11133246.78
GOA [16]	Max	**−13843.7**	556.5219	−59.4587	1254.326	524047875.8
	SD	1266.274	61.51136	0.142779	341.6776	**125063391.3**
	Mean	−16059.7	446.7562	−59.6615	528.9932	**60674853.38**
	Min	−18377.6	283.9794	−60.076	113.8766	8321958.122

Table 6. For function 6, the optimal value for parameter SD is attained for ECGOA4, and Mean and Min for ECGOA7; for function 9, the parameters Max, SD, and Mean attain optimal values for ECGOA7 variant. The optimal values of the statistical parameters are shown in boldface. Inspecting the results of this experiment on the multimodal functions, it is clearly evident that the exploration capability of the variants is enhanced substantially by employing the chaotic comfort zone functions. To judge the significance of the results, authors performed the Wilcoxon rank sum test [34] with 5% confidence interval. The test results (p values) are shown in Tables 7 and 8.

5.2.3. Discussion. Parameter c is an important parameter and acts as a bridging mechanism between exploration and exploitation phases. This parameter ensures the swift movement of grasshoppers from exploration phase to exploitation phase by reducing the comfort zone of grasshoppers. Chaotic mechanism not only enhances the exploitation phase by keeping alive the virtue of exploration till the last iteration but also adds random behavior with each iteration on the basis of different adaptive chaotic comfort zone function-enabled mechanisms. Further, the following section presents the statistical analysis of the performance of these variants.

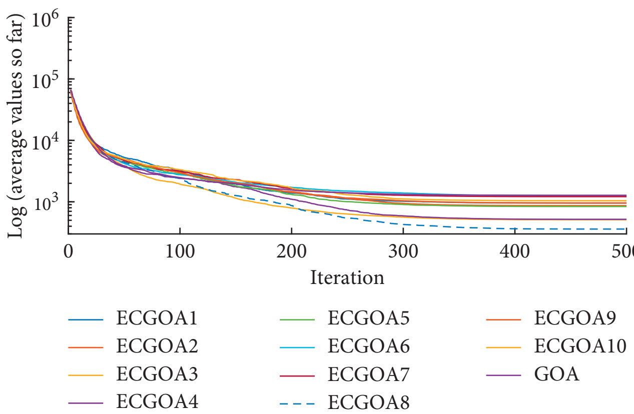

Figure 4: Convergence curve for function 1 (30-D).

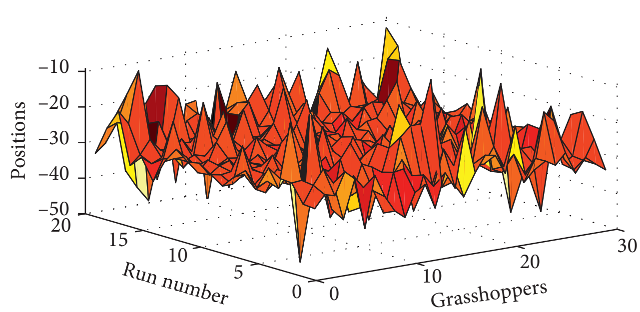

Figure 5: Results of ECGOA8 for unimodal function 1.

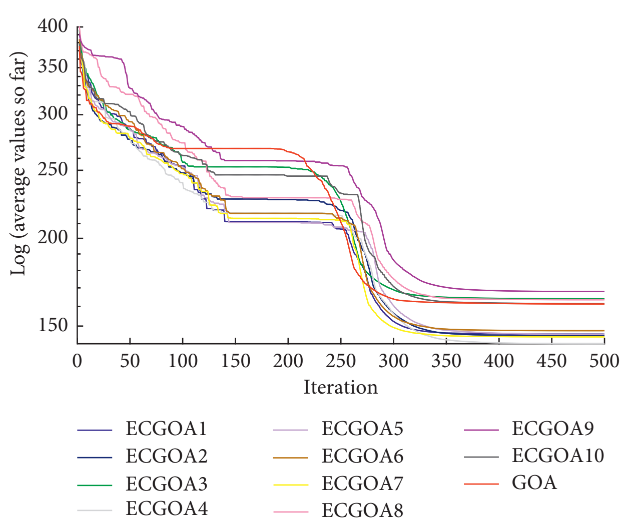

Figure 6: Convergence curve for multimodal function 7 (30-D).

5.3. Wilcoxon Rank Sum Test. To judge the significance of the results, the Wilcoxon rank sum test [34] is performed with 5% significance interval, and the p values are obtained. The test results are shown in Tables 7 and 8. The term N/A indicates that the variant has outperformed over others and cannot be compared with itself. From the results of function 1 (Table 7), it is observed that there is a significant difference between ECGOA1 and ECGOA7 as compared with ECGOA8 as the p values obtained for these variants are less than 0.05. Similarly, for function 3, ECGOA4 is the best algorithm; however, in this category, all the algorithms are statistically not different from each other as the p values are greater than 0.05. It is empirical to observe that the lowest p value other than ECGOA4 is for ECGOA9. For function 2, it is observed that ECGOA7 is the best performer and a significant difference exists between ECGOA2, 3, 4, 8, and 10 variants as the p values are less than 0.05. The p values which are less than 0.05 are highlighted in boldface and underlined.

Table 7: Results of the Wilcoxon rank sum test on unimodal benchmark functions.

Function	F_1	F_2	F_3	F_4	F_5
ECGOA1	**0.036048**	0.169275	0.967635	0.6359	0.3683
ECGOA2	0.147847	**0.013881**	0.336915	N/A	0.0540
ECGOA3	0.881731	**0.007431**	0.989209	0.8817	0.9533
ECGOA4	0.228694	**0.000986**	N/A	0.1404	0.4755
ECGOA5	0.081032	0.166588	0.524987	0.4094	0.7590
ECGOA6	0.072045	0.088317	0.409356	0.9892	0.4340
ECGOA7	**0.014364**	N/A	0.310402	0.6554	0.9423
ECGOA8	N/A	**0.04359**	0.797197	0.5792	N/A
ECGOA9	0.15557	0.14484	0.163596	0.1719	0.4971
ECGOA10	0.126431	**0.043738**	0.490334	0.3793	0.9917
GOA [16]	**0.014364**	0.07045	0.755743	**0.0036**	**0.0266**

Table 8: Results of the Wilcoxon rank sum test on multimodal benchmark functions.

Function	F_6	F_7	F_8	F_9	F_{10}
ECGOA1	0.56	0.776391	0.27	**0.03**	0.08
ECGOA2	0.2	0.542772	0.73	**8.36*E* − 04**	**8.36*E* − 04**
ECGOA3	0.81	0.14042	0.0098	N/A	**0.02**
ECGOA4	0.59	0.655361	N/A	**0.02**	0.23
ECGOA5	0.88	0.507505	**0.041**	**0.040**	**0.019**
ECGOA6	0.22	0.71498	**0.0114**	0.0909	N/A
ECGOA7	0.19	0.126431	0.525	**0.0337**	0.085
ECGOA8	0.9892	N/A	0.2503	**0.0207**	**0.02**
ECGOA9	0.23	0.113551	0.285	**5.85*E* − 06**	0.0601
ECGOA10	0.14	0.163596	0.6554	**1.99*E* − 04**	0.081
GOA [16]	N/A	0.208454	0.239	**0.0076**	0.1719

Inspecting the results of Table 8 for multimodal functions, ECGOA8 is the best performer, and for functions 4 and 5, variants ECGOA2 and ECGOA3 are the second best performers as per the p values. For function 6, it is observed that a significant difference exists between variants ECGOA3, ECGOA4, and ECGOA5. In most of the functions, chaotic variants outperform GOA and the variants are significantly different from each other. In the following section, the application of these variants on real-world problems and comparative performance of the variants with other contemporary algorithms is presented.

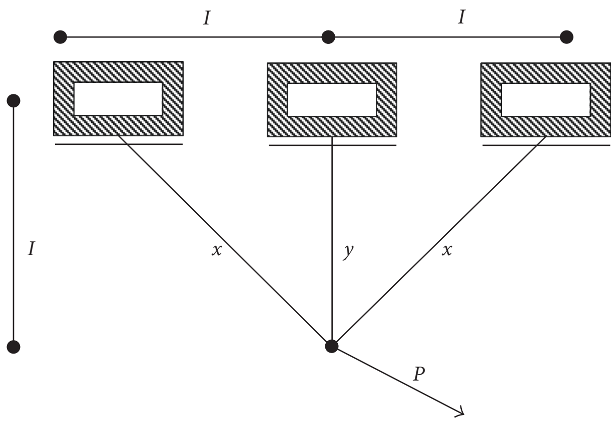

Figure 7: Three truss bar design problem.

5.4. Application of the ECGOAs on Real Applications. Application of these variants on structural design-constrained optimization problem and parameter estimation problem is investigated in this section. The impact of the chaotic bridging mechanism in reducing the comfort zone of the grasshoppers that results in the better exploration and exploitation properties of GOA are evaluated with these applications.

5.4.1. Three-Bar Truss Design Problem. Three truss bar design problem is a well-known engineering design problem and has been used for benchmarking of many problems [14–16]. A schematic diagram of this problem is shown in Figure 7. The objective of this problem is to minimize the volume (X) by adjusting the cross-sectional area (x, y) as per (16) subject to the constraints [17–19]. This objective function is nonlinear in nature and possesses three nonlinear constraints which contain the stress parameter. For solving this optimization problem, the number of search agents (30) and maximum iteration count (500) are considered and kept constant for all the variants. Each algorithm is run for 20 times and results shown in Table 9 are averaged over these runs. The convergence curve of the problem is shown in Figure 8. The expression for the volume is given as

$$\overrightarrow{X} = [x, y]. \tag{16}$$

$$\begin{aligned} \min \quad & f(\overrightarrow{X}) = (2\sqrt{2}\,x + y)l \\ & h_1(\overrightarrow{X}) = \frac{2\sqrt{2}x + y}{\sqrt{2}x^2 + 2xy}P - \sigma \leq 0, \\ \text{subject to} \quad & h_2(\overrightarrow{X}) = \frac{y}{\sqrt{2}x^2 + 2xy}P - \sigma \leq 0, \\ & h_3(\overrightarrow{X}) = \frac{1}{\sqrt{2}y + x}P - \sigma \leq 0, \end{aligned} \tag{17}$$

Various parameters for this optimization problem have been considered as $l = 100\,\text{cm}$, $P = 2\,\text{KN/cm}^2$, and $\sigma = 2\,\text{KN/cm}^2$ with variable range $0 \leq x, y \leq 1$.

Table 9: Results of three truss bar design problem.

Algorithm	Max	Mean	Min	SD
ECGOA1	266.5823	264.5394	263.8976	0.851265
ECGOA2	268.235	264.6162	263.8974	1.107169
ECGOA3	268.0221	264.5413	263.896	1.202524
ECGOA4	266.4968	264.3473	263.8961	0.76741
ECGOA5	265.6408	264.2707	263.8963	0.513983
ECGOA6	266.5492	264.4754	263.897	0.739352
ECGOA7	268.1783	264.3391	263.8971	0.936519
ECGOA8	265.4146	264.1322	263.8965	0.401496
ECGOA9	268.8148	264.3604	263.8962	1.108937
ECGOA10	266.0352	264.2575	263.8961	0.562701
GOA [16]	265.528	264.3357	263.3274	0.57364

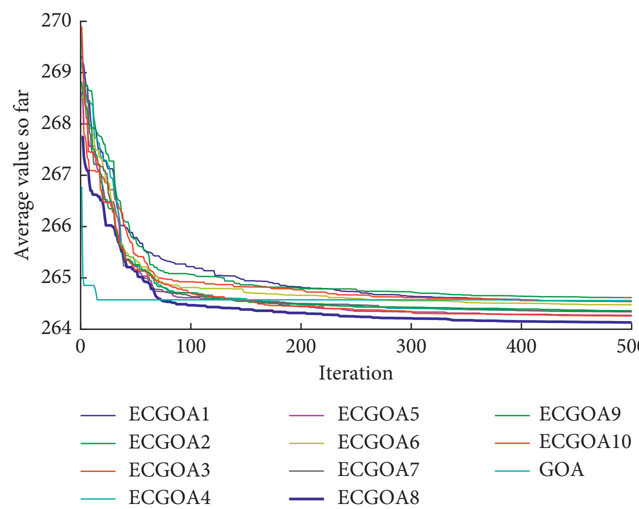

Figure 8: Convergence curve for three truss bar design problem.

The results of this problem are shown in Table 9, and it is observed that, for this design problem also, ECGOA8 variant outperforms others as the values of the standard deviation and other statistical parameters are optimal as compared with other opponents. This variant exhibits better convergence properties; for the sake of clarification, the convergence curve is shown in Figure 8.

5.4.2. Parameter Estimation for Frequency-Modulated Sound Waves. Parameter estimation of the frequency-modulated synthesizer is a six-dimensional optimization problem and a part of FM sound wave synthesis. The problem is formulated as the parameter estimation for generation of the sound as per the target sound. The problem is complex and multimodal in nature. The minima for the objective function is at zero. The parameter vector is estimated through the optimization process [35]. The vector has 6 parameters as per the following equation:

$$X = (a_1, \omega_1, a_2, \omega_2, a_3, \omega_3). \tag{18}$$

The expressions for the estimated and the target sound waves are as follows:

Table 10: Error in objective function values for frequency-modulated sound waves synthesis [35].

Algorithm	Min	Max	Mean	SD
ECGOA1	14.048	26.720	20.748	**2.754**
ECGOA2	8.416	25.560	20.266	4.067
ECGOA3	11.407	27.283	20.652	4.335
ECGOA4	**1.4E − 7**	27.280	19.863	5.439
ECGOA5	8.416	27.354	20.708	4.771
ECGOA6	8.416	26.522	19.687	5.183
ECGOA7	10.177	25.913	18.830	4.596
ECGOA8	**0.000**	26.999	19.108	5.710
ECGOA9	11.549	27.123	20.896	4.064
ECGOA10	13.393	27.462	21.266	3.890
GOA [16]	8.416	26.745	20.110	4.673
CPSOH [36–38]	3.45	42.52	27.08	60.61
GWO [13]	1.9311	**20.03**	25.1633	5.9177
TRIBES-D [38–40]	2.22	22.24	**14.68**	4.57
CGSA [28]	8.4161	24.71	17.43	4.1609
G-CMA-ES [36, 38, 39]	3.326	55.09	38.75	16.77

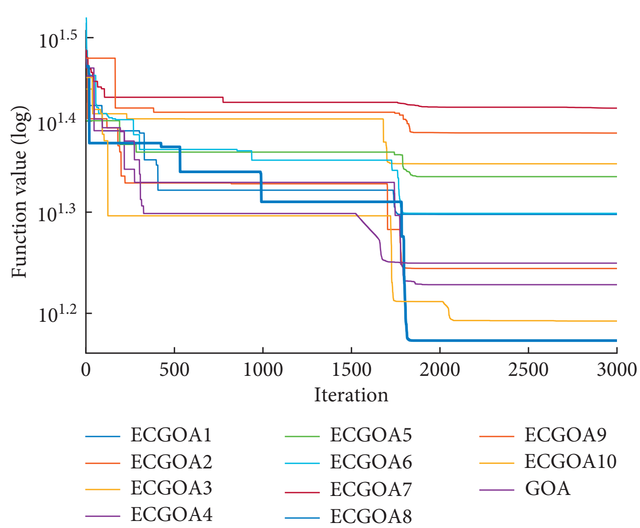

Figure 9: Convergence curve for frequency synthesis parameter estimation problem.

$$y(t) = a_1 \sin(\omega_1 t\theta + \sin(\omega_2 t\theta + \sin(\omega_3 t\theta))),$$
$$y_0(t) = (1.0)\sin((5.0)t\theta + (1.5)\sin((4.8)t\theta + (2.0)\sin((4.9)t\theta))), \quad (19)$$
$$\min\ f_2 = \sum_{t=0}^{100} (y(t) - y_0(t))^2.$$

The results of this problem are shown in Table 10, and the convergence of the variants along with GOA is shown in Figure 9. To solve this optimization problem, the maximum number of function evaluations and the number of search agents are set to be 30,000 and 30, respectively. The optimization results are averaged over 30 independent runs. It has been observed that the performance of these variants is competitive with some of the recently published approaches. ECGOA1 possesses the minimum SD parameter value as compared with others. From these applications, it can be concluded that the variants show the competitive performance not only on shifted and biased benchmark functions but also on the real applications. In the following section, the conclusions drawn from this study are presented.

6. Conclusion

Exploration and exploitation phases of a metaheuristic algorithm are connected with a bridging mechanism. The efficacy of this bridging mechanism is important to achieve better convergence characteristics, solution quality, and optimization performance. This paper focuses on this mechanism, and 10 chaotic bridging mechanisms have been proposed for GOA. Following are the major highlights of this work:

(1) 10 different chaotic maps have been embedded with the conventional GOA parameter "*c*", and the chaotic mechanism has been proposed. The benefit of this mechanism is that it enables exploration phase till last iteration with chaotic properties.

(2) Ten shifted and biased benchmark functions have been considered to benchmark the variants. The proposed variants have been evaluated on 30-dimensional and 50-dimensional benchmark problems. It has been observed that the mechanism which is enabled with the Singer chaotic map, that is, ECGOA8 is suited for unimodal and multimodal optimization problems.

(3) Further the application of these variants on three truss bar design problem and parameter estimation of the frequency-modulated sound wave synthesis problem have also been investigated. It is observed that the performance of the developed variants is competitive with other contemporary algorithms. In some cases, variants outperform.

(4) A nonparametric Wilcoxon rank sum test has been conducted, and the p values have been obtained for all the ten functions. It has been concluded that variant ECGOA8 exhibits better results as compared with other opponents.

For further studies, it would be interesting to explore the application of different comfort zone reduction functions to improve the bridging mechanism between exploration and exploitation phases of GOA.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

[1] I. Fister Jr., X.-S. Yang, I. Fister, J. Brest, and D. Fister, *A Brief Review of Nature-Inspired Algorithms for Optimization*, 2013.

[2] A. E. Eiben and J. E. Smith, *Introduction to Evolutionary Computing*, Vol. 53, Springer, Berlin, Germany, 2003.

[3] I. Rechenberg, "Evolution strategy: natures way of optimization," in *Optimization: Methods and Applications, Possibilities and Limitations*, pp. 106–126, Springer, Berlin, Germany, 1989.

[4] J. H. Holland, "Genetic algorithms," *Scientific American*, vol. 267, no. 1, pp. 66–73, 1992.

[5] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 2, pp. 82–102, 1999.

[6] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: a gravitational search algorithm," *Information Sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.

[7] O. K. Erol and I. Eksin, "A new optimization method: big bang–big crunch," *Advances in Engineering Software*, vol. 37, no. 2, pp. 106–111, 2006.

[8] A. Hatamlou, "Black hole: a new heuristic optimization approach for data clustering," *Information Sciences*, vol. 222, pp. 175–184, 2013.

[9] J. Kennedy, *Particle Swarm Optimization Encyclopedia of Machine Learning*, Springer, New York, NY, USA, 2010.

[10] X. Yang, *A New Metaheuristic Bat-Inspired Algorithm Nature Inspired Cooperative Strategies for Optimization (NICSO 2010)*, Springer, Berlin, Germany, 2010.

[11] X.-S. Yang, "Firefly algorithm, stochastic test functions and design optimisation," *International Journal of Bio-Inspired Computation*, vol. 2, no. 2, pp. 78–84, 2010.

[12] X.-S. Yang and S. Deb, "Cuckoo search via lévy flights," in *Proceedings of the World Congress on Nature & Biologically Inspired Computing (NaBIC 2009)*, pp. 210–214, Coimbatore, India, December 2009.

[13] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.

[14] A. Askarzadeh, "A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm," *Computers & Structures*, vol. 169, pp. 1–12, 2016.

[15] S. Mirjalili, "The ant lion optimizer," *Advances in Engineering Software*, vol. 83, pp. 80–98, 2015.

[16] S. Saremi, S. Mirjalili, and A. Lewis, "Grasshopper optimisation algorithm: theory and application," *Advances in Engineering Software*, vol. 105, pp. 30–47, 2017.

[17] E. Gupta and A. Saxena, "Performance evaluation of antlion optimizer based regulator in automatic generation control of interconnected power system," *Journal of Engineering*, vol. 2016, Article ID 4570617, 14 pages, 2016.

[18] L. K. Panwar, S. Reddy, and R. Kumar, "Binary fireworks algorithm based thermal unit commitment," *International Journal of Swarm Intelligence Research*, vol. 6, no. 2, pp. 87–101, 2015.

[19] L. K. Panwar, S. Reddy, A. Verma, B. Panigrahi, and R. Kumar, "Binary grey wolf optimizer for large scale unit commitment problem," *Swarm and Evolutionary Computation*, vol. 38, pp. 251–266, 2018.

[20] M. Mafarja and S. Mirjalili, "Whale optimization approaches for wrapper feature selection," *Applied Soft Computing*, vol. 62, pp. 441–453, 2018.

[21] A. Saxena and S. Shekhawat, "Ambient air quality classification by grey wolf optimizer based support vector machine," *Journal of Environmental and Public Health*, vol. 2017, Article ID 3131083, 12 pages, 2017.

[22] A. H. Gandomi, X.-S. Yang, S. Talatahari, and A. H. Alavi, "Firefly algorithm with chaos," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 1, pp. 89–98, 2013.

[23] A. H. Gandomi and X.-S. Yang, "Chaotic bat algorithm," *Journal of Computational Science*, vol. 5, no. 2, pp. 224–232, 2014.

[24] G.-G. Wang, L. Guo, A. H. Gandomi, G.-S. Hao, and H. Wang, "Chaotic krill herd algorithm," *Information Sciences*, vol. 274, pp. 17–34, 2014.

[25] A. H. Gandomi, G. J. Yun, X.-S. Yang, and S. Talatahari, "Chaos-enhanced accelerated particle swarm optimization," *Communications in Nonlinear Science and Numerical Simulation*, vol. 18, no. 2, pp. 327–340, 2013.

[26] G.-G. Wang, S. Deb, A. H. Gandomi, Z. Zhang, and A. H. Alavi, "Chaotic cuckoo search," *Soft Computing*, vol. 20, no. 9, pp. 3349–3362, 2016.

[27] M. Kohli and S. Arora, "Chaotic grey wolf optimization algorithm for constrained optimization problems," *Journal of Computational Design and Engineering*, 2017.

[28] S. Mirjalili and A. H. Gandomi, "Chaotic gravitational constants for the gravitational search algorithm," *Applied Soft Computing*, vol. 53, pp. 407–419, 2017.

[29] A. Tharwat, E. H. Houssein, M. M. Ahmed, A. E. Hassanien, and T. Gabel, "Mogoa algorithm for constrained and unconstrained multi-objective optimization problems," *Applied Intelligence*, pp. 1–16, 2017.

[30] J. Wu, H. Wang, N. Li et al., "Distributed trajectory optimization for multiple solar-powered UAVs target tracking in urban environment by adaptive grasshopper optimization algorithm," *Aerospace Science and Technology*, vol. 70, pp. 497–510, 2017.

[31] X.-S. Yang, *Test problems in optimization, Engineering Optimization: An Introduction with Metaheuristic Applications*, X.-S. Yang, Ed., John Wiley & Sons, Hoboken, NJ, USA, 2010.

[32] J. G. Digalakis and K. G. Margaritis, "On benchmarking functions for genetic algorithms," *International Journal of Computer Mathematics*, vol. 77, no. 4, pp. 481–506, 2001.

[33] M. Molga and C. Smutnicki, *Test Functions for Optimization Needs. Test Functions for Optimization Needs*, 2005.

[34] F. Wilcoxon, "Individual comparisons by ranking methods," *Biometrics Bulletin*, vol. 1, no. 6, pp. 80–83, 1945.

[35] S. Das and P. N. Suganthan, "Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems," Technical Report, Jadavpur University, Kolkata, India, 2011.

[36] J. J. Liang, A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Transactions on Evolutionary Computation*, vol. 10, no. 3, pp. 281–295, 2006.

[37] F. Van den Bergh and A. P. Engelbrecht, "A cooperative approach to particle swarm optimization," *IEEE Transactions on Evolutionary Computation*, vol. 8, no. 3, pp. 225–239, 2004.

[38] S. Gupta and K. Deep, "A novel random walk grey wolf optimizer," *Swarm and Evolutionary Computation*, 2018.

[39] A. Auger and N. Hansen, "A restart CMA evolution strategy with increasing population size," in *Proceedings of the IEEE Congress on Evolutionary Computation*, vol. 2, pp. 1769–1776, Edinburgh, UK, September 2005.

[40] J. Kumpiene, A. Lagerkvist, and C. Maurice, "Stabilization of pb- and cu-contaminated soil using coal fly ash and peat," *Environmental Pollution*, vol. 145, no. 1, pp. 365–373, 2007.