Research Article

Structured Clanning-Based Ensemble Optimization Algorithm: A Novel Approach for Solving Complex Numerical Problems

Avinash Sharma ¹, Rajesh Kumar ¹, Akash Saxena ² and B. K. Panigrahi ³

¹Malaviya National Institute of Technology, Jaipur, India

²Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur, India

³Indian Institute of Technology Delhi, New Delhi, India

Correspondence should be addressed to Avinash Sharma; avinashmnit30@gmail.com

Received 18 August 2018; Revised 10 October 2018; Accepted 4 November 2018; Published 9 December 2018

Academic Editor: Azah Mohamed

Copyright © 2018 Avinash Sharma et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

In this paper, a novel swarm intelligence-based ensemble metaheuristic optimization algorithm, called Structured Clanning-based Ensemble Optimization, is proposed for solving complex numerical optimization problems. The proposed algorithm is inspired by the complex and diversified behaviour present within the fission-fusion-based social structure of the elephant society. The population of elephants can consist of various groups with relationship between individuals ranging from mother-child bond, bond groups, independent males, and strangers. The algorithm tries to model this individualistic behaviour to formulate an ensemble-based optimization algorithm. To test the efficiency and utility of the proposed algorithm, various benchmark functions of different geometric properties are used. The algorithm performance on these test benchmarks is compared to various state-of-the-art optimization algorithms. Experiments clearly showcase the success of the proposed algorithm in optimizing the benchmark functions to better values.

1. Introduction

With the ever increasing demand for new and better utility-driven technologies, the challenges surrounding them are becoming more and more complex. These optimization problems were traditionally solved using classical deterministic methods [1], which were often quite efficient in finding the solutions. But the ever increasing complexity of these problems has made these classical techniques quite unreliable for various real-world engineering problems. Here, various metaheuristic optimization [1] algorithms prove to be quite successful. Often inspired by nature, these algorithms make use of various stochastic [2] techniques for finding the solution. These metaheuristics are higher-level heuristics that try to find a partial solution called heuristic representing a sufficiently good solution to the optimization problem [3–6]. The presence of randomness drives the algorithms towards promising regions of the search space. Simplicity of use, high reliability, and high

flexibility has made these algorithms quite popular in last few years [7].

Among these algorithms, many are based on multiagent paradigms. These include various evolutionary algorithms [8] inspired by the evolutionary mechanisms found in nature. These algorithms [9–15] use various mechanisms like crossover, mutation, and selection for solving the optimization problems. Among the most popular metaheuristic, GA [9] was proposed by Holland in 1970s. It represents the information in the form of genetic representation as bits and performs evolutionary processes like selection, mutation, and crossover on it. Other population-based algorithms utilize swarm intelligence-based mechanisms [16–20] to solve the optimization problem. These algorithms make use of information distributed in the individual participants of the swarm. The lack of any central control structure makes it vital for the distributed information to be shared amongst the individuals of the swarm. The interaction between these individuals leads to

the global intelligent behaviour which plays an important role in the performance of these algorithms. Thus swarm intelligence can be seen as the accumulation of various agents generally comprising social creatures like birds, ants, bees, fish, and so on and their intelligent behaviour at the individual level to replicate a complicated and collaborative behaviour at the social level. Social researchers around the world have analyzed such behaviours and developed algorithms replicating these behaviours that can be used to solve optimization problems in scientific and engineering domains.

Most of these algorithms focus on one specific strategy to evaluate the search space for the solution. This makes them vulnerable to solve problems that they are not designed to solve. Due to the presence of multiple solution update strategies, ensemble-based approaches can generally be more robust to find the solution in a more complex and unknown search space. The proposed algorithm incorporates an ensemble of solution update strategies to efficiently evaluate the problem search space.

Formulating any swarm intelligence-based optimization algorithm can be a tricky task. For any stochastic optimization method to be successful, it requires to maintain a proper balance between its exploratory and exploitative nature. It requires the understanding of whether a group replicates features of self-organization or not. Self Organization is quite common in various decentralized societies like fish schools and insect colonies. These systems utilize the local information individually to achieve a common complex task. Here, the self-organization process helps in the form of local interactions between the lower system of entities and enables some coordination in the working of the group. This process occurs with some sort of initial randomness and is not controlled by any external/central authority. This overall process takes place in a decentralized manner such that individual entities work on the basis of local information.

This paper proposes a novel swarm intelligence-based metaheuristic optimization algorithm “Structured Clanning-based Ensemble Optimization (SCEO)” inspired by the social organization of the elephant clan. The algorithm utilizes the individual diversity found within the complex social structure of the elephant population to formulate an ensemble-based optimization algorithm. So far, only a small amount of work has been done on developing elephant-social-structure-inspired algorithms. In 2016, Wang et al. [21] proposed the elephant herding optimization (EHO) algorithm inspired by herding behaviour in the elephant groups. In elephant herding optimization (EHO) algorithm, only one type of male elephants (apart from the bounded clan elephants) are simulated that although live in isolation with the parent clan, stay in touch with the clan by simulating low-frequency vibrations. Compared to that, the proposed SCEO algorithm simulates a clan with elephants possessing multiple types of behaviour. The algorithm simulates elephants with varying relational strength to the clan. Apart from the various types of bounded elephants (child, female, bounded, and semibounded), some male

elephants are unbounded and can move independently to any particular clan. This enables the proposed algorithm to use an ensemble of solution search strategies within it making it more robust to different types of optimization problem.

The rest of the paper is organized as follows: Section 2 gives an overview of the social structure of the elephant population followed by the explanation of the proposed algorithm in Section 3. Section 4 contains analysis of impact of various parameters on the performance of the algorithm. Section 5 compares the proposed algorithm with various state-of-the-art optimization algorithms. Finally, conclusion of this study is given in Section 6.

2. Social Structure of the Elephants

From the Elephantidae family and the Proboscidea order, elephants are one of the largest land animals on Earth. Studies had shown them to be among the most intelligent organisms. Elephants are well known to have a very complex social structure. Their social order is highly fission-fusion based with relationship between individuals ranging from mother-child bond, bond groups, independent males, and strangers [22]. Apart from this, the degree of bonding can vary quite a lot with time. The behaviour of individuals and their relationship with others varies widely over time. As with many sexually dimorphic mammals, the roles and behaviour characteristics of male and female elephant are quite different.

An elephant society is organized in the form of groups of families bound together in the form of a clan. Each clan is composed of those families who share the same foraging area. A clan is usually made up of several bond groups and numerous families such that several hundred elephants may make up a clan. Each family is led by a female elephant called Matriarch. These female leaders are generally the largest and oldest female in the group and largely affect the structure and fortune of their group. The family unit can vary widely in size with anywhere from 2 to 50 elephants. Apart from the female leader “Matriarch,” each family consists of a mix of other individuals including child elephants, younger female elephants, and younger dependent male elephants.

In contrast to female elephants, the male elephants can show different kinds of behaviour. Young male elephants are bounded with their native family and at 9 to 18 years of age, they leave the group as independent elephants. These independent male elephants can form a small group with other male elephants. But these groups are generally loosely bound. During the mating period, these independent elephants join different family groups moving from one group to other. They do not have preferences for specific family units and will randomly move to different groupings for reproductively receptive females.

This diversity in the individual behaviour and fission-fusion-based social structure is the main inspiration for the proposed algorithm Structured Clanning-based Ensemble Optimization (SCEO).

3. Elephant Social Order-Based Ensemble Optimization Algorithm

To properly mimic the fission-fusion-based social structure of the elephant clan, the population is divided into 2 or more groups. Further, the population as a whole consists of different types of agents with different individualistic property associated with each. These agents are based on behaviour of female leaders, child elephants, other female elephants, bounded male elephants (young males), semibounded male elephants, and unbounded male elephants (fully independent males) (Figure 1).

The agents have different behaviours associated with them modelled in the form of the properties and strategies associated with them.

3.1. Global Female Leader (GFL). The global female leader is a local female leader that occupies the best position found by the population so far. This leader retains its position in the next iteration and is used to guide other female local leaders to possibly better position in the search space.

3.2. Local Female Leader (LFL). The local female leader occupies the best position found by her family so far. All the other agents of the family move in a position relative to her; thus, this female acts as a guide to all other members in the family. The position of the local leader is updated relative to the global female leader (GFL). This helps in maintaining an interaction between different families of the swarm. The position of the local female leader is updated as per the following equation:

$$\begin{aligned} \text{LFL}[f] \cdot \text{pos}[j] &= \text{LFL}[f] \cdot \text{pos}[j] \\ &+ r_1 * w_1 * (\text{GFL} \cdot \text{pos}[j]) \\ &- \text{LFL}[f] \cdot \text{pos}[j]. \end{aligned} \quad (1)$$

The local female leader $\text{LFL}[f]$ belongs to the f th family unit. j represents the particular dimension of the position of the agent to be updated. r_1 is a randomly generated number between 0 and 1. w_1 is the weight coefficient affecting the influence of global female leader on the local female leader. Its value is taken between 0 and 2. By taking w_1 to be 2, the $r_1 * w_1$ value allows the local female leader (LFL) to explore the hypervolume covered by maximum $2 * \Delta d$ distance in each dimension around the global female leader (GFL) (where Δd is distance between LFL and GFL). The manuscript is now improved to include the above explanation. This limits the net movement of LFL to be within 0 to 2 times the distance between GFL in each direction. This can also be seen in Figure 2. In vector form, the position update equation for local female leader can be formulated as given in Equation (2). Here, P^{old} is the old position and P^{new} is the new position of the agent. The matrix operation $(r \leq \text{pr}_1)$ is a Boolean operation and compares the two vectors resulting in 1 at indexes where the condition is met and 0 where the condition is not met:

$$P^{\text{new}} = P^{\text{old}} + (r \leq \text{pr}_1) * r_1 * w_1 * (\text{GFL} - P^{\text{old}}). \quad (2)$$

The numerical example shown in Equation (3) shows the update equation step by step calculation for one of the trials for a 5-dimensional sphere function.

Here, $P^{\text{old}} = [54.50, -11.77, 22.11, 50.87, -26.21]$, $w_1 = 0.2$, and $\text{GFL} = [9.29, -16.19, 9.00, 56.41, -14.74]$:

$$\begin{aligned} P^{\text{new}} &= \begin{bmatrix} 54.50 \\ -11.77 \\ 22.11 \\ 50.87 \\ -26.21 \end{bmatrix} + \left(\begin{bmatrix} 0.17 \\ 0.18 \\ 0.09 \\ 0.50 \\ 0.21 \end{bmatrix} \leq \begin{bmatrix} 0.2 \\ 0.2 \\ 0.2 \\ 0.2 \\ 0.2 \end{bmatrix} \right) \begin{bmatrix} 0.24 \\ 0.40 \\ 0.19 \\ 0.19 \\ 0.14 \end{bmatrix} \begin{bmatrix} 1.5 \\ 1.5 \\ 1.5 \\ 1.5 \\ 1.5 \end{bmatrix} \\ &\cdot \left(\begin{bmatrix} 9.29 \\ -16.19 \\ 9.00 \\ 56.41 \\ -14.74 \end{bmatrix} - \begin{bmatrix} 54.50 \\ -11.77 \\ 22.11 \\ 50.87 \\ -26.21 \end{bmatrix} \right) \\ &= \begin{bmatrix} 54.50 \\ -11.77 \\ 22.11 \\ 50.87 \\ -26.21 \end{bmatrix} + \begin{bmatrix} 1 \\ 1 \\ 1 \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} 0.24 \\ 0.40 \\ 0.19 \\ 0.19 \\ 0.14 \end{bmatrix} \begin{bmatrix} 1.5 \\ 1.5 \\ 1.5 \\ 1.5 \\ 1.5 \end{bmatrix} \left(\begin{bmatrix} 9.29 \\ -16.19 \\ 9.00 \\ 56.41 \\ -14.74 \end{bmatrix} - \begin{bmatrix} 54.50 \\ -11.77 \\ 22.11 \\ 50.87 \\ -26.21 \end{bmatrix} \right) \\ &= \begin{bmatrix} -31.10 \\ -14.42 \\ 18.45 \\ 50.87 \\ -26.21 \end{bmatrix}. \end{aligned} \quad (3)$$

Not all the dimensions of the position is changed by GFL. This is decided on the basis of perturbation rate (pr_1). The value of perturbation rate is taken between 0 and 1 with larger value causing more dimensions to be affected. Perturbation rate (pr_1) in the SCEO algorithm is taken to introduce mutation factor into the position update of the LFL agents. In many ways, this is similar in principle to the perturbation factors used in various evolutionary algorithms like differential evolution. By generating a random number (r_5) between 0 and 1 and comparing it to the pr_1 value, some of the dimensions of the LFL can be restricted from change (if $r_5 > \text{pr}_1$). Thus, the perturbation rate can be defined as the average ratio of dimensions that are mutated in the direction of GFL.

3.3. Other Permanently Bounded Agents. These include agents inspired by the behaviour of child elephants, other female elephants, and young male elephants which are permanently bounded to their native family unit. These agents move in a fixed proximity of their group's local female leader. The child agents move relatively near to the local female leader forming the innermost family circle. Child agents are followed by the female agents that are still in close

FIGURE 1: Structured clanning-based ensemble optimization algorithm.

proximity to the local leader as compared to the young male agents. These agents change their position as per the following equation:

$$\text{Swarm}[i] \cdot \text{pos}[j] = \text{LFL}[f] \cdot \text{pos}[j] + r_2 * w_2 * \alpha * (\text{Ub}[j] - \text{Lb}[j]). \quad (4)$$

In vector form, the position update equation for bounded agents can be formulated as given in Equation (5). Equation (6) shows the numerical calculation example for solution update of a bounded agent. The example uses child agents with $w_2 = 0.02$. The α value in the example is taken to be 1 representing very first iteration in the solution search space:

$$P^{\text{new}} = \text{LFL} + r_2 * w_2 * \alpha * (\text{Ub} - \text{Lb}), \quad (5)$$

$$P^{\text{new}} = \begin{bmatrix} 54.50 \\ -11.77 \\ 22.11 \\ 50.87 \\ -26.21 \end{bmatrix} + \begin{bmatrix} 1.00 \\ -0.31 \\ 0.57 \\ 0.53 \\ 0.19 \end{bmatrix} \begin{bmatrix} 0.02 \\ 0.02 \\ 0.02 \\ 0.02 \\ 0.02 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{bmatrix} \left(\begin{bmatrix} 100 \\ 100 \\ 100 \\ 100 \\ 100 \end{bmatrix} - \begin{bmatrix} -100 \\ -100 \\ -100 \\ -100 \\ -100 \end{bmatrix} \right) \\ = \begin{bmatrix} 54.50 \\ -11.77 \\ 22.11 \\ 50.87 \\ -26.21 \end{bmatrix} + \begin{bmatrix} 1.00 \\ -0.31 \\ 0.57 \\ 0.53 \\ 0.19 \end{bmatrix} \begin{bmatrix} 4 \\ 4 \\ 4 \\ 4 \\ 4 \end{bmatrix} = \begin{bmatrix} -50.51 \\ -13.03 \\ 24.37 \\ 53.00 \\ -25.45 \end{bmatrix}. \quad (6)$$

Here, $\text{Swarm}[i]$ represents the i th agent of the swarm bounded to the f th group. Ub and Lb are the upper and

lower bounds of the search space. r_2 is a randomly generated number between -1 and 1 . w_2 is the weight coefficient defining the local leader proximity region in which the i th agent can move. Its value is taken to be the lowest for child agents and highest for the young male agents. The low value of w_2 helps in better exploitation of the search region. α is used to control the balance between exploration and exploitation of the algorithm. Its value is linearly decreased from high value (near 1) at the start to a low value (near 0) at the end of the algorithm. The high value at the start helps in better exploration as the agents are able to move larger area around the local female leader. The low value of α at the end reduces the movement region for the agent, thus helping in the exploitation of the search space near the female leaders at the end of Algorithm 1.

3.4. Semibounded Male Agents. These male agents are not bounded to a single group and instead move from one group to other randomly. The movement of these agents in the search space is implemented using the following equation:

$$\begin{aligned} \text{Swarm}[i] \cdot \text{pos}[j] &= \text{Swarm}[i] \cdot \text{pos}[j] \\ &+ r_3 * w_3 * (\text{LFL}[r_f] \cdot \text{pos}[j] \\ &- \text{Swarm}[i] \cdot \text{pos}[j]) \\ &+ r_4 * w_4 * (\text{Swarm}[i] \cdot \text{bestpos}[j] \\ &- \text{Swarm}[i] \cdot \text{pos}[j]). \end{aligned} \quad (7)$$

The position update equation in the vector form for semibounded agents can be formulated as given in Equation (8). Equation (9) shows the numerical calculation example

FIGURE 2: w_2 range for local female leader.

for solution update of a semibounded agent. In the example, the weights w_3 and w_4 are taken to be 1.33:

$$P^{\text{new}} = P^{\text{old}} + r_3 * w_3 * (LFL - P) + r_4 * w_4 * (P^{\text{best}} - P), \quad (8)$$

$$P^{\text{new}} = \begin{bmatrix} 59.87 \\ -55.56 \\ 65.74 \\ -18.35 \\ 80.26 \end{bmatrix} + \begin{bmatrix} 0.40 \\ 0.89 \\ 0.18 \\ 0.82 \\ 0.3 \end{bmatrix} \begin{bmatrix} 1.33 \\ 1.33 \\ 1.33 \\ 1.33 \\ 1.33 \end{bmatrix} \left(\begin{bmatrix} -31.1 \\ -14.42 \\ 18.45 \\ 50.87 \\ -26.21 \end{bmatrix} - \begin{bmatrix} 59.87 \\ -55.56 \\ 65.74 \\ -18.35 \\ 80.26 \end{bmatrix} \right) + \begin{bmatrix} 0.89 \\ 0.20 \\ 0.73 \\ 0.35 \\ 0.84 \end{bmatrix} \begin{bmatrix} 1.33 \\ 1.33 \\ 1.33 \\ 1.33 \\ 1.33 \end{bmatrix} \left(\begin{bmatrix} 59.87 \\ -55.56 \\ 65.74 \\ -18.35 \\ 80.26 \end{bmatrix} - \begin{bmatrix} 59.87 \\ -55.56 \\ 65.74 \\ -18.35 \\ 80.26 \end{bmatrix} \right) = \begin{bmatrix} 12.04 \\ -7.12 \\ 54.62 \\ 56.70 \\ 38.06 \end{bmatrix}. \quad (9)$$

Here, $\text{Swarm}[i]$ represents the i th agent of the swarm. r_f is the randomly selected family and bestpos is the best position that has been found by the i th so far. r_3 and r_4 are random numbers between 0 and 1. w_3 and w_4 are weight coefficients affecting the influence of the female leader and the agent's best position on its movement.

3.5. Unbounded Male Agents. These are independent agents that are not bound to any particular family and move randomly in the search space. This is formulated by random selection of three agents in the search space and using their positions to change the position of this unbounded agent. The position update equation is as follows:

$$\begin{aligned} \text{Swarm}[i] \cdot \text{pos}[j] &= \text{Swarm}[re_1] \cdot \text{pos}[j] \\ &+ w_5 * (\text{Swarm}[re_2] \cdot \text{pos}[j] \\ &- \text{Swarm}[re_3] \cdot \text{pos}[j]). \end{aligned} \quad (10)$$

The position update equation in the vector form for unbounded agents can be formulated as given in Equation (11). Equation (12) shows the numerical calculation example for solution update of an unbounded agent. In the example the weight w_5 is taken as 0.5 and perturbation rate is taken 0.6:

$$P^{\text{new}} = (r > pr_2) * P^{\text{old}} + (r \leq pr_2) * (P^{re_1} + w_5 * (P^{re_2} - P^{re_3})), \quad (11)$$

$$\begin{aligned} P^{\text{new}} &= \begin{pmatrix} \begin{bmatrix} 0.06 \\ 0.29 \\ 0.65 \\ 0.12 \\ 0.32 \end{bmatrix} \leq \begin{bmatrix} 0.6 \\ 0.6 \\ 0.6 \\ 0.6 \\ 0.6 \end{bmatrix} \end{pmatrix} \\ &* \left(\begin{bmatrix} -8.12 \\ -6.5 \\ 34.15 \\ -32.71 \\ 59.37 \end{bmatrix} + \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \end{bmatrix} * \left(\begin{bmatrix} -31.1 \\ -14.42 \\ 18.45 \\ 50.87 \\ -26.21 \end{bmatrix} - \begin{bmatrix} 1.23 \\ -11.35 \\ 30.43 \\ -25.67 \\ 66.75 \end{bmatrix} \right) \right) \\ &+ \begin{pmatrix} \begin{bmatrix} 0.06 \\ 0.29 \\ 0.65 \\ 0.12 \\ 0.32 \end{bmatrix} > \begin{bmatrix} 0.6 \\ 0.6 \\ 0.6 \\ 0.6 \\ 0.6 \end{bmatrix} \end{pmatrix} * \begin{bmatrix} -87.28 \\ 61.37 \\ 27.17 \\ 53.67 \\ 13.98 \end{bmatrix} \\ &= \begin{bmatrix} 1 \\ 1 \\ 0 \\ 1 \\ 1 \end{bmatrix} * \left(\begin{bmatrix} -8.12 \\ -6.5 \\ 34.15 \\ -32.71 \\ 59.37 \end{bmatrix} + \begin{bmatrix} 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \\ 0.5 \end{bmatrix} * \left(\begin{bmatrix} -31.1 \\ -14.42 \\ 18.45 \\ 50.87 \\ -26.21 \end{bmatrix} - \begin{bmatrix} 1.23 \\ -11.35 \\ 30.43 \\ -25.67 \\ 66.75 \end{bmatrix} \right) \right) \\ &+ \begin{bmatrix} 0 \\ 0 \\ 1 \\ 0 \\ 0 \end{bmatrix} * \begin{bmatrix} -87.28 \\ 61.37 \\ 27.17 \\ 53.67 \\ 13.98 \end{bmatrix} \\ &= \begin{bmatrix} -35.98 \\ -6.72 \\ 27.17 \\ 5.56 \\ 12.89 \end{bmatrix}. \end{aligned} \quad (12)$$

re_1 , re_2 , and re_3 are 3 randomly selected agents from the swarm. The update probability of the position is decided on

FIGURE 3: Flowchart of the SCEO algorithm.

- (1) **Procedure** Structured Clanning-based Ensemble Optimization
- (2) Initialize Swarm of N_g groups of agents with each group containing 1 female leader, N_c child agents, N_f female agents and N_b bounded male agents. D is dimension of the search space.
- (3) Initialize N_{sb} semi bounded agents and assign a random group to it.
- (4) Initialize N_{ub} unbounded agents.
- (5) Total agents $N = N_g * (1 + N_c + N_f + N_b) + N_{sb} + N_{ub}$
- (6) Find Fitness of all the agents.
- (7) **for** Stopping criteria not met **do**
- (8) Assign best fitness position in each f th group to the Local Leader (LFL_f).
- (9) Assign best LFL position to the Global Female Leader (GFL)
- (10) **for** each $i \in \{1, 2, \dots, N\}$ **do**
- (11) **if** Agent " i " is Local Female Leader of f th family **then**
- (12) **for** each $j \in \{1, 2, \dots, D\}$ **do**
- (13) Generate random number r between 0 and 1.
- (14) **if** $r \leq pr_1$ **then**
- (15) Update position using Equation (1)
- (16) **else if** Agent " i " is Bounded Agent of f th family **then**
- (17) **for** each $j \in \{1, 2, \dots, D\}$ **do**
- (18) Update position using Equation (4)
- (19) **else if** Agent " i " is Semibounded Agent **then**
- (20) **for** each $j \in \{1, 2, \dots, D\}$ **do**
- (21) Update position using Equation (7)
- (22) **else if** Agent " i " is Unbounded Agent **then**
- (23) **for** each $j \in \{1, 2, \dots, D\}$ **do**
- (24) Generate random number r between 0 and 1.
- (25) **if** $r \leq pr_2$ **then**
- (26) Update position using Equation (10)
- (27) Update fitness value of the Agent
- (28) Update alpha value (linearly decrease)

ALGORITHM 1

the basis of the perturbation rate (pr_2), with higher value causing more dimensions to get modified.

The overall SCEO algorithm can be implemented as in Algorithm 19. Figure 3 shows the overall flowchart of the proposed SCEO algorithm.

4. Impact of Parameters

The performance of every optimization algorithm is largely affected by the choice of the related hyperparameter values. The proposed algorithm has 8 such hyperparameters: number of groups (N_g), number of child agents per group (N_c), number of female agents per group (N_f) and number of bounded agents per group (N_b), number of semibounded agents (N_{sb}), number of unbounded agents (N_{ub}), and perturbation rates (pr_1 and pr_2).

To find out the proper configuration of the hyperparameters, a tuning strategy called F-Race [23] is used. F-Race is type of racing algorithm that utilizes Friedman test [24] to evaluate a set of configurations and eliminate the poor ones based on their performance. This leads to efficient allocation of the computational resources towards testing the promising configurations more. The proposed algorithm was tested with 4374 different configurations with different possible values of parameters as shown in Table 1.

For F-Race, the number of initial races r was taken as 5 and significance level α was taken as 0.05. The maximum

executions for the algorithm was set to 60000. For evaluation, 8 different functions (as mentioned in Table 2) were taken. The configurations were evaluated using vertical approach based on the mean fitness with a precision up to 8 decimals. The maximum function evaluation was set to 300000 ($= 30 * 10000$). For first 40000 executions, the functions g_1 , g_3 , g_4 , and g_5 were used for evaluation; after that, functions g_2 , g_6 , g_7 , and g_8 were used. In this way, both the learning and testing sets have functions with similar characteristics with regards to modality. The resulting tuned values are shown in Table 1.

5. Experimental Setup

This section benchmarks the performance of the proposed SCEO algorithm compared to other state-of-the-art algorithms on various benchmark functions taken from [25–28]. The benchmark functions in Tables 3–5 provide a wide range of optimization benchmarks with varying degree of complexity in terms of shift, rotation, and modality. The benchmark function tables show the number of dimensions, range, optimal value, and modality of the functions along with their mathematical expressions. Further, to crank up the benchmark difficulty, the proposed algorithm is tested on full set of CEC 2014 [28] test set and compared against some relatively new algorithms.

TABLE 1: SCEO checked configuration for hyperparameters.

Parameters	Configuration values	Tuned value
Number of groups (N_g)	[2, 3]	3
Number of child agents per group (N_c)	[5, 10, 15]	10
Number of female agents per group (N_f)	[5, 10, 15]	6
Number of bounded agents per group (N_b)	[5, 10, 15]	4
Number of semibounded agents (N_{sb})	[10, 20, 30]	20
Number of unbounded agents (N_{ub})	[10, 20, 30]	20
Perturbation rates (pr_1)	[0.25, 0.5, 0.75]	0.2
Perturbation rates (pr_2)	[0.25, 0.5, 0.75]	0.75

TABLE 2: Set of functions for F-Race-based SCEO tuning.

Function	Range	Dimension
Rotated Bent Cigar Function (g_1)	[-100, 100]	10
Rotated Discus Function (g_2)	[-100, 100]	10
Shifted and Rotated Rosenbrock Function (g_3)	[-100, 100]	10
Shifted and Rotated Ackley Function (g_4)	[-100, 100]	10
Shifted and Rotated Greiwank Function (g_5)	[-100, 100]	10
Shifted and Rotated Rastrigin Function (g_6)	[-100, 100]	10
Shifted and Rotated Schwefelr Function (g_7)	[-100, 100]	10
Shifted and Rotated Katsuura Function (g_8)	[-100, 100]	10

TABLE 3: Lattice parameters, volume, density, and crystallite size derived using Scherrer's equation.

Function		D	Range	$f_{\min}$	
Beale	$F_1(x) = [1.5 - x_1(1 - x_2)]^2 + [2.25 - x_1(1 - x_2^2)]^2 + [2.625 - x_1(1 - x_2^3)]^2$	2	[-4.5, 4.5]	0	U
Six Hump	$F_2(x) = (4 - 2.1x_1^2 + x_1^4/3)x_1^2 + x_1x_2 + (-4 + 4x_2^2)x_2^2$	2	[-5, 5]	-1.0316	M
Camel Back	$F_3(x) = [1 + (x_1 + x_2 + 1)^2(19 - 14x_1 + 13x_1^2 - 14x_2 + 6x_1x_2 - 3x_2^2)] * [30 + (2x_1 - 3x_2)^2(18 - 32x_1 + 12x_1^2 - 48x_2 - 36x_1x_2 + 27x_2^2)]$	2	[-2, 2]	3	M
Goldstein					
Easom	$F_4 = -\cos(x_1)\cos(x_2)\exp(-(x_1 - \pi)^2 - (x_2 - \pi)^2)$	2	[-100, 100]	-1	U
Hartmann 3D	$F_5(x) = -\sum_{i=1}^4 \alpha_i \exp[-\sum_{j=1}^3 A_{ij}(x_j - P_{ij})^2]$	3	[0, 1]	-3.8628	M
Hartmann 6D	$F_6(x) = -\sum_{i=1}^4 \alpha_i \exp[-\sum_{j=1}^6 A_{ij}(x_j - P_{ij})^2]$	6	[0, 1]	-3.3224	M
Elliptic	$F_7(x) = \sum_{i=1}^D (10^6)^{((i-1)/(D-1))} x_i^2$	30	[-100, 100]	0	U
Sphere	$F_8(x) = \sum_{i=1}^D x_i^2$	30	[-100, 100]	0	U
Schwefel 1.2	$F_9(x) = \sum_{i=1}^D (\sum_{j=1}^i x_j)^2$	30	[-100, 100]	0	U
Schwefel 2.22	$F_{10}(x) = \sum_{i=1}^D x_i + \prod_{i=1}^D x_i $	30	[-10, 10]	0	U
Schwefel 2.21	$F_{11}(x) = \max(x_i) \quad 1 \leq i \leq D$	30	[-100, 100]	0	U
Rosenbrock	$F_{12}(x) = \sum_{i=1}^{D-1} (100(x_i^2 - x_{i+1})^2 + (x_i - 1)^2)$	30	[-5, 10]	0	U
Step	$F_{13}(x) = \sum_{i=1}^D (\lfloor x_i + 0.5 \rfloor)^2$	30	[-100, 100]	0	U
Griewank	$F_{14}(x) = \sum_{i=1}^D (x_i^2/4000) - \prod_{i=1}^D \cos(x_i/\sqrt{i}) + 1$	30	[-600, 600]	0	M
Ackley	$F_{15}(x) = -20 \exp(-0.2 \sqrt{(1/D) \sum_{i=1}^D x_i^2}) - \exp((1/D) \sum_{i=1}^D \cos(2\pi x_i)) + 20 + \exp(1)$	30	[-32, 32]	0	M
Rastrigin	$F_{16}(x) = 10D + \sum_{i=1}^D [x_i^2 - 10 \cos(2\pi x_i)]$	30	[-5.12, 5.12]	0	M
Weierstrass	$F_{17}(x) = \sum_{i=1}^D (\sum_{k=0}^{k_{\max}} [a^k \cos(2\pi b^k (x_i + 0.5))]) - D \sum_{k=0}^{k_{\max}} [a^k \cos(2\pi b^k \cdot 0.5)]$	30	[-0.5, 0.5]	0	M
Schwefel	$F_{18} = 418.98288727243369 * D - \sum_{i=1}^D x_i \sin(\sqrt{ x_i })$	30	[-500, 500]	0	M
Noise	$F_{19}(x) = \sum_{i=1}^D ix^4 + \text{random} [0, 1]$	30	[-1.28, 1.28]	0	U

TABLE 4: Comparison (atomic % and weight %) of LiCoO₂ and Ce-doped compositions from EDX analysis.

Function		D	Range	$f_{\min}$	
Penalised1 Function	$F_{20} = \pi/D\{10 \sin^2(\pi y_1) + \sum_{i=1}^{D-1} (y_i - 1)^2 [1 + 10 \sin^2(\pi y_{i+1})] + (y_D - 1)^2\} + \sum_{i=1}^D u(x_i, 10, 100, 4)$	30	[-50, 50]	0	M
	$u(x_i, p, q, r) = \begin{cases} q(x_i - p)^r, & x_i > p, \\ 0, & -p < x_i < p, \\ q(-x_i - p)^r, & x_i < -p. \end{cases}$				
Penalised2 Function	$F_{21} = 0.1\{\sin^2(3\pi x_1) + \sum_{i=1}^D (x_i - 1)^2 [1 + \sin^2(3\pi x_i + 1)] + (x_D - 1)^2 [1 + \sin^2(2\pi x_D)]\} + \sum_{i=1}^D u(x_i, 10, 100, 4)$	30	[-50, 50]	0	M
Shifted Elliptic	$F_{22}(z) = \sum_{i=1}^D (10^6)^{((i-1)/(D-1))} z_i^2, z = x - o$	30	[-100, 100]	0	U
Shifted Ackley	$F_{23}(z) = -20 \exp(-0.2 \sqrt{(1/D) \sum_{i=1}^D z_i^2}) - \exp((1/D) \sum_{i=1}^D \cos(2\pi z_i)) + 20 + \exp(1), z = x - o$	30	[-32, 32]	0	M
Rotated Rastrigin	$F_{24}(z) = f_{16}(M(5.12(z)/100)),$ M is rotation matrix, $z = x - o$	30	[-100, 100]	0	M
Rotated Griewank	$F_{25}(z) = f_{14}(M(z)),$ M is rotation matrix, $z = x - o$	30	[-600, 600]	0	M
Rotated Ackley	$F_{26}(z) = f_{15}(M(z)),$ M is rotation matrix, $z = x - o$	30	[-32, 32]	0	M
Rotated Rosenbrock	$F_{27}(z) = f_{12}(M(z)),$ M is rotation matrix, $z = x - o$	30	[-5, 10]	0	M
Rotated Weierstrass	$F_{28}(z) = f_{17}(M(z)),$ M is rotation matrix, $z = x - o$	30	[-32, 32]	0	M

TABLE 5: Benchmark functions.

Function		D	Range	$f_{\min}$	
Composition Function 1	$F_{29}(x):$	10	[-5, 5]	0	M
	$f_1, f_2 = \text{Ackley Function}$				
	$f_3, f_4 = \text{Rastrigin Function}$				
	$f_5, f_6 = \text{Weierstrass Function}$				
	$f_7, f_8 = \text{Griewank Function}$				
	$f_9, f_{10} = \text{Sphere Function}$				
Composition Function 2	$[\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_{10}] = [1, 1, 1, \dots, 1]$	10	[-5, 5]	0	M
	$[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] =$				
	$[5/32, 5/32, 1, 1, 10, 10, 5/100, 5/100, 5/100, 5/100]$				
	$F_{30}(x):$				
	$f_1, f_2 = \text{Rastrigin Function}$				
	$f_3, f_4 = \text{Weierstrass Function}$				
Composition Function 2	$f_5, f_6 = \text{Griewank Function}$	10	[-5, 5]	0	M
	$f_7, f_8 = \text{Ackley Function}$				
	$f_9, f_{10} = \text{Sphere Function}$				
	$[\sigma_1, \sigma_2, \sigma_3, \dots, \sigma_{10}] = [1, 1, 1, \dots, 1]$				
	$[\lambda_1, \lambda_2, \lambda_3, \dots, \lambda_{10}] =$				
	$[1/5, 1/5, 10, 10, 5/100, 5/100, 5/32, 5/32, 5/100, 5/100]$				

5.1. Evaluation on Benchmark Functions. For each function, the optimization algorithm is used for 10000 function evaluations per dimension of the benchmark function. Each function is tested 51 times and mean, standard deviation, and best and worst of these runs are used to evaluate the performance of the compared algorithms. An error of 10^{-8} has been taken as error tolerance and is considered to be zero error. The proposed algorithm (SCEO) has been compared with other state-of-the-art optimization algorithms: FEP [15], ABC [29], GWO [7], and GSA [30]. The comparison

results are tabulated in Tables 6–9. The parameters of these algorithms were tuned by using sensitivity analysis [31]. The hyperparameter setting for the compared algorithms is shown in Table 10.

Further comparison of these algorithms has been done using parametric and nonparametric tests. t -value and h -value of the algorithm pairs is calculated using two-tailed t -test (parametric test with 5% significance level and 100 degrees of freedom) and tabulated in Tables 11 and 12. In Tables 11 and 12, the negative t -value

TABLE 6: Comparison with FEP [15], ABC [29], GWO [7], and GSA [30].

		SCEO	FEP	ABC	GWO	GSA
F1	Mean	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
	SD	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
	Best	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
	Worst	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
F2	Mean	-1.03E + 00	-1.03E + 00	-1.03E + 00	-1.03E + 00	-1.03E + 00
	SD	0.00E + 00	4.04E - 06	6.82E - 05	0.00E + 00	0.00E + 00
	Best	-1.03E + 00	-1.03E + 00	-1.03E + 00	-1.03E + 00	-1.03E + 00
	Worst	-1.03E + 00	-1.03E + 00	-1.03E + 00	-1.03E + 00	-1.03E + 00
F3	Mean	3.00E + 00	3.02E + 00	3.09E + 00	3.00E + 00	3.00E + 00
	SD	0.00E + 00	1.10E - 01	1.56E - 01	1.12E - 06	0.00E + 00
	Best	3.00E + 00	3.00E + 00	3.00E + 00	3.00E + 00	3.00E + 00
	Worst	3.00E + 00	3.08E + 00	3.27E + 00	3.00E + 00	3.00E + 00
F4	Mean	-1.00E + 00	-1.00E + 00	-1.00E + 00	-1.00E + 00	-1.00E + 00
	SD	0.00E + 00	0.00E + 00	3.45E - 06	0.00E + 00	0.00E + 00
	Best	-1.00E + 00	-1.00E + 00	-1.00E + 00	-1.00E + 00	-1.00E + 00
	Worst	-1.00E + 00	-1.00E + 00	-1.00E + 00	-1.00E + 00	-1.00E + 00
F5	Mean	-3.86E + 00	-3.86E + 00	-3.85E + 00	-3.86E + 00	-3.86E + 00
	SD	0.00E + 00	1.40E - 05	7.75E - 03	5.22E - 05	0.00E + 00
	Best	-3.86E + 00	-3.86E + 00	-3.86E + 00	-3.86E + 00	-3.86E + 00
	Worst	-3.86E + 00	-3.86E + 00	-3.84E + 00	-3.86E + 00	-3.86E + 00
F6	Mean	-3.32E + 00	-3.27E + 00	-3.32E + 00	-3.32E + 00	-3.29E + 00
	SD	5.61E - 03	4.15E - 02	3.22E - 03	4.81E - 03	3.00E - 02
	Best	-3.32E + 00	-3.32E + 00	-3.32E + 00	-3.32E + 00	-3.32E + 00
	Worst	-3.32E + 00	-3.22E + 00	-3.32E + 00	-3.32E + 00	-3.20E + 00
F7	Mean	0.00E + 00	2.10E - 04	0.00E + 00	0.00E + 00	0.00E + 00
	SD	0.00E + 00	1.10E - 04	0.00E + 00	0.00E + 00	0.00E + 00
	Best	0.00E + 00	8.56E - 07	0.00E + 00	0.00E + 00	0.00E + 00
	Worst	0.00E + 00	6.24E - 04	0.00E + 00	0.00E + 00	0.00E + 00
F8	Mean	0.00E + 00	5.70E - 04	0.00E + 00	0.00E + 00	0.00E + 00
	SD	0.00E + 00	1.30E - 04	0.00E + 00	0.00E + 00	0.00E + 00
	Best	0.00E + 00	5.45E - 07	0.00E + 00	0.00E + 00	0.00E + 00
	Worst	0.00E + 00	3.12E - 03	0.00E + 00	0.00E + 00	0.00E + 00

TABLE 7: Comparison with FEP [15], ABC [29], GWO [7], and GSA [30].

		SCEO	FEP	ABC	GWO	GSA
F9	Mean	0.00E + 00	1.60E - 02	0.00E + 00	3.29E - 06	8.97E + 01
	SD	0.00E + 00	1.40E - 02	0.00E + 00	7.12E - 07	3.19E + 01
	Best	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00	1.57E + 02
	Worst	0.00E + 00	2.15E - 01	0.00E + 00	8.77E - 06	2.22E + 01
F10	Mean	0.00E + 00	8.10E - 03	0.00E + 00	0.00E + 00	5.57E - 02
	SD	0.00E + 00	7.70E - 04	0.00E + 00	0.00E + 00	1.97E - 02
	Best	0.00E + 00	1.74E - 05	0.00E + 00	0.00E + 00	3.33E - 05
	Worst	0.00E + 00	4.56E - 02	0.00E + 00	0.00E + 00	4.55E - 01
F11	Mean	0.00E + 00	3.00E - 01	1.10E + 00	5.61E - 07	7.35E + 00
	SD	0.00E + 00	2.15E - 01	1.29E - 01	6.77E - 08	1.74E + 00
	Best	0.00E + 00	7.46E - 05	1.57E - 04	0.00E + 00	1.20E - 03
	Worst	0.00E + 00	1.80E + 00	4.77E + 00	6.23E - 07	1.55E + 01
F12	Mean	4.01E - 02	5.06E + 00	4.86E - 02	2.68E + 01	6.75E + 01
	SD	3.15E - 02	3.95E + 00	4.84E - 02	1.92E + 01	6.22E + 01
	Best	1.81E - 03	1.54E - 02	1.55E - 04	9.77E + 01	1.26E + 01
	Worst	1.58E - 01	2.85E + 01	1.40E - 01	1.11E - 02	1.58E + 02
F13	Mean	0.00E + 00	0.00E + 00	0.00E + 00	8.17E - 01	0.00E + 00
	SD	0.00E + 00	0.00E + 00	0.00E + 00	1.26E - 01	0.00E + 00
	Best	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
	Worst	0.00E + 00	0.00E + 00	0.00E + 00	5.00E + 00	0.00E + 00

TABLE 7: Continued.

		SCEO	FEP	ABC	GWO	GSA
F14	Mean	0.00E + 00	1.62E - 02	0.00E + 00	4.49E - 03	3.48E - 02
	SD	0.00E + 00	2.20E - 02	0.00E + 00	6.66E - 03	4.80E - 02
	Best	0.00E + 00	5.45E - 03	0.00E + 00	0.00E + 00	4.55E - 04
	Worst	0.00E + 00	1.66E - 01	0.00E + 00	1.21E - 02	8.97E - 01
F15	Mean	0.00E + 00	1.80E - 02	0.00E + 00	0.00E + 00	6.21E - 02
	SD	0.00E + 00	2.12E - 03	0.00E + 00	0.00E + 00	4.88E - 02
	Best	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
	Worst	0.00E + 00	7.77E - 02	0.00E + 00	0.00E + 00	2.45E - 01
F16	Mean	1.78E - 05	3.41E + 01	9.58E - 04	3.11E - 01	2.60E + 01
	SD	2.12E - 05	1.22E + 01	1.05E - 03	1.13E - 01	7.47E + 00
	Best	2.69E - 06	5.70E + 01	3.12E - 05	2.77E - 04	4.49E + 01
	Worst	3.85E - 05	2.29E + 01	9.80E - 03	9.98E - 01	4.98E + 00

TABLE 8: Comparison with PSO [17], FEP [15], DE [14], ABC [29], GWO [7], and GSA [30].

		SCEO	FEP	ABC	GWO	GSA
F17	Mean	7.16E - 03	5.45E - 01	3.12E - 03	2.88E - 02	5.35E - 02
	SD	3.12E - 03	2.12E - 01	1.17E - 03	1.55E - 02	4.87E - 02
	Best	7.03E - 04	8.75E - 01	7.03E - 04	9.92E - 04	1.20E - 02
	Worst	1.65E - 02	2.36E - 02	1.17E - 03	4.30E - 02	1.75E - 01
F18	Mean	2.31E + 02	1.50E + 01	8.86E + 01	6.45E + 03	6.75E + 03
	SD	4.55E + 01	5.26E + 01	8.82E + 01	4.09E + 03	4.93E + 02
	Best	0.00E + 00	0.00E + 00	0.00E + 00	1.50E + 03	1.78E + 03
	Worst	3.55E + 02	1.18E + 02	3.55E + 02	7.67E + 03	7.94E + 03
F19	Mean	2.55E - 02	1.42E - 01	1.17E - 01	2.21E - 03	8.94E - 02
	SD	1.21E - 02	3.52E - 01	2.57E - 02	8.70E - 04	4.34E - 02
	Best	2.22E - 03	2.70E - 02	1.87E - 02	1.11E - 03	7.45E - 03
	Worst	4.74E - 02	3.12E - 01	2.87E - 01	4.52E - 03	2.47E - 01
F20	Mean	0.00E + 00	9.20E - 06	0.00E + 00	5.34E - 02	6.21E - 01
	SD	0.00E + 00	3.60E - 06	0.00E + 00	2.07E - 02	9.51E - 01
	Best	0.00E + 00	0.00E + 00	0.00E + 00	4.14E - 03	1.53E - 02
	Worst	0.00E + 00	4.12E - 05	0.00E + 00	7.55E - 02	9.87E - 01
F21	Mean	0.00E + 00	1.60E - 04	0.00E + 00	6.54E - 01	8.89E + 00
	SD	0.00E + 00	7.30E - 05	0.00E + 00	4.47E - 03	7.13E + 00
	Best	0.00E + 00	0.00E + 00	0.00E + 00	6.54E - 01	1.42E + 01
	Worst	0.00E + 00	8.70E - 04	0.00E + 00	6.55E - 01	6.54E - 01
F22	Mean	0.00E + 00	5.22E - 05	0.00E + 00	0.00E + 00	0.00E + 00
	SD	0.00E + 00	2.12E - 02	0.00E + 00	0.00E + 00	0.00E + 00
	Best	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
	Worst	0.00E + 00	8.87E - 04	0.00E + 00	0.00E + 00	0.00E + 00
F23	Mean	0.00E + 00	5.12E + 00	9.73E - 05	0.00E + 00	1.72E + 01
	SD	0.00E + 00	2.11E + 00	5.69E - 05	0.00E + 00	4.11E + 00
	Best	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
	Worst	0.00E + 00	2.01E + 01	4.17E - 04	0.00E + 00	2.01E + 01
F24	Mean	3.71E + 01	8.72E + 01	1.27E + 02	1.21E + 02	1.01E + 02
	SD	1.21E + 01	1.02E + 01	2.13E + 01	3.45E + 01	3.13E + 01
	Best	3.09E + 01	4.59E + 01	7.81E + 01	4.87E + 01	3.97E + 01
	Worst	7.81E + 01	1.23E + 02	1.45E + 02	1.42E + 02	1.45E + 02

indicates that SCEO algorithm is better than the compared algorithm with higher value indicating more statistical difference between from compared algorithms. Further, Shapiro-wilk test [32, 33] was conducted and tabulated in Table 13. This tests the null hypothesis that if the result data are coming from a normal distribution or not. h -value of 1 indicates that the benchmark performance data is not coming from normal distribution and

the parametric test is not sufficient to compare the performance of the algorithms. Thus, to properly compare the algorithms, a nonparametric test in the form of Wilcoxon ranked sum test [34, 35] is conducted and tabulated in Table 14.

It is clearly seen from Tables 6–9, 11, and 12 that SCEO easily outperformed other state-of-the-art algorithms for most of the tested benchmark functions. This is clearly

TABLE 9: Comparison with FEP [15], ABC [29], GWO [7], and GSA [30].

		SCEO	FEP	ABC	GWO	GSA
F25	Mean	0.00E + 00	4.41E - 02	5.12E - 02	7.40E - 03	4.75E - 02
	SD	0.00E + 00	2.71E - 02	2.12E - 02	6.11E - 03	3.12E - 02
	Best	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
	Worst	0.00E + 00	6.77E - 02	7.45E - 02	6.77E - 02	1.42E - 01
F26	Mean	3.48E + 00	2.09E + 01	2.09E + 01	2.09E + 01	2.09E + 01
	SD	8.70E - 01	1.87E - 02	1.29E - 02	3.12E - 03	5.65E - 03
	Best	0.00E + 00	2.09E + 01	2.09E + 01	2.09E + 01	2.09E + 01
	Worst	2.09E + 01	2.09E + 01	2.09E + 01	2.09E + 01	2.09E + 01
F27	Mean	2.88E + 01	6.25E + 01	4.86E + 01	3.55E + 01	7.85E + 01
	SD	1.45E + 01	5.24E + 01	1.55E + 01	1.25E + 01	4.85E + 01
	Best	1.22E - 01	2.55E + 01	1.36E + 00	8.20E - 01	4.10E + 00
	Worst	7.86E + 01	1.45E + 02	1.45E + 02	4.85E + 01	1.88E + 02
F28	Mean	1.25E + 01	2.45E + 01	1.91E + 01	1.35E + 01	1.75E + 01
	SD	4.41E + 00	1.46E + 00	5.45E + 00	4.77E + 00	4.49E + 00
	Best	8.14E - 01	2.22E + 01	1.77E + 00	1.12E + 00	2.55E + 00
	Worst	2.71E + 01	2.72E + 01	2.75E + 01	2.72E + 01	2.78E + 01
F29	Mean	1.58E + 02	3.22E + 02	3.15E + 02	1.42E + 02	1.70E + 02
	SD	7.40E + 01	2.58E + 01	1.81E + 01	8.71E + 01	8.23E + 01
	Best	5.81E + 01	2.97E + 02	3.01E + 02	4.01E + 01	8.21E + 01
	Worst	3.38E + 02	3.38E + 02	3.38E + 02	3.38E + 02	3.38E + 02
F30	Mean	8.81E + 01	1.98E + 02	1.75E + 02	1.02E + 02	2.00E + 02
	SD	2.78E + 01	4.41E + 01	4.91E + 01	6.81E + 01	4.71E + 01
	Best	8.12E + 00	1.48E + 02	1.48E + 02	4.41E + 01	1.48E + 02
	Worst	1.58E + 02	3.02E + 02	3.02E + 02	3.02E + 02	3.02E + 02

TABLE 10: Hyperparameter setting for compared algorithms.

Algorithm	Hyperparameters
FEP	Population-size = 10
ABC	Population-size = 121
	Onlooker-bees = 60
	Employed-bees = 50
	Scout-bees = 1
GWO	Population-size = 40
GSA	Population-size = 50
	Gravitational-constant = 100
	$\alpha = 20$

indicated by the negative t -value for t -test with h -value being 1 indicating better performance of SCEO in comparison to these algorithms. As evident from Tables 11 and 12, t -value and h -value of the 30 tested benchmark functions SCEO outperformed FEP on 25, ABC on 14 (ABC outperformed SCEO on 1 function: F_{17}), GWO on 16, and GSA on 21 of the functions. On other functions, the performance of the algorithms was comparable, and there was not much of a statistical difference in results for those functions. Shapiro-Wilk test (Table 13) clearly shows that parametric test is enough for performance comparison of the tested algorithms as most of the h -values in the table are 0. Similar results can be seen with Wilcoxon test (Table 14) as well. The Wilcoxon test clearly shows that SCEO outperformed FEP on 24, ABC on 13, GWO on 15, and GSA on 21 functions.

5.2. Evaluation on CEC 2014 Benchmark Functions. In order to test the efficiency and reliability of the algorithm, it has

been tested on CEC 2014 [28] test set against some of the newly proposed algorithms. These include L-SHADE [36], MVMO [37, 38], jDE [39], CETMS [40], FWA-DM [41], TSC-PSO [42], and HCA-SA [43] algorithms. L-Shade [36] proposed by Tanabe et al. is a modification of the SHADE [44] algorithm incorporating linear population size reduction (LPSR) strategy within it. MVMO [37] proposed by Erlich et al. makes use of a mechanism that adopts a single parent-offspring pair approach along a normalized search space. jDE [39] algorithm introduces self-adaptivity to the vanilla DE [14] algorithm. CETMS [40] algorithm uses various features of tissue membrane systems and FWA-DM [41] uses differential mutation strategy on the vanilla fireworks algorithm [45]. TSC-PSO [42] applies spatial correlation-based movement strategy on the PSO [17] algorithm. HCA-SA [43] introduces self-adaptivity in the vanilla cuckoo search algorithm (CSA) [46].

Tables 15 and 16 compares the proposed algorithm with L-SHADE [36], MVMO [37, 38], jDE [39], CETMS [40], FWA-DM [41], TSC-PSO [42], and HCA-SA [43] algorithms on CEC 2014 [28] test set. Comparison is done in terms of mean and standard deviation of the 51 runs of each algorithm as per the guidelines of the CEC 2014 [28]. Clearly, the SCEO algorithm offers a comparable performance against all the newly proposed algorithms. To analyze the results, Shapiro-Wilk and Wilcoxon tests were performed as shown in Tables 17 and 18, respectively. As most of the h -values returned by Shapiro-Wilk test are 1, it is possible that the samples are not representing normal distribution, making parametric tests like t -test inefficient for comparing the algorithms. Thus, Wilcoxon test was performed and tabulated in Table 18. The Wilcoxon test

TABLE 11: t-value and h-value comparison with FEP [15], ABC [29], GWO [7], and GSA [30].

		FEP	ABC	GWO	GSA
F1	<i>t</i>	0.00	0.00	0.00	0.00
	<i>h</i>	0	0	0	0
F2	<i>t</i>	0.00	0.00	0.00	0.00
	<i>h</i>	0	0	0	0
F3	<i>t</i>	-1.29	-4.29	-176.78	0.00
	<i>h</i>	0	1	1	0
F4	<i>t</i>	0.00	0.00	0.00	0.00
	<i>h</i>	0	0	0	0
F5	<i>t</i>	-1403.61	-7.83	-20.18	0.00
	<i>h</i>	1	1	1	0
F6	<i>t</i>	-8.78	0.00	0.00	-7.41
	<i>h</i>	1	0	0	1
F7	<i>t</i>	-13.50	0.00	0.00	0.00
	<i>h</i>	1	0	0	0
F8	<i>t</i>	-31.00	0.00	0.00	0.00
	<i>h</i>	1	0	0	0
F9	<i>t</i>	-8.08	0.00	-32.67	-19.88
	<i>h</i>	1	0	1	1
F10	<i>t</i>	-74.38	0.00	0.00	-19.98
	<i>h</i>	1	0	0	1
F11	<i>t</i>	-9.87	-60.30	-58.59	-29.86
	<i>h</i>	1	1	1	1
F12	<i>t</i>	-8.99	-1.04	-9.88	-7.67
	<i>h</i>	1	0	1	1
F13	<i>t</i>	0.00	0.00	-45.83	0.00
	<i>h</i>	0	0	1	0
F14	<i>t</i>	-5.21	0.00	-4.76	-5.13
	<i>h</i>	1	0	1	1
F15	<i>t</i>	-60.04	0.00	0.00	-9.00
	<i>h</i>	1	0	0	1

TABLE 12: *t*-test against FEP [15], ABC [29], GWO [7], and GSA [30].

		FEP	ABC	GWO	GSA
F16	<i>t</i>	-19.85	-6.33	-19.40	-24.58
	<i>h</i>	1	1	1	1
F17	<i>t</i>	-17.94	8.57	-9.67	-6.71
	<i>h</i>	1	1	1	1
F18	<i>t</i>	-19.85	-6.33	-19.40	-24.58
	<i>h</i>	1	1	1	1
F19	<i>t</i>	-2.33	-22.78	1.63	-10.04
	<i>h</i>	1	1	0	1
F20	<i>t</i>	-18.07	0.00	-18.22	-13.38
	<i>h</i>	1	0	1	1
F21	<i>t</i>	-15.50	0.00	-1034.37	-8.82
	<i>h</i>	1	0	1	1
F22	<i>t</i>	-0.02	0.00	0.00	0.00
	<i>h</i>	0	0	0	0
F23	<i>t</i>	-17.16	-12.09	0.00	-29.61
	<i>h</i>	1	1	0	1
F24	<i>t</i>	-22.35	-26.01	-16.28	-13.54
	<i>h</i>	1	1	1	1
F25	<i>t</i>	-11.51	-17.08	-8.56	-10.76
	<i>h</i>	1	1	1	1
F26	<i>t</i>	-141.55	-141.57	-141.58	-141.58
	<i>h</i>	1	1	1	1
F27	<i>t</i>	-4.38	-6.59	-2.49	-6.94
	<i>h</i>	1	1	1	1
F28	<i>t</i>	-18.17	-6.62	-1.05	-5.59
	<i>h</i>	1	1	0	1

TABLE 12: Continued.

		FEP	ABC	GWO	GSA
F29	t	-14.80	-14.58	0.99	-0.79
	h	1	1	0	0
F30	t	-14.92	-10.90	-1.35	-14.47
	h	1	1	0	1

TABLE 13: Shapiro–Wilk test for SCEO, FEP [15], ABC [29], GWO [7], and GSA [30] samples.

	SCEO	FEP	ABC	GWO	GSA
F1	8.20E - 02(0)	1.72E - 01(0)	1.65E - 01(0)	1.72E - 01(0)	1.42E - 01(0)
F2	1.74E - 01(0)	1.44E - 02(1)	4.23E - 02(1)	1.45E - 01(0)	1.72E - 01(0)
F3	8.40E - 02(0)	1.56E - 02(1)	6.45E - 04(1)	8.81E - 04(1)	1.97E - 01(0)
F4	9.00E - 02(0)	1.45E - 01(0)	2.64E - 02(1)	1.17E - 01(0)	1.40E - 01(0)
F5	1.36E - 01(0)	6.95E - 03(1)	1.07E - 02(1)	2.69E - 02(1)	1.60E - 01(0)
F6	1.73E - 02(1)	7.74E - 03(1)	1.25E - 02(1)	1.95E - 02(1)	6.17E - 03(1)
F7	1.11E - 01(0)	1.81E - 02(1)	1.51E - 01(0)	1.74E - 01(0)	1.30E - 01(0)
F8	1.58E - 01(0)	2.97E - 03(1)	1.59E - 01(0)	1.10E - 01(0)	9.70E-02(0)
F9	1.46E - 01(0)	2.74E - 03(1)	1.85E - 01(0)	5.05E - 03(1)	2.48E - 02(1)
F10	9.30E - 02(0)	5.29E - 03(1)	1.97E - 01(0)	1.13E - 01(0)	1.04E - 01(0)
F11	1.87E - 01(0)	3.12E - 03(1)	7.98E - 02(0)	9.09E - 04(1)	1.29E - 03(1)
F12	2.31E - 03(1)	4.54E - 03(1)	5.47E - 02(0)	1.62E - 02(1)	3.25E - 02(1)
F13	8.60E - 02(0)	1.76E - 01(0)	9.50E - 02(0)	2.55E - 02(1)	1.53E - 01(0)
F14	1.42E - 01(0)	4.71E - 02(1)	1.53E - 01(0)	3.48E - 02(1)	4.05E - 03(1)
F15	1.12E - 01(0)	1.38E - 03(1)	1.70E - 01(0)	1.21E - 01(0)	1.32E - 01(0)
F16	6.39E - 02(0)	3.20E - 03(1)	1.17E - 02(1)	1.10E - 02(1)	2.72E - 02(1)
F17	2.75E - 02(1)	1.50E - 02(1)	1.37E - 02(1)	1.68E - 02(1)	3.21E - 02(1)
F18	5.40E - 02(0)	1.19E - 01(0)	1.10E - 02(1)	1.82E - 03(1)	6.26E - 03(1)
F19	8.05E - 03(1)	1.59E - 02(1)	2.06E - 02(1)	6.20E - 02(0)	1.57E - 02(1)
F20	1.85E - 01(0)	6.49E - 02(0)	1.64E - 01(0)	1.08E - 01(0)	1.90E - 02(1)
F21	8.60E - 02(0)	1.00E - 01(0)	1.57E - 01(0)	2.67E - 02(1)	1.06E - 01(0)
F22	1.54E - 01(0)	1.69E - 02(1)	1.32E - 01(0)	9.60E - 02(0)	1.82E - 01(0)
F23	1.24E - 01(0)	6.11E - 02(0)	6.10E - 02(0)	1.05E - 01(0)	1.12E - 02(1)
F24	2.46E - 02(1)	9.70E - 02(0)	1.71E - 02(1)	1.26E - 02(1)	6.09E - 03(1)
F25	1.09E - 01(0)	9.88E - 03(1)	2.71E - 02(1)	1.18E - 01(0)	1.38E - 02(1)
F26	1.46E - 02(1)	3.14E - 02(1)	3.73E - 04(1)	5.59E - 03(1)	1.12E - 02(1)
F27	9.77E - 02(0)	2.35E - 02(1)	3.55E - 02(1)	3.70E - 02(1)	2.62E - 03(1)
F28	2.53E - 03(1)	5.41E - 03(1)	1.18E - 01(0)	1.74E - 02(1)	5.18E - 02(0)
F29	1.09E - 02(1)	1.21E - 03(1)	1.77E - 02(1)	1.46E - 01(0)	3.00E - 04(1)
F30	3.84E - 02(1)	8.10E - 03(1)	1.03E - 03(1)	3.73E - 02(1)	9.42E - 02(0)

TABLE 14: Wilcoxon test against FEP [15], ABC [29], GWO [7], and GSA [30].

	FEP	ABC	GWO	GSA
F1	1.78E - 01(0)	1.07E - 01(0)	8.80E - 02(0)	1.36E - 01(0)
F2	1.92E - 01(0)	8.50E - 02(0)	1.04E - 01(0)	1.27E - 01(0)
F3	6.05E - 02(0)	6.05E - 02(0)	1.76E - 01(0)	1.70E - 01(0)
F4	1.12E - 01(0)	8.80E - 02(0)	1.16E - 01(0)	1.05E - 01(0)
F5	1.69E - 01(0)	2.82E - 11(1)	1.36E - 01(0)	1.17E - 01(0)
F6	5.62E - 10(1)	1.38E - 01(0)	1.27E - 01(0)	1.68E - 06(1)
F7	1.18E - 02(1)	8.10E - 02(0)	1.51E - 01(0)	1.32E - 01(0)
F8	7.39E - 03(1)	8.40E - 02(0)	1.31E - 01(0)	1.75E - 01(0)
F9	2.33E - 09(1)	1.60E - 01(0)	5.23E - 04(1)	8.35E - 04(1)
F10	1.62E - 03(1)	1.67E - 01(0)	1.08E - 01(0)	1.08E - 02(1)
F11	2.71E - 10(1)	1.31E - 02(1)	2.19E - 03(1)	8.27E - 03(1)
F12	2.33E - 09(1)	1.27E - 01(0)	2.82E - 11(1)	1.78E - 08(1)
F13	1.26E - 01(0)	1.96E - 01(0)	5.50E - 03(1)	1.47E - 01(0)
F14	8.66E - 05(1)	1.58E - 01(0)	3.72E - 03(1)	3.72E - 03(1)
F15	5.76E - 03(1)	8.00E - 02(0)	1.39E - 01(0)	2.82E - 11(1)

TABLE 14: Continued.

	FEP	ABC	GWO	GSA
F16	8.52E − 03(1)	7.45E − 07(1)	1.30E − 02(1)	7.53E − 03(1)
F17	1.15E − 02(1)	2.71E − 10(1)	2.09E − 10(1)	5.46E − 07(1)
F18	3.46E − 03(1)	7.45E − 07(1)	5.90E − 03(1)	6.27E − 03(1)
F19	1.24E − 02(1)	8.84E − 03(1)	1.28E − 01(0)	8.95E − 04(1)
F20	1.22E − 02(1)	1.49E − 01(0)	7.36E − 03(1)	6.58E − 03(1)
F21	1.32E − 02(1)	1.47E − 01(0)	3.36E − 03(1)	2.71E − 10(1)
F22	1.95E − 01(0)	1.34E − 01(0)	1.05E − 01(0)	8.70E − 02(0)
F23	8.16E − 03(1)	1.61E − 03(1)	1.53E − 01(0)	8.79E − 04(1)
F24	3.71E − 03(1)	1.02E − 02(1)	5.74E − 03(1)	9.11E − 03(1)
F25	1.28E − 02(1)	1.41E − 02(1)	2.71E − 10(1)	1.59E − 14(1)
F26	2.02E − 04(1)	5.74E − 03(1)	6.90E − 03(1)	3.63E − 03(1)
F27	9.12E − 04(1)	1.01E − 07(1)	2.15E − 02(1)	5.60E − 08(1)
F28	1.43E − 02(1)	6.75E − 08(1)	1.04E − 01(0)	8.70E − 06(1)
F29	1.00E − 02(1)	1.38E − 02(1)	9.60E − 02(0)	9.00E − 02(0)
F30	8.84E − 03(1)	6.42E − 14(1)	1.93E − 01(0)	6.03E − 03(1)

TABLE 15: Comparison with L-SHADE [36], MVMO [37], jDE [39], CETMS [40], FWA-DM [41], TSC-PSO [42], and HCA-SA [43] algorithms on CEC 2014 functions.

		SCEO	L-shade	MVMO	jDE	CETMS	FWA-DM	TSC-PSO	HCS-SA
f1	Mean	2.80E + 03	0.00E + 00	1.07E − 03	6.12E + 04	0.00E + 00	2.76E + 05	3.05E + 01	1.16E + 04
	SD	1.73E + 03	0.00E + 00	1.09E − 03	7.64E + 04	0.00E + 00	1.82E + 05	3.48E + 01	8.79E + 03
f2	Mean	8.10E + 02	0.00E + 00	2.38E − 05	2.27E − 15	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
	SD	0.00E + 00	0.00E + 00	1.19E − 05	7.87E − 15	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
f3	Mean	0.00E + 00	0.00E + 00	1.11E − 03	4.09E − 14	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
	SD	0.00E + 00	0.00E + 00	1.03E − 03	2.60E − 14	0.00E + 00	0.00E + 00	0.00E + 00	0.00E + 00
f4	Mean	1.27E + 01	0.00E + 00	0.00E + 00	8.53E + 00	0.00E + 00	2.04E + 01	0.00E + 00	5.08E − 02
	SD	4.12E + 00	0.00E + 00	0.00E + 00	2.16E + 01	0.00E + 00	1.91E + 01	0.00E + 00	1.33E − 01
f5	Mean	2.00E + 01	2.00E + 01	2.00E + 01	2.03E + 01	2.01E + 01	2.05E + 01	2.00E + 01	2.00E + 01
	SD	5.18E − 03	3.70E − 02	2.00E + 01	3.26E − 02	3.49E − 01	5.36E − 02	2.98E − 04	5.53E − 03
f6	Mean	6.30E + 00	1.40E − 07	1.62E + 00	5.31E + 00	1.86E + 00	1.29E + 01	0.00E + 00	1.91E + 00
	SD	1.57E + 00	9.90E − 07	3.04E + 00	4.04E + 00	1.77E + 00	8.25E + 00	0.00E + 00	2.40E + 00
f7	Mean	0.00E + 00	0.00E + 00	2.99E − 03	2.96E − 04	6.53E − 01	8.55E − 03	0.00E + 00	0.00E + 00
	SD	0.00E + 00	0.00E + 00	0.00E + 00	1.48E − 03	2.65E + 00	9.81E − 03	0.00E + 00	0.00E + 00
f8	Mean	0.00E + 00	0.00E + 00	8.58E − 01	1.19E − 01	1.31E + 01	0.00E + 00	0.00E + 00	0.00E + 00
	SD	0.00E + 00	0.00E + 00	9.95E − 01	3.30E − 01	3.32E + 00	0.00E + 00	0.00E + 00	0.00E + 00
f9	Mean	1.52E + 01	6.80E + 00	2.51E + 01	3.81E + 01	1.24E + 01	5.66E + 01	2.36E + 01	1.84E + 01
	SD	4.23E + 00	1.50E + 00	2.39E + 01	5.71E + 00	5.20E + 00	1.08E + 01	2.56E + 00	3.35E + 00
f10	Mean	2.54E + 02	1.60E − 02	1.79E + 01	3.17E + 00	2.04E + 03	8.53E + 00	2.92E − 01	6.35E − 01
	SD	1.44E + 02	1.60E − 02	9.76E + 00	3.18E + 00	1.26E + 03	2.42E + 00	1.11E − 02	6.84E − 01
f11	Mean	2.16E + 03	1.20E + 03	1.54E + 03	2.71E + 03	2.51E + 03	2.63E + 03	1.25E + 03	1.86E + 03
	SD	2.86E + 02	1.80E + 02	1.59E + 03	2.75E + 02	1.06E + 03	2.48E + 02	2.13E + 02	3.46E + 02
f12	Mean	1.62E − 01	1.60E − 01	7.21E − 02	4.77E − 01	2.65E − 03	3.71E − 01	1.41E − 01	1.48E − 01
	SD	5.18E − 02	2.30E − 02	6.24E − 02	5.41E − 02	3.15E − 03	6.66E − 02	2.14E − 02	4.33E − 02
f13	Mean	2.39E − 01	1.20E − 01	1.57E − 01	2.84E − 01	5.32E − 02	3.89E − 01	4.15E − 01	1.48E − 01
	SD	4.98E − 02	1.70E − 02	1.62E − 01	3.55E − 02	1.98E − 02	5.51E − 02	1.89E − 02	2.43E − 02
f14	Mean	2.06E − 01	2.40E − 01	1.99E − 01	3.02E − 01	5.15E − 01	2.69E − 01	2.53E − 01	2.05E − 01
	SD	5.88E − 02	3.00E − 02	1.99E − 01	4.15E − 02	5.65E − 02	7.76E − 02	4.00E − 02	3.24E − 02
f15	Mean	2.35E + 00	2.10E + 00	2.86E + 00	5.36E + 00	2.04E + 01	7.37E + 00	8.12E + 00	2.66E + 00
	SD	6.49E − 01	2.50E − 01	2.69E + 00	7.43E − 01	8.36E + 01	8.46E − 01	1.12E + 00	5.08E − 01

TABLE 16: Comparison with L-SHADE [36], MVMO [37], jDE [39], CETMS [40], FWA-DM [41], TSC-PSO [42], and HCA-SA [43] algorithms on CEC 2014 functions.

		SCEO	L-shade	MVMO	jDE	CETMS	FWA-DM	TSC-PSO	HCS-SA
f16	Mean	8.39E + 00	8.50E + 00	1.02E + 01	1.03E + 01	1.34E + 01	1.10E + 01	8.15E + 00	9.62E + 00
	SD	7.98E − 01	4.60E − 01	9.84E + 00	3.23E − 01	1.10E + 00	2.71E − 01	2.21E − 01	5.44E − 01
f17	Mean	4.48E + 02	1.90E + 02	9.01E + 02	1.62E + 03	1.48E + 03	6.29E + 03	1.27E + 03	1.22E + 03
	SD	9.85E + 01	7.50E + 01	1.03E + 03	1.49E + 03	4.41E + 02	5.95E + 03	3.87E + 02	4.00E + 02

TABLE 16: Continued.

		SCEO	L-shade	MVMO	jDE	CETMS	FWA-DM	TSC-PSO	HCS-SA
f18	Mean	4.86E + 01	5.90E + 00	2.89E + 01	1.86E + 01	1.57E + 02	7.67E + 01	8.12E - 01	6.11E + 01
	SD	1.28E + 01	2.90E + 00	2.08E + 01	1.04E + 01	4.71E + 01	3.66E + 01	3.30E - 01	1.99E + 01
f19	Mean	3.85E + 00	3.70E + 00	3.08E + 00	4.97E + 00	7.42E + 00	9.95E + 00	2.55E + 00	4.86E + 00
	SD	9.18E - 01	6.80E - 01	3.02E + 00	9.61E - 01	1.97E + 00	1.93E + 00	4.20E - 01	7.68E - 01
f20	Mean	7.48E + 00	3.10E + 00	1.09E + 02	1.36E + 01	1.13E + 02	4.28E + 01	5.21E + 00	1.17E + 01
	SD	2.57E + 00	1.50E + 00	5.69E + 01	6.64E + 00	1.07E + 02	2.61E + 01	1.46E + 00	4.09E + 00
f21	Mean	1.04E + 02	8.70E + 01	4.67E + 02	2.98E + 02	1.07E + 03	7.29E + 02	1.32E + 02	3.31E + 02
	SD	2.75E + 01	9.00E + 01	4.89E + 02	2.25E + 02	2.81E + 03	9.49E + 02	4.86E + 01	1.26E + 02
f22	Mean	3.88E + 01	2.80E + 01	1.45E + 02	1.38E + 02	2.08E + 02	1.46E + 02	2.17E + 01	4.83E + 01
	SD	1.26E + 01	1.80E + 01	1.46E + 02	5.38E + 01	1.42E + 02	8.83E + 01	4.45E + 00	4.52E + 01
f23	Mean	2.81E + 02	3.20E + 02	3.15E + 02	3.15E + 02	3.15E + 02	3.14E + 02	2.60E + 02	3.15E + 02
	SD	3.88E + 01	0.00E + 00	3.15E + 02	0.00E + 00	3.44E - 13	9.28E - 14	5.49E + 01	0.00E + 00
f24	Mean	2.01E + 02	2.20E + 02	2.25E + 02	2.26E + 02	2.29E + 02	2.26E + 02	2.01E + 02	2.24E + 02
	SD	1.86E - 01	1.10E + 00	2.25E + 02	3.34E + 00	6.53E + 00	3.59E + 00	4.71E - 01	7.20E - 01
f25	Mean	2.00E + 02	2.00E + 02	2.03E + 02	2.04E + 02	2.03E + 02	2.01E + 02	2.00E + 02	2.03E + 02
	SD	1.59E - 02	5.00E - 02	2.03E + 02	8.81E - 01	1.47E + 00	1.98E - 01	1.07E - 03	4.55E - 01
f26	Mean	1.00E + 02	1.00E + 02	1.00E + 02	1.00E + 02	1.07E + 02	1.00E + 02	1.01E + 02	1.00E + 02
	SD	5.48E - 02	1.60E - 02	1.00E + 02	5.05E - 02	2.21E + 01	5.35E - 02	1.41E - 01	2.35E - 02
f27	Mean	4.00E + 02	3.00E + 02	4.01E + 02	4.01E + 02	3.76E + 02	4.01E + 02	4.00E + 02	3.49E + 02
	SD	3.04E - 03	0.00E + 00	4.01E + 02	5.44E + 01	4.95E + 01	3.06E + 01	3.04E - 03	4.97E + 01
f28	Mean	5.41E + 02	8.40E + 02	8.77E + 02	8.38E + 02	9.33E + 02	3.93E + 02	3.00E + 02	7.91E + 02
	SD	4.98E + 01	1.40E + 01	8.78E + 02	2.99E + 01	1.25E + 02	1.46E + 01	4.96E - 03	3.69E + 01
f29	Mean	7.13E + 02	7.20E + 02	7.36E + 02	8.66E + 02	1.56E + 06	2.11E + 02	6.43E + 02	3.45E + 05
	SD	8.80E + 01	5.10E + 00	7.42E + 02	1.62E + 02	3.42E + 06	2.90E + 00	4.85E + 01	1.71E + 06
f30	Mean	1.18E - 03	1.20E + 03	2.00E + 03	2.79E + 03	2.92E + 03	4.51E + 02	8.12E + 02	1.95E + 03
	SD	1.48E + 02	6.20E + 02	2.08E + 03	1.22E + 03	1.11E + 03	1.96E + 02	3.46E + 02	7.96E + 02

TABLE 17: Shapiro-Wilk test for SCEO, L-SHADE [36], MVMO [37], jDE [39], CETMS [40], FWA-DM [41], TSC-PSO [42], and HCA-SA [43] algorithms on CEC 2014 functions.

	SCEO	L-shade	MVMO	jDE	CETMS	FWA-DM	TSC-PSO	HCS-SA
f1	3.73E - 03(1)	1.33E - 01(0)	1.48E - 02(1)	1.80E - 03(1)	1.21E - 01(0)	4.02E - 03(1)	7.39E - 04(1)	1.15E - 02(1)
f2	9.10E - 02(0)	8.60E - 02(0)	2.83E - 02(1)	6.22E - 03(1)	1.34E - 01(0)	1.95E - 01(0)	1.68E - 01(0)	1.87E - 01(0)
f3	1.69E - 01(0)	1.81E - 01(0)	4.68E - 02(1)	8.67E - 02(0)	9.70E - 02(0)	9.70E - 02(0)	1.09E - 01(0)	1.88E - 01(0)
f4	4.87E - 02(1)	1.00E - 01(0)	1.53E - 01(0)	4.28E - 03(1)	1.71E - 01(0)	5.67E - 02(0)	1.54E - 01(0)	2.27E - 02(1)
f5	8.69E - 04(1)	1.00E - 02(1)	7.32E - 03(1)	3.78E - 03(1)	8.08E - 04(1)	9.45E - 04(1)	4.67E - 02(1)	2.32E - 02(1)
f6	2.24E - 02(1)	2.73E - 02(1)	7.57E - 03(1)	5.29E - 03(1)	2.09E - 02(1)	7.55E - 03(1)	1.12E - 01(0)	1.17E - 03(1)
f7	1.10E - 01(0)	1.75E - 01(0)	1.39E - 01(0)	8.36E - 04(1)	3.78E - 02(1)	8.73E - 03(1)	1.94E - 01(0)	1.39E - 01(0)
f8	1.29E - 01(0)	1.85E - 01(0)	2.05E - 02(1)	4.44E - 02(1)	3.91E - 02(1)	1.36E - 01(0)	1.21E - 01(0)	8.40E - 02(0)
f9	3.28E - 02(1)	3.66E - 02(1)	4.26E - 02(1)	7.22E - 04(1)	1.26E - 01(0)	4.89E - 03(1)	1.54E - 01(0)	8.00E - 02(0)
f10	3.81E - 02(1)	9.76E - 02(0)	7.77E - 02(0)	2.03E - 02(1)	4.17E - 03(1)	9.03E - 03(1)	3.21E - 03(1)	3.48E - 02(1)
f11	2.24E - 02(1)	3.58E - 03(1)	9.02E - 04(1)	6.21E - 03(1)	3.44E - 02(1)	5.11E - 03(1)	2.09E - 02(1)	4.03E - 02(1)
f12	4.03E - 04(1)	3.81E - 02(1)	2.26E - 03(1)	3.34E - 02(1)	7.44E - 03(1)	6.50E - 03(1)	8.16E - 03(1)	3.60E - 02(1)
f13	5.83E - 04(1)	6.72E - 03(1)	1.24E - 02(1)	5.28E - 03(1)	4.51E - 03(1)	1.52E - 02(1)	1.49E - 02(1)	2.87E - 02(1)
f14	8.30E - 02(0)	8.88E - 02(0)	5.16E - 03(1)	1.78E - 02(1)	7.18E - 03(1)	2.10E - 02(1)	1.13E - 01(0)	3.13E - 03(1)
f15	1.43E - 01(0)	2.50E - 03(1)	1.53E - 01(0)	9.30E - 02(0)	1.46E - 01(0)	1.99E - 01(0)	1.87E - 02(1)	5.12E - 04(1)
f16	6.29E - 02(0)	2.68E - 03(1)	5.37E - 03(1)	3.20E - 02(1)	7.21E - 03(1)	5.18E - 03(1)	7.19E - 02(0)	5.61E - 04(1)
f17	1.51E - 01(0)	1.55E - 01(0)	7.84E - 04(1)	1.50E - 01(0)	6.01E - 03(1)	1.43E - 03(1)	1.89E - 01(0)	1.98E - 03(1)
f18	1.64E - 02(1)	1.05E - 02(1)	1.57E - 02(1)	1.09E - 02(1)	2.89E - 02(1)	4.90E - 04(1)	3.26E - 02(1)	1.43E - 02(1)
f19	6.13E - 03(1)	1.59E - 02(1)	9.29E - 03(1)	5.50E - 03(1)	5.29E - 02(0)	3.28E - 03(1)	9.82E - 02(0)	3.88E - 03(1)
f20	3.18E - 03(1)	1.89E - 03(1)	1.06E - 03(1)	2.17E - 02(1)	4.66E - 04(1)	1.28E - 02(1)	1.15E - 01(0)	1.21E - 02(1)
f21	7.20E - 03(1)	1.97E - 03(1)	8.17E - 03(1)	6.18E - 02(0)	7.48E - 02(0)	5.59E - 04(1)	7.97E - 02(0)	2.10E - 03(1)
f22	4.48E - 02(1)	1.08E - 01(0)	3.61E - 03(1)	1.90E - 03(1)	1.70E - 02(1)	5.75E - 04(1)	1.24E - 01(0)	1.59E - 02(1)
f23	1.99E - 02(1)	1.35E - 01(0)	6.43E - 03(1)	1.77E - 01(0)	1.18E - 01(0)	1.03E - 01(0)	1.75E - 03(1)	1.16E - 01(0)
f24	2.84E - 03(1)	1.04E - 02(1)	8.46E - 02(0)	3.10E - 02(1)	1.95E - 03(1)	5.33E - 03(1)	8.53E - 02(0)	5.75E - 03(1)
f25	5.41E - 02(0)	2.19E - 02(1)	1.53E - 02(1)	1.94E - 02(1)	4.21E - 03(1)	1.55E - 01(0)	1.52E - 01(0)	5.82E - 04(1)

TABLE 17: Continued.

	SCEO	L-shade	MVMO	jDE	CETMS	FWA-DM	TSC-PSO	HCS-SA
f26	1.65E - 03(1)	3.85E - 03(1)	4.63E - 03(1)	2.40E - 02(1)	1.14E - 02(1)	1.80E - 02(1)	1.80E - 02(1)	1.91E - 02(1)
f27	5.13E - 02(0)	9.80E - 02(0)	1.07E - 01(0)	2.49E - 02(1)	2.69E - 02(1)	1.47E - 01(0)	4.08E - 03(1)	8.98E - 02(0)
f28	2.02E - 03(1)	1.50E - 02(1)	1.10E - 02(1)	1.81E - 03(1)	1.37E - 02(1)	1.85E - 02(1)	6.57E - 03(1)	1.70E - 02(1)
f29	2.41E - 02(1)	1.57E - 02(1)	5.22E - 02(0)	2.19E - 02(1)	1.79E - 02(1)	1.93E - 02(1)	1.88E - 03(1)	1.33E - 02(1)
f30	1.08E - 01(0)	1.21E - 02(1)	7.13E - 02(0)	5.76E - 02(0)	8.37E - 03(1)	4.81E - 04(1)	1.06E - 01(0)	4.58E - 03(1)

TABLE 18: Wilcoxon test against L-SHADE [36], MVMO [37], jDE [39], CETMS [40], FWA-DM [41], TSC-PSO [42], and HCA-SA [43] algorithms.

	L-shade	MVMO	jDE	CETMS	FWA-DM	TSC-PSO	HCS-SA
f1	1.01E - 02(1)	1.29E - 02(1)	3.38E - 04(1)	4.62E - 03(1)	2.82E - 11(1)	3.82E - 03(1)	2.70E - 07(1)
f2	8.01E - 04(1)	1.03E - 03(1)	1.31E - 02(1)	9.77E - 03(1)	6.48E - 03(1)	5.91E - 03(1)	5.42E - 03(1)
f3	1.47E - 01(0)	1.78E - 08(1)	5.86E - 03(1)	1.83E - 01(0)	1.26E - 01(0)	1.21E - 01(0)	1.04E - 01(0)
f4	9.09E - 03(1)	7.84E - 03(1)	1.28E - 01(0)	1.36E - 02(1)	1.73E - 02(1)	1.38E - 02(1)	1.20E - 02(1)
f5	1.87E - 01(0)	1.87E - 01(0)	1.38E - 02(1)	8.60E - 02(0)	1.07E - 02(1)	1.36E - 01(0)	1.35E - 01(0)
f6	2.11E - 03(1)	1.12E - 11(1)	9.90E - 02(0)	3.93E - 03(1)	3.13E - 04(1)	1.21E - 02(1)	2.97E - 14(1)
f7	1.41E - 01(0)	1.03E - 02(1)	1.35E - 01(0)	1.28E - 01(0)	1.98E - 05(1)	1.21E - 01(0)	1.06E - 01(0)
f8	9.70E - 02(0)	1.22E - 07(1)	1.25E - 01(0)	1.21E - 02(1)	1.19E - 01(0)	1.61E - 01(0)	1.60E - 01(0)
f9	1.52E - 03(1)	2.61E - 02(1)	1.42E - 03(1)	8.12E - 03(1)	4.39E - 03(1)	2.14E - 03(1)	2.06E - 04(1)
f10	4.92E - 03(1)	1.36E - 02(1)	2.54E - 03(1)	7.08E - 13(1)	3.77E - 04(1)	1.24E - 02(1)	1.05E - 03(1)
f11	3.02E - 03(1)	9.80E - 02(0)	4.01E - 12(1)	9.76E - 02(0)	7.25E - 11(1)	2.43E - 03(1)	7.53E - 05(1)
f12	8.10E - 02(0)	4.35E - 10(1)	1.59E - 03(1)	2.56E - 04(1)	1.15E - 02(1)	1.77E - 02(1)	1.44E - 01(0)
f13	1.76E - 03(1)	5.10E - 03(1)	1.56E - 05(1)	9.73E - 03(1)	1.39E - 02(1)	1.31E - 03(1)	1.94E - 03(1)
f14	2.74E - 03(1)	1.83E - 01(0)	4.20E - 12(1)	6.69E - 03(1)	1.08E - 04(1)	1.05E - 04(1)	1.89E - 01(0)
f15	1.86E - 01(0)	1.93E - 01(0)	1.35E - 02(1)	1.49E - 01(0)	1.37E - 02(1)	4.75E - 03(1)	2.23E - 02(1)
f16	1.48E - 02(1)	1.17E - 02(1)	1.16E - 02(1)	1.30E - 02(1)	8.96E - 03(1)	8.85E - 03(1)	2.24E - 11(1)
f17	3.05E - 03(1)	1.31E - 02(1)	1.05E - 04(1)	1.67E - 03(1)	1.28E - 06(1)	6.38E - 03(1)	1.11E - 02(1)
f18	7.47E - 04(1)	3.04E - 06(1)	1.28E - 02(1)	1.53E - 03(1)	3.22E - 04(1)	8.19E - 03(1)	2.51E - 03(1)
f19	1.79E - 01(0)	1.12E - 01(0)	1.20E - 06(1)	1.05E - 03(1)	5.84E - 03(1)	1.96E - 12(1)	7.98E - 07(1)
f20	1.77E - 13(1)	9.59E - 03(1)	4.77E - 06(1)	8.66E - 05(1)	1.21E - 12(1)	1.98E - 05(1)	6.72E - 07(1)
f21	9.23E - 02(0)	1.53E - 03(1)	2.37E - 05(1)	1.25E - 02(1)	1.05E - 02(1)	1.92E - 03(1)	9.80E - 03(1)
f22	1.36E - 03(1)	2.20E - 03(1)	9.78E - 03(1)	4.75E - 07(1)	2.74E - 09(1)	4.84E - 12(1)	1.42E - 01(0)
f23	1.78E - 08(1)	1.05E - 01(0)	7.45E - 07(1)	7.45E - 07(1)	7.45E - 07(1)	6.23E - 02(0)	7.45E - 07(1)
f24	3.34E - 03(1)	1.03E - 01(0)	4.10E - 03(1)	1.12E - 03(1)	9.50E - 03(1)	1.78E - 01(0)	1.04E - 02(1)
f25	1.69E - 01(0)	1.58E - 01(0)	2.59E - 03(1)	5.28E - 03(1)	1.29E - 03(1)	9.30E - 02(0)	3.84E - 04(1)
f26	8.20E - 02(0)	1.31E - 01(0)	1.78E - 01(0)	4.05E - 02(1)	1.72E - 01(0)	1.03E - 02(1)	1.10E - 01(0)
f27	1.11E - 02(1)	1.43E - 01(0)	9.60E - 02(0)	6.05E - 02(0)	9.90E - 02(0)	1.40E - 01(0)	2.33E - 09(1)
f28	1.30E - 02(1)	2.19E - 02(1)	3.72E - 03(1)	1.04E - 02(1)	3.72E - 03(1)	8.84E - 03(1)	1.42E - 02(1)
f29	7.67E - 02(0)	1.52E - 01(0)	3.04E - 06(1)	2.65E - 02(1)	2.82E - 03(1)	9.41E - 05(1)	6.05E - 04(1)
f30	1.21E - 02(1)	2.42E - 07(1)	8.54E - 03(1)	4.36E - 03(1)	9.88E - 03(1)	1.20E - 02(1)	5.90E - 03(1)

clearly shows that SCEO outperformed L-SHADE on 7 (L-SHADE outperformed on 12), MVMO on 11 (MVMO outperformed on 8), jDE on 20 (jDE outperformed on 2), CETMS on 18 (CETMS outperformed on 7), FWA-DM on 21 (FWA-DM outperformed on 1), TSC-PSO on 8 (TSC-PSO outperformed on 14), and HCA-SA on 15 (HCA-SA outperformed on 8) functions. On other functions, the performance of the compared algorithms was similar.

This clearly proves the reliability of the proposed algorithm in complex optimization scenarios. This was even more evident in case of composition functions (f_{23} – f_{30}) where the SCEO algorithm performed much better than all the compared algorithms. Further, the high convergence rate (Figures 4–5) clearly shows the applicability of

the proposed algorithm in real-world optimization problems.

6. Conclusion

Elephant population in nature is generally composed of highly complex fission-fusion-based social structure. The individuals in the elephant society can exhibit diverse types of behaviour. The population of elephants can consist of various groups with different types of individuals exhibiting different types of behaviours with member of the society. The relationship between individuals can range from mother-child bond, bond groups, independent males, and strangers. The paper proposes a novel swarm intelligence-based metaheuristic optimization algorithm

FIGURE 4: CEC14 Fun3 convergence plot.

FIGURE 5: CEC14 Fun7 convergence plot.

called Structured Clanning-based Ensemble Optimization. The proposed algorithm is inspired by this complex social order of the elephant society and tries to model the individualistic behaviour. The algorithm is tested against other algorithms (FEP, ABC, GWO, and GSA) on 60 different benchmark functions. The experiments clearly show the superiority of the SCEO algorithm against the other tested algorithms proving it to be reliable and efficient for solving complex real-world numerical optimization problems.

The proposed algorithm still has a scope for improvement in its performance. Future research can be directed towards studying the various hyperparameters of the algorithm by incorporating adaptability within them. Further, constrained and multiobjective models of the proposed algorithm can be formulated to extend the utility of the proposed algorithm in the real-world optimization applications.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

References

- [1] M.-H. Lin, J.-F. Tsai, and C.-S. Yu, "A review of deterministic optimization methods in engineering and management," *Mathematical Problems in Engineering*, vol. 2012, Article ID 756023, 15 pages, 2012.
- [2] D. B. Shmoys and C. Swamy, "Stochastic Optimization is (almost) as Easy as deterministic optimization," in *Proceedings of 45th Annual IEEE Symposium on Foundations of Computer Science*, pp. 228–237, Rome, Italy, October 2004.
- [3] I. Boussaïd, J. Lepagnot, and P. Siarry, "A survey on optimization metaheuristics," *Information Sciences*, vol. 237, pp. 82–117, 2013.
- [4] L. Bianchi, M. Dorigo, L. M. Gambardella, and W. J. Gutjahr, "A survey on metaheuristics for stochastic combinatorial optimization," *Natural Computing: An International Journal*, vol. 8, no. 2, pp. 239–287, 2009.
- [5] C. Blum and A. Roli, "Metaheuristics in combinatorial optimization: overview and conceptual comparison," *ACM Computing Surveys*, vol. 35, no. 3, pp. 268–308, 2003.
- [6] E.-G. Talbi, *Metaheuristics: from Design to Implementation*, Wiley, Hoboken, NJ, USA, 2009.
- [7] S. Mirjalili, S. M. Mirjalili, and A. Lewis, "Grey wolf optimizer," *Advances in Engineering Software*, vol. 69, pp. 46–61, 2014.
- [8] A. Trivedi, D. Srinivasan, K. Sanyal, and A. Ghosh, "A survey of multiobjective evolutionary algorithms based on decomposition," *IEEE Transactions on Evolutionary Computation*, vol. 21, no. 3, pp. 440–462, 2017.
- [9] J. H. Holland, "Genetic algorithms: computer programs that 'evolve' in ways that resemble natural selection can solve complex problems ever their creators do not fully understand," in *Scientific American*, Nature Publishing Group, London, UK, 1992.
- [10] L. Poli, G. Oliveri, and A. Massa, "An integer genetic algorithm for optimal clustering in phased array antenna," in *Proceedings of 2017 International IEEE Applied Computational Electromagnetics Society Symposium-Italy (ACES)*, pp. 1–2, Firenze, Italy, March 2017.
- [11] M. H. Manser, I. Musirin, and M. M. Othman, "Immune Log-Normal Evolutionary Programming (ILNEP) for solving economic dispatch problem with prohibited operating zones," in *Proceedings of 2017 4th International Conference on Industrial Engineering and Applications (ICIEA)*, pp. 163–167, IEEE, Nagoya, Japan, April 2017.
- [12] I. Rechenberg, "Evolution strategy," in *Computational Intelligence: Imitating Life I*, Institute of Electrical & Electronics Engineers, Piscataway, NJ, USA, 1994.
- [13] J. R. Koza, *Genetic Programming: On the Programming of Computers by Means of Natural Selection*, Vol. 1, MIT Press, Cambridge, MA, USA, 1992.
- [14] R. Storn and K. Price, "Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, vol. 11, no. 4, pp. 341–359, 1997.
- [15] X. Yao, Y. Liu, and G. Lin, "Evolutionary programming made faster," *IEEE Transactions on Evolutionary Computation*, vol. 3, no. 2, pp. 82–102, 1999.
- [16] M. Dorigo, M. Birattari, and T. Stutzle, "Ant colony optimization," *IEEE Computational Intelligence Magazine*, vol. 1, no. 4, pp. 28–39, 2006.
- [17] R. C. Eberhart and J. Kennedy, "A new optimizer using particle swarm theory," in *Proceedings of the Sixth*

- International Symposium on Micro Machine and Human Science*, vol. 1, Nagoya, Japan, October 1995.
- [18] K. M. Passino, "Biomimicry of bacterial foraging for distributed optimization and control," *IEEE Control Systems*, vol. 22, no. 3, pp. 52–67, 2002.
 - [19] J. C. Bansal, H. Sharma, S. S. Jadon, and M. Clerc, "Spider monkey optimization algorithm for numerical optimization," *Memetic Computing*, vol. 6, no. 1, pp. 31–47, 2014.
 - [20] A. Sharma, A. Sharma, B. K. Panigrahi, D. Kiran, and R. Kumar, "Ageist spider monkey optimization algorithm," *Swarm and Evolutionary Computation*, vol. 28, pp. 58–77, 2016.
 - [21] G.-G. Wang, S. Deb, X.-Z. Gao, and L. dos Santos Coelho, "A new metaheuristic optimization algorithm motivated by elephant herding behavior," *International Journal of Bio-Inspired Computation*, vol. 8, no. 6, p. 394, 2016.
 - [22] R. Sukumar, *The Asian Elephant: Ecology and Management*, Cambridge University Press, Cambridge, UK, 1992.
 - [23] M. Birattari, T. Stützle, L. Paquete, and K. Varrentrapp, "A racing algorithm for configuring metaheuristics," in *Proceedings of the 4th Annual Conference on Genetic and Evolutionary Computation*, pp. 11–18, Morgan Kaufmann Publishers Inc., Honolulu, Hawaii, May 2002.
 - [24] E. Theodorsson-Norheim, "Friedman and Quade tests: BASIC computer program to perform nonparametric two-way analysis of variance and multiple comparisons on ranks of several related samples," *Computers in Biology and Medicine*, vol. 17, no. 2, pp. 85–99, 1987.
 - [25] 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.
 - [26] P. N. Suganthan, N. Hansen, J. J. Liang et al., "Problem definitions and evaluation criteria for the CEC 2005 special session on real-parameter optimization," KanGAL Report, Rep. No. 2005005, 2005.
 - [27] X. S. Yang, "Appendix A: test problems in optimization," in *Engineering Optimization*, pp. 261–266, Taylor & Francis, UK, 2010.
 - [28] J. J. Liang, B. Y. Qu, and P. N. Suganthan, "Problem definitions and evaluation criteria for the CEC 2014 special session and competition on single objective real-parameter numerical optimization," Computational Intelligence Laboratory, Zhengzhou University, Zhengzhou China and Technical Report, Nanyang Technological University, Singapore, 2013.
 - [29] B. Basturk and D. Karaboga, "An artificial bee colony (ABC) algorithm for numeric function optimization," in *Proceedings of IEEE Swarm Intelligence Symposium*, vol. 8, no. 1, pp. 687–697, Indianapolis, IN, USA, May 2006.
 - [30] E. Rashedi, H. Nezamabadi-Pour, and S. Saryazdi, "GSA: a gravitational search algorithm," *Information Sciences*, vol. 179, no. 13, pp. 2232–2248, 2009.
 - [31] A. E. Eiben and S. K. Smit, "Parameter tuning for configuring and analyzing evolutionary algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 19–31, 2011.
 - [32] P. Royston, "Approximating the shapiro-wilk W-test for non-normality," *Statistics and Computing*, vol. 2, no. 3, pp. 117–119, 1992.
 - [33] N. M. Razali and Y. B. Wah, "Power comparisons of shapiro-wilk, Kolmogorov-smirnov, lilliefors and anderson-darling tests," *Journal of Statistical Modeling and Analytics*, vol. 2, no. 1, pp. 21–33, 2011.
 - [34] F. Wilcoxon, S. K. Katti, and R. A. Wilcox, "Critical values and probability levels for the Wilcoxon rank sum test and the Wilcoxon signed rank test," *Selected Tables in Mathematical Statistics*, vol. 1, pp. 171–259, 1970.
 - [35] J. Derrac, S. García, D. Molina, and F. Herrera, "A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms," *Swarm and Evolutionary Computation*, vol. 1, no. 1, pp. 3–18.
 - [36] R. Tanabe and A. S. Fukunaga, "Improving the search performance of SHADE using linear population size reduction," in *Proceedings of 2014 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1658–1665, IEEE, Beijing, China, July 2014.
 - [37] I. Erlich, J. L. Rueda, S. Wildenhues, and F. Shewarega, "Solving the IEEE-CEC 2014 expensive optimization test problems by using single-particle MVMO," in *Proceedings of 2014 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1084–1091, IEEE, Beijing, China, July 2014.
 - [38] I. Erlich, J. L. Rueda, S. Wildenhues, and F. Shewarega, "Evaluating the mean-variance mapping optimization on the IEEE-CEC 2014 test suite," in *Proceedings of 2014 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1625–1632, IEEE, Beijing, China, July 2014.
 - [39] J. Brest and M. S. Maučec, "Self-adaptive differential evolution algorithm using population size reduction and three strategies," *Soft Computing*, vol. 15, no. 11, pp. 2157–2174, 2011.
 - [40] C. Liu and L. Fan, "A hybrid evolutionary algorithm based on tissue membrane systems and CMA-ES for solving numerical optimization problems," *Knowledge-Based Systems*, vol. 105, pp. 38–47, 2016.
 - [41] C. Yu, L. Kelley, S. Zheng, and Y. Tan, "Fireworks algorithm with differential mutation for solving the CEC 2014 competition problems," in *Proceedings of 2014 IEEE Congress on Evolutionary Computation (CEC)*, pp. 3238–3245, IEEE, Beijing, China, July 2014.
 - [42] A. Sharma, R. Kumar, B. K. Panigrahi, and S. Das, "Termite spatial correlation based particle swarm optimization for unconstrained optimization," *Swarm and Evolutionary Computation*, vol. 30, pp. 93–107, 2017.
 - [43] U. Mlakar and I. Fister, "Hybrid self-adaptive cuckoo search for global optimization," *Swarm and Evolutionary Computation*, vol. 29, pp. 47–72, 2016.
 - [44] R. Tanabe and A. Fukunaga, "Evaluating the performance of SHADE on CEC 2013 benchmark problems," in *Proceedings of 2013 IEEE Congress on Evolutionary Computation (CEC)*, pp. 1952–1959, IEEE, Beijing, China, July 2014.
 - [45] Y. Tan and Y. Zhu, "Fireworks algorithm for optimization," in *Proceedings of International Conference in Swarm Intelligence*, pp. 355–364, Springer Berlin Heidelberg, Beijing, China, June 2010.
 - [46] A. H. Gandomi, X.-S. Yang, and A. H. Alavi, "Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems," *Engineering with Computers*, vol. 29, no. 1, pp. 17–35, 2013.