ARTICLE IN PRESS

ISA Transactions xxx (xxxx) xxx

Contents lists available at ScienceDirect

ISA Transactions

journal homepage: www.elsevier.com/locate/isatrans

Research article

Development and applications of an intelligent crow search algorithm based on opposition based learning

Shalini Shekhawat [a], Akash Saxena [b,*]

[a] Department of Mathematics, Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur, 302017, India
[b] Department of Electrical Engineering, Swami Keshvanand Institute of Technology, Management & Gramothan, Jaipur, 302017, India

ARTICLE INFO

Article history:
Received 19 March 2019
Received in revised form 30 August 2019
Accepted 1 September 2019
Available online xxxx

Keywords:
CSA
Metaheuristic algorithms
Benchmark functions
Bio-inspired algorithms

ABSTRACT

Metaheuristics are proven beneficial tools for solving complex, hard optimization problems. Recently, a plethora of work has been reported on bio inspired optimization algorithms. These algorithms are mimicry of behavior of animals, plants and processes into mathematical paradigms. With these developments, a new entrant in this group is Crow Search Algorithm (CSA). CSA is based on the strategic behavior of crows while searching food, thievery and chasing behavior. This algorithm sometimes suffers with local minima stagnation and unbalance exploration and exploitation phases. To overcome this problem, a cosine function is proposed first, to accelerate the exploration and retard the exploitation process with due course of the iterative process. Secondly the opposition based learning concept is incorporated for enhancing the exploration virtue of CSA. The evolved variant with the inculcation of these two concepts is named as Intelligent Crow Search Algorithm (ICSA). The algorithm is benchmarked on two benchmark function sets, one is the set of 23 standard test functions and another is set of latest benchmark function CEC-2017. Further, the applicability of this variant is tested over structural design problem, frequency wave synthesis problem and Model Order Reduction (MOR). Results reveal that ICSA exhibits competitive performance on benchmarks and real applications when compared with some contemporary optimizers.

1. Introduction

Improvement is the essence of success. In improving or designing of any system or products we require a deep knowledge of all the factors those affect the performance and produce the best results out of available resources. For achieving this, a systematic decision making technique is needed. Optimization techniques provide a key tool in making the best decisions within a set of constraints. Basically optimization is a branch of mathematics which was developed at the time of Second World War, when teams of experts from different fields like science, mathematics, psychology, economics and engineers used it, in analyzing and decision making. Their success attracts the world towards this field and Since then, different methods were developed with every passing year to apply in business and industries for improvement in production, decreasing the cost of any product or in various decision making tools. Today a plethora of literature is available on optimization and its applications.

Optimization is a frequent and fundamentally applied technique for most engineering activities. Optimization can also be done by hit and trial method, but the success of this process can questionable. By choosing a systematic and efficient approach to the task we can find a better solution of an engineering problem, which could not be found in any other way. For applying a systematic approach, to an optimization process one has a deep knowledge of following elements:

1. Objective function, which represents quantitative measurements that have to be maximized or minimized.
2. A set of inequalities and constraints that provide us a region, which may be feasible or unfeasible and create limits for system performance.
3. Variables, that represents all the factors affecting the mathematical model and satisfy the constraints.

A wide range of problems and practical applications are available in the literature which can be solved or improved with the help of optimization. For this purpose, a working knowledge of different levels of optimization is needed. At the level of mathematical programming the focus should be on the formulation of optimization problem, including all the affecting parameters, constraints, their stability and convergence rate. At computing level, the selection of "right" optimization method for efficient and best solution is important. A keen knowledge of the software used, is also required to avoid computational complexity. At the

* Corresponding author.
E-mail addresses: drshalini@skit.ac.in (S. Shekhawat), aakash.saxena@hotmail.com, akash@skit.ac.in (A. Saxena).

https://doi.org/10.1016/j.isatra.2019.09.004
0019-0578/

engineering level, the right optimization algorithms should be applied to real world practical problems to diagnose the solution or for improvement.

In recent years, Metaheuristic optimization algorithms have replaced classical methods (direct search method and gradient search method) of optimization due to their superior abilities. Direct search method is computationally slow due to large number of iterations, whereas Gradient search methods have failed in the case of discontinuous or non-differentiable functions. Also, the classical methods often get trapped in local minima and thus could not find the optimal solution easily. Meta-heuristic algorithms can be categorized in the following way:

1. Evolutionary based algorithms (EA): These algorithms are based on the principle of evolution. In EA's every randomly selected population is improved through changes due to environmental and genetic mutations and a better and advanced population exist in new generation. Some popular algorithms in this category are Genetic Algorithm (GA) [1], Genetic programming [2], Biogeography-based optimizer [3], Probability-based increment learning [4].
2. Physics based algorithms: These algorithms are developed by following the physics rules. The search space is optimized using basic physics laws, i.e. inertia force, magnetic force, gravitational force and many more. Some contemporary physics based algorithms are Big-Bang-Big-Crunch algorithm [5], Black hole algorithm [6], Ray optimization [7], Small world optimization algorithm [8], Curved space optimization [9], Central space optimization [10], Gravitational search algorithm [11] and so on.
3. Swarm Intelligence based algorithms: These types of algorithms are based on social behavior of a pack of herd or swarm. The most famous swarm based algorithm is Particle swarm optimization [12]. The main features of these algorithms are that there are very few parameters and operators, so implementation on real problems becomes very easy. Also, many animals store information about their search space and utilize it in the future, which provides an extra advantage. Ant Colony Optimization (ACO) [13], Artificial Bee Colony [14], Bat Algorithm [15], Krill herd Algorithm [16] and Firefly Algorithm [17] are some of the examples based on swarm intelligence. Recently published swarm algorithms are Grey wolf optimizer (GWO) and Grasshopper optimization algorithm (GOA), which are inspiring factor of work reported in ambient air quality classification [18] and development of enhanced chaotic grasshopper optimization algorithms (ECGOAs) [19].
4. Human based algorithms: Harmony search [20], Group search optimizer [21], Group counseling optimization algorithm [22] are some of the algorithms inspired by human behaviors.

It has been proven from no free lunch theorem [23] that any algorithm is not capable to solve all type of optimization problems hence there lies a future scope to propose a new algorithm or improve the existing one. Many algorithms improved by the researchers using different type of theories like the chaotic theory, opposition based learning and many more. Askarzadeh [24] proposed a nature inspired algorithm named Crow search algorithm (CSA) which is based on intelligent behavior of crows. CSA has a simple derivation free structure which makes it very easy to adapt. The number of control parameters are also very few, resulting a user friendly algorithm. Due to these features, researchers attracted towards it and new variants and application of CSA are unfolded.

Some modified versions of CSA have been recently developed and applied to several engineering problems. Chaotic versions of CSA have been developed in the work [25] for the feature selection problem. In that work, authors addressed the problem of feature selection by chaotic versions of CSA. An economic load dispatch problem has been addressed by the Modified CSA (MCSA) in work [26]. An another new version of CSA has been developed in [27] for prediction and usability of feature extraction. In [28] authors presented a new version of CSA named OCSA and then used it in diagnosis of Parkinson's disease. Similarly, CSA is also used for the optimal selection of conductors in a radial distribution network in [29]. Like other meta-heuristic algorithms, CSA has also some shortcomings [30]. Some of them are

- The position updating mechanism is a unidirectional search process, which can lead to local minima trapping.
- For the position updation of crow's memory, the awareness probability should be less than or equal to a random number denoted as R_i. If R_i is less than awareness probability, then memory of the crow is not updated, hence the quality of the solution is not improved.
- The convergence rate of CSA is less due to inefficient search strategy.

Motivating to remove these types of entrapment, we propose an Intelligent crow search algorithm (ICSA) in which we propose the modifications in two folds: First, we apply opposition based learning theory in the initialization phase and Second, an acceleration factor based on cosine function is in position updating phase. The original CSA, updates the positions of crows in two phases. In phase one, crows acquire random positions but in the second phase the positions are governed by awareness probability. In this phase, we introduce an acceleration factor which is adaptive and monotonically decreases as the iterative process progresses. In a way this factor shrinks the search space, which accelerates exploitation and convergence speed. In the light of above discussion, following are the major contributions of this manuscript:

1. Opposition based learning is employed in the initialization phase of CSA and an efficient cosine function is employed in position updation phase of CSA for improving the exploration and exploitation virtues of CSA.
2. The effectiveness of proposed modifications is elaborated through various statistical analyses, namely calculation of standard deviations, mean of function values of standard benchmark functions. Statistical significance is observed through Wilcoxon rank-sum test. A few analyses for advocating the superiority of the proposed variant are also incorporated namely trajectory analysis, box plot analysis, scalability test and iterative time execution analysis.
3. Further, the applicability of the proposed variant is tested over three real engineering problems, namely Three truss bar design problem, Frequency modulated sound wave parameter estimation and Model order reduction.

Further, the remainder of the paper is organized as follows: In Section 2 the background and mathematical model of CSA is described. Section 3 presents the basic theory of opposition based learning and acceleration factor and our proposed variant of CSA. Section 4 comprises the details of two sets of Benchmark function used in validation of proposed ICSA. BFS-I is a combination of 23 conventional benchmark functions. BFS-II is latest benchmark function CEC-2017. Their simulation results are also tabulated here as indicated by maximum, mean, minimum and standard deviation values. A comparison of our proposed variant ICSA with some recently developed versions is also presented here. Different analyses, including Wilcoxon rank sum test, box plot analysis, trajectory analysis, time execution analysis and convergence analysis have been also incorporated with our proposed variant in Section 5. To validate the performance of our proposed variant ICSA we solved three real world optimization problem in Section 6. At last, a brief conclusion of the paper is also presented.

2. Crow search algorithm: Background and mathematical model

Crows are a common species of birds and considered as one of the most intelligent birds. A crow has a bigger brain as compare to other birds. Literature [31] show that their brain is the biggest according to their body sizes. Crows can mimic voices, remember faces and communicate the feel of danger through sophisticated manner. They proved their cleverness in mirror-test and tool making [32]. A crow steals the food of other birds and keep it in a safe place and in this process if it is followed, it tries to make it fool by moving one position to another. In this process, they try to find the best hiding place and also make it safe from other crows, which is similar to optimization.

The four attributes of a crow's behavior are that they live in a flock (group of crow), they can memorize the position of hidden food, follow each other at the time of food theft and protect their food from other crows.

Each crow remembers its position of hiding food which can be given as M_i^t. This also denotes the best position of ith crow. According to crows behavior one crow follows another to know about the location of hidden food. Here $i = 1, \dots, P$, P is taken as number of crows/flock, position of ith crow in kth iteration is u_i^k. Let at any iteration t, ith crow want to follow crow x. Then their arise two cases:

1. Crow x does not know that it is followed by ith crow. In this case the update in position of ith crow is given by

$$u_i^{t+1} = u_i^t + R_i \times l_i^t \times \left(M_i^t - u_i^t\right) \tag{1}$$

where l denotes the flight length, R_i is a random number between [0,1] which can be lead to local search or global search according to its length.

2. Crow x knows that it is followed by ith crow and makes him a fool by moving from one position to another.

These two cases can be combined by the following equation

$$u_i^{t+1} = \begin{cases} u_i^t + R_i \times l_i^t \times \left(M_i^t - u_i^t\right) & if \quad R_i \geq AP^{i,t} \\ \text{a random position} & \text{otherwise} \end{cases} \tag{2}$$

Here, R_i is a uniformly distributed random number between 0 to 1. $AP_{i,t}$ is the awareness probability of crow x at iteration t. $AP_{i,t}$ works as a control parameter between exploration and exploitation phases. Small values of $AP_{i,t}$ show the search is in exploitation phase where larger values of $AP_{i,t}$ indicate the exploration phase.

To initialize CSA, the adjustable parameters like flock size (P), flight length (l), maximum number of iteration ($iter_{max}$) and awareness probability have given values. The initial position of each crow is randomly generated. Crow x has not the experience to hide its food and initial position of stored food is given by M. As the algorithm runs, each crow updates its positions according to Eq. (2). Finally, Memory updating of crows can be denoted as:

$$M_i^{t+1} = \begin{cases} u_i^{t+1}, & fn\left(u_i^{t+1}\right) \text{ is better than } fn\left(M_i^t\right) \\ M_i^t & \text{otherwise} \end{cases} \tag{3}$$

where $fn()$ is an objective function of the problem. The final solution is obtained, when the termination criterion is fulfilled.

3. Intelligent crow search algorithm

From the quick review presented in introduction section, it is evident that some modifications are always required to make the metaheuristic suitable for particular engineering application. Although crow search is a robust algorithm yet it suffers from local minima entrapment and poor convergence behavior in some cases. To overcome these problems, we propose two modifications in CSA. The first modification is in the initialization stage with the opposite population generation method suggested by H.R. Tizoosh [33] for better exploration capability as compared to the parent algorithm. The second modification is to employ an accelerator factor based on the cosine function for effective exploitation in the position update phase, so that convergence of the algorithm can be accelerated. We observed from the literature that adaptive position update mechanism always helps the algorithm to avoid local minima trap. Some examples of such position update mechanism are easily visible in the structure of GWO,Whale Optimization Algorithm (WOA), Gravitational Search Algorithm (GSA). This concept of position update can be classified into three subcategories:

1. Linear adaptive mechanism (WOA [34], Moth Flame Optimizer (MFO) [35], GWO [36])
2. Mechanism govern by trigonometric functions [37].
3. Exponentially decaying mechanism (GSA) [11].

From these literature evidences, authors are motivated to employ this adaptive mechanism that is governed by cosine function. It is evident that the performance of any algorithm highly depends upon the search capability and exploitation capability. For improving the search capability, opposition based learning is employed in ICSA. Again, there are many evidences and beautiful examples of the OBL inculcation for improving the performance of the algorithms [38]. Based on this fact, authors are motivated to employ OBL in CSA.

In this section, the details of these two modifications are discussed.

- Opposition Based Learning (OBL): Proper utilization of the search space is inevitable for any metaheuristic optimization algorithm to find the global optima of any given problem. The algorithm starts with random initial points, and the search is guided towards the optima by various control parameters, mechanisms and processes. In some cases, these random, initial points lead the algorithm in a local minima trap. As a solution to this problem, in 2005, H.R. Tizoosh [33] gave a theory which segregates the random population into two parts, i.e. random population and opposite population. The inculcation of this opposition based theory enables the algorithm to explore the search space effectively. The philosophy behind this theory is that a potential solution can exist in any direction. Hence, it is necessary to explore the search space in random as well as opposite direction. A review from year 2005 to year 2012 on the application of OBL for improving metaheuristics was presented in [38]. For enhancing Krill Herd (KH) Algorithm, an OBL-KH algorithm is proposed in [39]. Shan and Liu [40] enhance the diversity and convergence of Bat algorithm using OBL in 2016. The OBL theory has been applied to harmony search algorithm to increase its convergence rate in [41]. The same concept of convergence increasing capacity and increment the diversity through OBL theory has been used in [39]. In 2018 an opposition based whale optimization algorithm [42] has been proposed to improve the original algorithm in terms of solution accuracy and reliability. The successful applications of OBL for improving different meta-heuristic algorithm motivates authors to apply OBL to improving CSA. The basic concepts involved in OBL strategy are as follows:-

Definition 1. Let $y \in [p, q]$ is a real number, then opposite number of y is defined by

$$\bar{y} = p + q - y$$

Definition 2. Let $A = (y_1, y_2, \ldots, y_S)$ is a point of S dimensional space, where $y_i \in R, \forall i \in \{1, 2, \ldots, S\}$ and bounded by $[p, q]$, the opposite points matrix can be given by $\bar{A} = [\bar{y}_1, \bar{y}_2, \bar{y}_3 \ldots, \bar{y}_S]$. Hence

$$\bar{y}_i = [p_i + q_i - y_i]$$

We have applied opposition based learning for improving CSA in following manner:

1. Initialize the positions of crows in random search space. Population size and maximum no. of iterations should be kept constant.
2. Fitness of each search agent has been calculated.
3. Generate the opposite points as per Definition 2 and obtain the fitness of each search agent.
4. Perform sorting between the opposite population fitness values and original population fitness values. Based on the fitness values potential solutions are identified.
5. Modify the positions of the crows according to Definition 2.

- Effective cosine function as an accelerator factor:
  Earlier in [37] and [43] authors have used trigonometric function to fill the gap between the exploitation and exploration phase. Motivated by these results, we propose a cosine function for acceleration in the exploration phase and comparatively less swift behavior in the exploitation phase of the original algorithm. Although in many cases the exponential functions, chaotic functions and linear functions are used to bridge this mechanism but in this work we consider a cosine function, as it a has high gradient in the first phase (i.e. exploration phase) as compared with the linear function. Variation of this factor is shown in Fig. 1. It is evident from the figure that the positions of crow expanded in a larger area in the first half as compared with original CSA and area of search shrinks during the second half after using cosine function. Mathematical formulation of this function can be given as:

$$\alpha = \pi \times \frac{Current\ Iteration}{Maximum\ Iteration} \tag{4}$$

$$AF = \left(cos^2\left(\frac{\alpha}{2}\right)\right) \tag{5}$$

  Now the proposed modification is incorporated in the position update equation in phase 2.

$$u_i^{t+1} = \begin{cases} u_i^t + AF \times R_i \times l_i^t \times \left(M_i^t - u_i^t\right) & if \quad R_i \geq AP^{i,t} \\ \text{a random position} & \text{otherwise} \end{cases} \tag{6}$$

  This function is utilized in the position updating phase of the algorithm. The function decreases monotonically during the course of iteration, this behavior of function helps to promote exploitation with the faster pace as compared with linear variations as iteration count increases. In the original CSA, this function is missing due to which the exploration and exploitation phases are not tuned. This causes CSA to suffer with local minima entrapment. As per this discussion, both above mentioned virtues will help the CSA to converge in lesser time and exploring the search space effectively.
- Step by step progression of ICSA are described as follows:

Step 1: The problem is initialized and different parameters like number of crows in a flock (P), flight length (l), maximum number of iterations ($iter_{max}$) are chosen.

Step 2: The total no of crows from a flock size is divided into two parts. The first part includes (P/2) crows, whose positions are generated randomly. Remaining (P/2) crows generate their position according to Definition 2 in Section 3.

Step 3: The position of each crow are calculated through decision variables and objective function.

Step 4: The new positions of crows generated in two ways as according to Eq. (6). To generate the position, crow i randomly choose another crow j and follows it to know about the location of hidden food. The obtained positions in this phase are accelerated by proposed acceleration factor defined in (4) and (5).

Step 5: The feasibility of the positions of the crows is checked. The position has remained unaltered if it is not feasible.

Step 6: The position of each crow is evaluated in terms of the fitness function.

Step 7: Every crow updates its memory using Eq. (3) and if it is better than earlier saved position in the memory, it is updated.

Step 8: Step 4 to 7 are repeated till the maximum number of iterations is reached. After completing the total iterations, we get the final position of crow as the optimal solution. For more clarity, flow chart of the proposed variant is shown in Fig. 2.

4. Benchmark functions and result analysis

Benchmark functions are set of diverse test functions in nature and properties. These functions are useful in measuring the performance of any algorithm and to compare the results of an algorithm with others. These functions can be further classified in terms of modality, separability or in dimensionality. In this section we present two sets of benchmark functions and results of ICSA on these benchmark functions.

4.1. Benchmark functions set I (BFS-I)

Benchmark Functions set I is a rich set of benchmark functions available in [36], out of which we have chosen 23 test functions. Functions F-1 to F-7 are unimodal, F-8 to F-13 are multimodal and remaining functions are given in Appendix (Table 13) are multimodal functions with fixed dimensions. The unimodal functions have only one minima and used to test the exploitation capability of an algorithm, on the other hand, the multimodal functions have multiple local minimas and these functions test the exploration feature of an algorithm. 2D shapes of 20 functions are shown in Fig. 3 for clarity.

4.2. Discussion of results on BFS-I

Here, we present the results of our proposed algorithm on chosen set of 23 test functions, that are averaged over 30 independent runs. These results are presented in terms of Minimum (Min), Maximum (Max), Mean and Standard deviation(SD) values. The results on benchmark functions of our proposed variant of CSA and original CSA are depicted in Table 1. Table 2 presents the comparison between our proposed variant and some other well-known contemporary algorithms. For the simulation results depicted in Tables 1 and 3, the maximum function evaluation is kept as termination criterion which is equal to 3×10^4. Awareness probability is set to 0.1 and flight length is considered as 2 as per the results reported in [24].

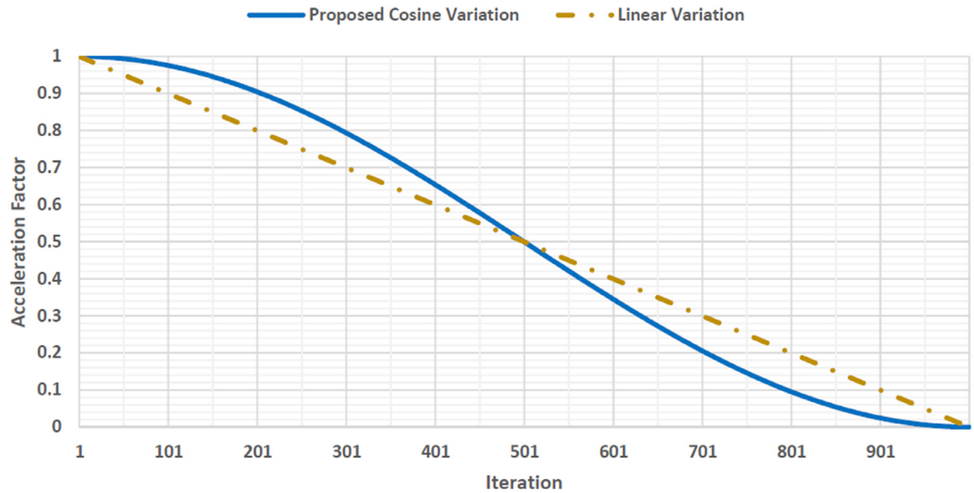

Fig. 1. Acceleration factor.

Table 1
Results on BFS-I.

Function	CSA [24]				ICSA				
	Min	Max	Mean	SD	Min	Max	Mean	SD	p-values
F-1	4.66E−02	5.21E−01	2.04E−01	1.21E−01	**9.44E−61**	**1.75E−60**	**1.33E−60**	**2.26E−61**	3.02E−11
F-2	4.27E+00	1.64E+01	8.92E+00	3.29E+00	2.06E+00	**3.54E+00**	**2.57E+00**	**3.39E−01**	3.02E−11
F-3	6.13E+01	2.54E+02	1.30E+02	3.96E+01	**1.63E−60**	**3.77E−05**	**1.36E−06**	**6.87E−06**	3.02E−11
F-4	2.49E+00	7.59E+00	5.23E+00	1.33E+00	**2.92E−31**	**2.45E−04**	**5.39E−05**	**9.95E−05**	3.02E−11
F-5	3.29E+01	6.98E+02	1.48E+02	1.36E+02	2.56E+01	**2.90E+01**	**2.88E+01**	**5.98E−01**	3.02E−11
F-6	5.03E−02	3.87E−01	1.87E−01	1.05E−01	**9.99E−03**	1.97E−01	**7.78E−02**	**4.14E−02**	1.61E−06
F-7	1.78E−02	7.21E−02	4.03E−02	1.44E−02	**1.89E−04**	**5.17E−03**	**1.50E−03**	**1.52E−03**	3.02E−11
F-8	−8.06E+03	−4.33E+03	−6.68E+03	8.32E+02	**−8.27E+03**	**−5.37E+03**	**−6.70E+03**	6.97E+02	8.07E−01
F-9	1.30E+01	5.68E+01	2.90E+01	1.09E+01	**0.00E+00**	**3.12E−07**	**2.91E−08**	**8.91E−08**	3.16E−12
F-10	2.60E+00	7.55E+00	4.09E+00	1.24E+00	**8.88E−16**	**1.79E−04**	**1.16E−05**	**4.43E−05**	2.37E−12
F-11	1.52E−01	7.01E−01	4.22E−01	1.32E−01	0.00E+00	**1.63E−05**	**5.44E−07**	**2.98E−06**	1.72E−12
F-12	2.52E+00	9.27E+00	5.00E+00	1.77E+00	**1.19E−03**	**8.20E−01**	**2.27E−01**	**1.57E−01**	3.02E−11
F-13	1.81E−01	5.54E+01	4.64E+00	1.04E+01	**9.66E−03**	**1.58E+00**	**7.41E−01**	**4.61E−01**	8.68E−03
F-14	9.98E−01	9.98E−01	9.98E−01	3.60E−13	9.98E−01	2.98E+00	1.16E+00	4.58E−01	6.29E−01
F-15	3.07E−04	1.59E−03	3.89E−04	2.85E−04	3.07E−04	**4.24E−04**	3.11E−04	**2.13E−05**	5.56E−04
F-16	−1.03E+00	−1.03E+00	−1.03E+00	5.68E−16	−1.03E+00	−1.03E+00	−1.03E+00	5.68E−16	1.00E+00
F-17	3.30E+00	5.49E+01	5.19E+00	3.15E+00	**3.99E−01**	**4.12E−01**	**3.28E−01**	**2.20E−01**	5.60E−02
F-18	3.00E+00	3.00E+00	3.00E+00	1.91E−15	3.00E+00	3.00E+00	3.00E+00	2.74E−15	9.94E−01
F-19	−3.86E+00	−3.86E+00	−3.86E+00	2.58E−15	−3.86E+00	−3.86E+00	−3.86E+00	**2.46E−15**	3.96E−02
F-20	−3.32E+00	−3.20E+00	−3.29E+00	5.35E−02	−3.32E+00	−3.20E+00	−3.31E+00	**4.12E−02**	7.29E−03
F-21	−1.02E+01	−2.68E+00	**−9.15E+00**	2.58E+00	−1.02E+01	**−5.06E+00**	−9.98E+00	**9.31E−01**	5.03E−01
F-22	−1.04E+01	−2.77E+00	−1.00E+01	1.57E+00	−1.04E+01	**−3.72E+00**	**−9.65E+00**	**1.97E+00**	3.62E−02
F-23	−1.05E+01	−2.43E+00	−9.56E+00	2.53E+00	−1.05E+01	**−5.13E+00**	**−1.04E+01**	**9.87E−01**	9.81E−01

4.2.1. Evaluation of exploitation capacity through unimodal functions

Functions depicted in Table 1 i.e. (F-1 to F-7) are unimodal functions. Unimodal functions have only one global optima and due to this feature they are suitable to test the exploitation capacity of any algorithm. It is evident from Table 1 that minimum, maximum, mean and standard deviation values all are optimum for proposed ICSA than the original algorithm for these functions. This indicates a significant performance of ICSA in terms of exploitation capabilities. For functions F-1 to F-4 our proposed variant ICSA exactly reaches to global optima. Standard deviation values are also comparatively very low than original CSA, that shows enhancement of stability and robustness of the proposed algorithm.

4.2.2. Evaluation of exploration capacity through multimodal functions

The exploration capacity of any algorithm is measured through multimodal functions as these functions have many local optimas. These optimas increase exponentially with the size of the problem. The benchmarking results of multimodal functions F-8 to F-13 are exhibited in Table 1. Lower values of all parameters show that our combined theory of opposition based learning and the cosine function efficiently increase the exploration capacity of the algorithm without entrapping in local minima.

4.2.3. Evaluation of exploration capacity through fixed dimension multimodal functions

Functions F-14 to F-23 are multimodal functions with fixed dimensions. The minimum values of these functions are equal to original algorithm but maximum values for F-21 to F-23, mean for F-22 and F-23 and standard deviation of F-19 to F-23 are optimal that shows ICSA outperforms original CSA overall.

4.3. Comparison of ICSA with other variants of CSA on BFS-I

We have presented the comparison of optimization performance of ICSA with two recently published versions of CSA here. The first opponent is a chaotic version of CSA. In this variant different chaotic maps have been employed in the position updation phase. On the basis of the results reported in that approach we have chosen best chaotic variant which is based on the sine

Please cite this article as: S. Shekhawat and A. Saxena, Development and applications of an intelligent crow search algorithm based on opposition based learning. ISA Transactions (2019), https://doi.org/10.1016/j.isatra.2019.09.004.

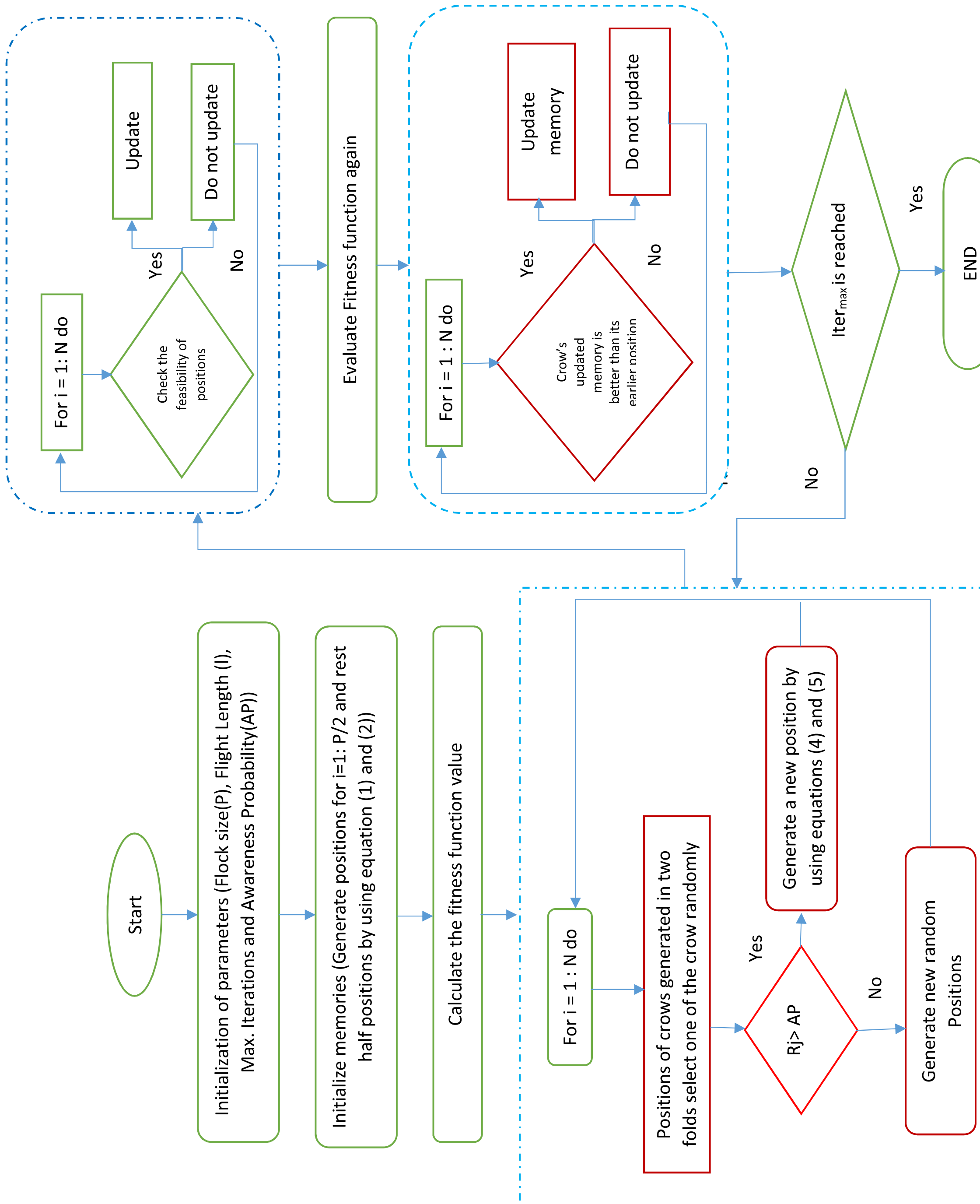

Fig. 2. Flow chart of proposed ICSA.

map for the sake of comparison with proposed ICSA [25]. The second opponent optimizer is the modified version of crow search algorithm where the adaptive flight length has been incorporated along with Levy flight movement to enhance the search capacities of the original random perturbation [30]. The optimization process is carried out and the results are reported in Table 3. For reporting the results, we calculate four statistical attributes of the function values of 30 independent runs. These attributes are:

- Maximum
- Minimum
- Mean
- Standard deviation

The stopping criterion for this optimization is maximum iteration. The results for 30 dimension problems are reported in Table 3, indicate that for almost all the functions mean values are optimal for proposed ICSA. Similarly, other statistical attributes maximum, minimum, and standard deviation are also quite competitive with other opponents. These results clearly indicate and advocate the applicability of the proposed ICSA on unimodal, multimodal and fixed dimension problems over the recently published variants of CSA.

4.4. Benchmark function set-II (BFS-II)

To validate the performance of ICSA, benchmarking of the ICSA is done over the set of latest benchmark function set congress on

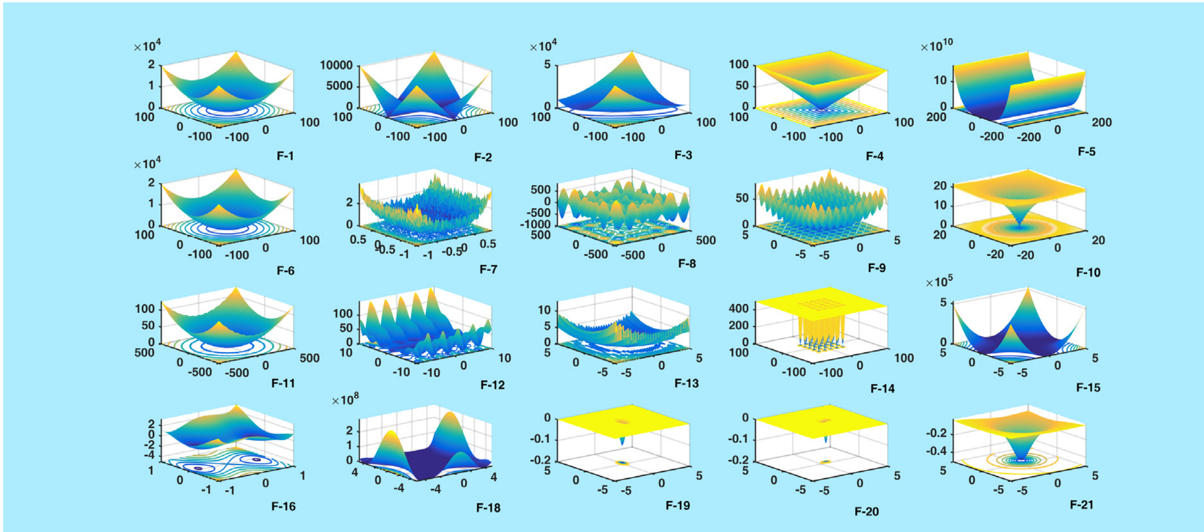

Fig. 3. 2D-shapes of BFS-I.

Table 2
Comparison of ICSA with other algorithms on BFS-I.

	ICSA		β-GWO [44]		GWO [36]		GOA [45]		WOA [34]		MFO [35]	
	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean	SD	Mean
F-1	2.26E−61	1.33E−60	2.23E−28	1.13E−28	1.02E−27	8.32E−28	0.00E+00	0.00E+00	4.91E−30	1.41E−30	1.50E−04	1.50E−04
F-2	3.39E−01	2.57E+00	1.68E−17	2.02E−17	9.14E−17	1.10E−16	1.00E−03	2.00E−03	2.39E−21	1.06E−30	8.77E−04	6.39E−04
F-3	6.87E−06	1.36E−06	2.85E−05	9.39E−06	2.77E−04	7.61E−05	2.03E−02	1.00E−03	2.93E−06	5.39E−07	1.89E+02	6.97E+02
F-4	9.95E−05	5.39E−05	4.31E−07	3.85E−07	7.79E−07	8.61E−07	0.00E+00	0.00E+00	3.97E−01	7.26E−02	5.28E+00	7.07E+01
F-5	5.98E−01	2.88E+01	6.20E−01	2.72E+01	8.11E−01	2.74E+01	0.00E+00	0.00E+00	7.64E−01	2.79E+01	1.20E+02	1.39E+02
F-6	4.14E−02	7.78E−02	2.99E−01	6.45E−01	4.40E−01	8.49E−01	0.00E+00	0.00E+00	5.32E−01	3.12E+00	9.87E−05	1.13E−04
F-7	1.52E−03	1.50E−03	7.77E−04	1.80E−03	1.12E−03	1.97E−03	0.00E+00	0.00E+00	1.15E−03	1.43E−03	4.64E−02	9.12E−02
F-8	6.97E+02	−6.70E+03	8.03E+02	−6.11E+03	9.61E+02	−5.97E+03	2.00E−04	1.00E+00	6.96E+02	−5.08E+03	7.26E+02	−8.50E+00
F-9	8.91E−08	2.91E−08	4.78E+00	2.19E+00	2.91E+00	2.90E+00	7.00E−04	0.00E+00	0.00E+00	0.00E+00	1.62E+01	8.46E+01
F-10	4.43E−05	1.16E−05	1.03E−14	7.89E−14	1.79E−14	9.95E−14	1.00E+00	9.75E−02	9.90E+00	7.40E+00	7.30E−01	1.26E+00
F-11	2.98E−06	5.44E−07	9.29E−03	5.87E−03	8.04E−03	4.79E−03	0.00E+00	0.00E+00	1.59E−03	2.89E−04	2.17E−02	1.91E−02
F-12	1.57E−01	2.27E−01	2.17E−02	4.03E−02	2.33E−02	4.79E−02	7.00E−04	0.00E+00	2.15E−01	3.40E−01	8.81E−01	8.94E−01
F-13	4.61E−01	7.41E−01	2.18E−01	5.43E−01	2.10E−01	6.19E−01	0.00E+00	0.00E+00	2.66E−01	1.89E+00	1.93E−01	1.16E−01
F-14	4.58E−01	1.16E+00	3.76E+00	3.91E+00	3.52E+00	3.42E+00	5.30E−16	9.98E−01	2.50E+00	2.11E+00	1.61E+00	1.95E+00
F-15	2.13E−05	3.11E−04	8.10E−03	4.44E−03	8.08E−03	4.49E−03	9.47E−03	1.05E−02	3.24E−04	5.72E−04	3.56E−03	1.61E−03
F-16	5.68E−16	−1.03E+00	1.49E−07	−1.03E+00	2.17E−08	−1.03E+00	2.91E−13	−1.03E+00	4.20E−07	−1.03E+00	6.78E−16	−1.03E+00
F-17	2.20E−01	3.28E−01	3.49E−05	3.98E−01	1.17E−04	3.98E−01	1.99E−13	3.98E−01	2.70E−05	3.98E−02	0.00E+00	3.98E−01
F-18	2.74E−15	3.00E+00	4.10E−05	3.00E+00	4.93E−05	3.00E+00	2.48E−12	3.00E+00	4.22E−15	3.00E+00	1.59E−15	3.00E+00
F-19	2.46E−15	−3.86E+00	2.62E−03	−3.86E+00	2.97E−03	−3.86E+00	2.65E−01	−3.73E+00	2.71E−03	−3.86E+00	1.44E−03	−3.86E+00
F-20	4.12E−02	−3.31E+00	7.69E−02	−3.23E+00	7.45E−02	−3.26E+00	6.13E−02	−3.28E+00	3.77E−01	2.98E+00	6.46E−02	−3.24E+00
F-21	9.31E−01	−9.98E+00	2.44E+00	−8.96E+00	2.42E+00	−8.86E+00	2.93E+00	−5.30E+00	3.63E+00	−7.05E+00	3.21E+00	−6.47E+00
F-22	1.97E+00	−9.65E+00	1.31E+00	−1.01E+01	1.18E−03	−1.04E+01	3.69E+00	−6.36E+00	3.83E+00	−8.18E+00	3.39E+00	−8.08E+00
F-23	9.87E−01	−1.04E+01	1.48E+00	−1.02E+01	1.48E+00	−1.03E+01	3.67E+00	−5.60E+00	2.41E+00	−9.34E+00	3.62E+00	−7.27E+00

evolutionary computation 2017 (CEC-2017). These functions are defined in Appendix (Table 14). This table presents properties like composition and search space properties and other important information related to the characteristics of functions. From table, it can be seen that this set consists of 29 diverse characteristics functions. These functions are unimodal, multimodal, hybrid and composite functions. For more details of these functions readers are directed to [46]. In this regard, statistical analysis of the ICSA results are presented with some of the recently reported approaches and contemporary optimizers. These optimizers are namely: β-GWO [44] GWO [36] GOA [45] MFO [35] CSA [24] and WOA [34]. While compiling these results, we have strictly followed the guidelines issued by the CEC-17, such as, results are averaged over 51 independent runs, maximum function evaluations are considered as $10^4 \times$ D. In this experimental set, we have considered only 30 D functions, no. of search agents are kept same for all the competitors. Due to the sake of clarity, results of these simulations are depicted in two tables. For displaying the results of first 14 functions, Table 4 is utilized and for rest of the functions Table 5 can be seen.

Results of the first 14 benchmark functions are depicted in Table 4. In this table statistical attributes namely mean and standard deviation of the independent runs that depict the solution quality of the optimization process are presented. Likewise, the results of the rest of functions, are depicted in Table 5 in terms of mean and SD values. Along with the statistical analysis, we have performed statistical significance test, named as Wilcoxon rank sum test [47]. The results of this test will be discussed in the algorithm analysis section. Following conclusions can be drawn from the statistical attributes calculations.

- Inspecting the results of Table 4, We have observed that the mean values are optimal for 13 functions. Mean values that are optimal for ICSA are shown in bold face. ICSA shows very competitive results in the case of function CECF1, as other opponents are getting the function values very higher the ICSA and CSA are quite competitive for this function.

Table 3
Comparison of ICSA with other CSA.

Variants	ImCSA [30]				CCSA [25]				ICSA			
Statistical values	Min	Max	Mean	SD	Min	Max	Mean	SD	Min	Max	Mean	SD
F-1	0.00E+00	8.71E−03	2.94E−04	1.59E−03	3.35E+04	4.58E+04	4.05E+04	3.44E+03	9.44E−61	1.75E−60	**1.33E−60**	2.26E−61
F-2	−3.99E+00	8.10E−01	−2.22E+00	1.18E+00	1.28E+03	3.44E+06	7.11E+05	1.05E+06	2.06E+00	3.54E+00	**2.57E+00**	3.39E−01
F-3	1.80E+00	7.23E+03	2.57E+03	2.16E+03	3.29E+04	5.38E+04	4.54E+04	5.29E+03	1.63E−60	3.77E−05	**1.36E−06**	6.87E−06
F-4	6.27E−06	1.52E−03	3.99E−04	4.17E−04	6.04E+01	7.56E+01	6.99E+01	3.03E+00	2.92E−31	2.45E−04	**5.39E−05**	9.95E−05
F-5	3.13E−08	2.98E+01	1.92E+01	1.38E+01	5.58E+07	1.25E+08	9.54E+07	1.90E+07	2.56E+01	2.90E+01	**2.88E+01**	5.98E−01
F-6	8.55E−10	6.32E+00	2.11E−01	1.15E+00	3.49E+04	4.60E+04	3.99E+04	3.05E+03	9.99E−03	1.97E−01	**7.78E−02**	4.14E−02
F-7	1.50E−04	1.54E−02	1.55E−03	2.74E−03	2.17E+01	5.22E+01	4.35E+01	6.54E+00	1.89E−04	5.17E−03	**1.50E−03**	1.52E−03
F-8	−9.02E+03	−2.74E+03	−5.23E+03	2.36E+03	−4.85E+03	−3.87E+03	−4.30E+03	2.44E+02	−8.27E+03	−5.37E+03	**−6.70E+03**	6.97E+02
F-9	1.25E−08	2.98E+01	9.95E−01	5.45E+00	3.03E+02	3.72E+02	3.38E+02	1.42E+01	0.00E+00	3.12E−07	**2.91E−08**	8.91E−08
F-10	5.29E−06	3.75E−03	7.43E−04	8.13E−04	1.88E+01	1.99E+01	1.96E+01	2.60E−01	8.88E−16	1.79E−04	**1.16E−05**	4.43E−05
F-11	5.09E−12	9.90E−01	1.60E−01	3.62E−01	2.56E+02	4.16E+02	3.58E+02	3.81E+01	0.00E+00	1.63E−05	**5.44E−07**	2.98E−06
F-12	7.73E−11	1.91E+01	9.65E+00	5.72E+00	9.74E+07	2.17E+08	1.63E+08	3.35E+07	1.19E−03	8.20E−01	**2.27E−01**	1.57E−01
F-13	3.23E−10	6.11E+01	1.56E+01	1.65E+01	1.73E+08	5.26E+08	3.66E+08	8.24E+07	9.66E−03	1.58E+00	**7.41E−01**	4.61E−01
F-14	9.98E−01	9.98E−01	9.98E−01	1.36E−12	9.98E−01	1.28E+00	1.02E+00	5.74E−02	9.98E−01	2.98E+00	**1.16E+00**	4.58E−01
F-15	3.07E−04	1.59E−03	3.82E−04	2.86E−04	9.12E−04	3.10E−03	1.99E−03	5.97E−04	3.07E−04	4.24E−04	**3.11E−04**	2.13E−05
F-16	−1.03E+00	−1.03E+00	−1.03E+00	6.71E−16	−1.03E+00	−1.02E+00	−1.03E+00	3.27E−03	−1.03E+00	−1.03E+00	**−1.03E+00**	5.68E−16
F-17	1.12E+01	6.67E+01	3.34E+01	1.87E+00	3.14E+02	7.87E+02	5.56E+02	5.16E+02	2.19E−01	4.12E−01	**3.28E−01**	2.20E−01
F-18	3.00E+00	3.00E+00	3.00E+00	2.33E−10	3.00E+00	3.21E+00	3.06E+00	5.39E−02	3.00E+00	3.00E+00	**3.00E+00**	2.74E−15
F-19	−3.82E+00	−2.65E+00	−3.57E+00	2.79E−01	−3.86E+00	−3.85E+00	−3.86E+00	2.40E−03	−3.86E+00	−3.86E+00	**−3.86E+00**	2.46E−15
F-20	−2.94E+00	−1.63E+00	−2.27E+00	3.71E−01	−3.25E+00	−2.94E+00	−3.08E+00	6.76E−02	−3.32E+00	−3.20E+00	**−3.31E+00**	4.12E−02
F-21	−1.02E+01	−1.02E+01	−1.02E+01	3.06E−05	−7.50E+00	−2.40E+00	−4.03E+00	1.36E+00	−1.02E+01	−5.06E+00	**−9.98E+00**	9.31E−01
F-22	−1.04E+01	−1.04E+01	−1.04E+01	6.43E−05	−7.27E+00	−2.49E+00	−4.06E+00	1.11E+00	−1.04E+01	−3.72E+00	**−9.65E+00**	1.97E+00
F-23	−1.05E+01	−1.05E+01	−1.05E+01	6.48E−05	−6.93E+00	−2.68E+00	−4.11E+00	9.78E−01	−1.05E+01	−5.13E+00	**−1.04E+01**	9.87E−01

Similar performance can be observed for functions CECF4, 12, 13 and 15. These optimal mean values indicate that optimization process is handled very well by the proposed ICSA. It is evident that for some functions this variant completely outperforms and in case of rest of the functions performance is quite competitive.

- After careful inspection of the results depicted in Table 5, we have arrived to the conclusion that proposed variant is capable to handle complex optimization problems as the mean values are optimal for 10 functions.
  We also observed that for CECF18-19 mean values are quite competitive and ICSA and CSA outperforms other optimizers. Similar observations can be noted from the function CECF30. Based on these results, it is quite evident that proposed variant exhibits superior performance on complex functions like CEC17. Superior performance of ICSA as compared to contemporary optimizers, motivated authors to conduct algorithm analysis.

After conducting series of experiments, we have arrived on the conclusion that proposed variant exhibits excellent properties when it is benchmarked over conventional and latest set of CEC-17 functions. Further, the comparisons based on the calculations of mean and SD values can be misleading sometimes or in other words, algorithm analysis in terms of statistical significance, data distribution arrangement is required to test the performance of the algorithm statistically. Along with that many other analyses such as trajectory, iterative time execution analysis is also required to arrive on a final conclusion. Following section presents algorithm analysis.

5. Algorithm analysis

In this section, we present different analyses in favor of our proposed algorithm. First is Wilcoxon rank sum analysis test [47], which is used to test statistical significance of two pairwise samples. Second is boxplot analysis, which efficiently analyze the distribution of data. Third is trajectory analysis, of original CSA and our proposed variant for some functions. Then, we have presented a comparison of execution time taken in each iteration in both the algorithms. At last, analysis of convergence characteristics for some function has also been plotted.

5.1. Wilcoxon ranksum analysis

The results of the Wilcoxon rank sum test are exhibited in Table 1 for benchmark function set-I and Table 4. We have carried out this test on 5% significance level. The results of this test are p-values that suggests, if the p-values of two distributions are less than 0.05, these two distributions are statistically different. In a precise manner the results which have p-values greater than 0.05 are statistically same. While compiling these results, we have compared proposed ICSA with other optimizers for both set of functions. Following conclusions can be drawn from this analysis.

- Inspecting the results of this test on BFS-I, it can be concluded that the majority of the p-values for ICSA and CSA are less than 0.05, with p-values less than 0.05 and with optimal values of SD and mean for the majority of the benchmark functions, it can be concluded that ICSA is significantly performed better than CSA on these benchmark functions. The opposition theory and acceleration factor are significantly enhancing the performance of ICSA.
- Inspecting the results of this test on BFS-II in Table 4 we have observed that the p-values are less than 0.05 for almost all the functions for β-GWO [44], GOA [45], MFO [35], WOA [34] and for the case of GWO [36] CSA [24] the performance of proposed variant is competitive. With low values of mean and p-values less than 0.05 it is concluded that proposed variant exhibits superior optimization performance.
- From the results depicted in the Table 5, it can be concluded that the performance of proposed variant is statistically significant for almost all the functions for β-GWO [44], GOA [45], MFO [35], WOA [34] and comparable to CSA in some cases.

To test the applicability and data distribution of the results box plot analysis is carried out in next subsection.

5.2. Boxplot analysis

Boxplot analysis is a tool to judge the data agreement between different independent runs. The box plot analysis indicates the solution quality of the optimizer. In this analysis we have plotted the fitness values obtained from independent runs.

Table 4
Comparison of ICSA with other algorithms on (BFS-II) CEC-2017-1.

Function	Parameters	β-GWO [44]	GWO [36]	GOA [45]	MFO [35]	CSA [24]	WOA [34]	ICSA
CECF1	mean	8.35E+08	1.05E+09	2.91E+09	7.35E+09	4.16E+03	1.26E+09	**3.33E+03**
	SD	6.84E+08	9.51E+08	2.25E+09	4.09E+09	4.68E+03	1.23E+09	3.29E+03
	p-values	**3.30E−18**	**2.75E−02**	**3.99E−17**	**3.30E−18**	9.04E−01	**3.30E−18**	N/A
CECF3	mean	2.51E+04	2.72E+04	1.11E+05	8.34E+04	3.09E+02	4.67E+04	**3.09E+02**
	SD	7.11E+03	9.31E+03	5.82E+04	5.60E+04	8.57E+00	1.17E+04	1.02E+01
	p-values	**3.30E−18**	2.84E−01	**3.30E−18**	**3.30E−18**	7.13E−01	**3.30E−18**	N/A
CECF4	mean	5.48E+02	5.52E+02	7.06E+02	9.75E+02	5.03E+02	6.19E+02	**4.94E+02**
	SD	4.16E+01	4.64E+01	1.83E+02	5.71E+02	3.45E+01	8.80E+01	3.40E+01
	p-values	**7.09E−10**	7.68E−01	**3.39E−16**	**1.13E−17**	1.34E−01	**2.43E−16**	N/A
CECF5	mean	5.83E+02	5.82E+02	6.69E+02	6.82E+02	6.29E+02	7.64E+02	**6.29E+02**
	SD	3.36E+01	2.78E+01	4.96E+01	3.82E+01	2.76E+01	5.64E+01	3.08E+01
	p-values	**2.29E−11**	6.20E−01	**1.17E−03**	**5.10E−08**	9.04E−02	**2.26E−17**	N/A
CECF6	mean	6.24E+02	6.04E+02	6.48E+02	6.24E+02	6.30E+02	6.65E+02	6.20E+02
	SD	2.12E+00	2.38E+00	1.48E+01	7.89E+00	7.12E+00	1.22E+01	7.14E+00
	p-values	**3.30E−18**	4.34E−01	**4.62E−10**	**4.72E−05**	8.51E−01	**6.30E−18**	N/A
CECF7	mean	8.30E+02	8.36E+02	8.76E+02	9.78E+02	8.76E+02	1.15E+03	**8.26E+02**
	SD	2.99E+01	3.69E+01	4.70E+01	1.14E+02	4.58E+01	7.81E+01	3.48E+01
	p-values	**4.55E−08**	5.47E−01	3.06E−01	**1.13E−09**	3.35E−01	**3.30E−18**	N/A
CECF8	mean	8.69E+02	8.75E+02	9.49E+02	9.88E+02	9.11E+02	1.02E+03	**8.66E+02**
	SD	1.27E+01	1.82E+01	4.23E+01	4.54E+01	1.89E+01	5.79E+01	2.26E+01
	p-values	**1.21E−13**	9.56E−02	**5.92E−08**	**8.26E−15**	2.96E−01	**6.31E−17**	N/A
CECF9	mean	1.28E+03	1.43E+03	5.27E+03	5.69E+03	1.88E+03	7.57E+03	**1.28E+03**
	SD	2.60E+02	4.33E+02	2.55E+03	1.41E+03	5.35E+02	2.67E+03	4.09E+02
	p-values	**4.26E−10**	8.91E−02	**1.85E−15**	**4.70E−18**	1.39E−01	**3.72E−18**	N/A
CECF10	mean	3.96E+03	3.77E+03	5.50E+03	5.40E+03	4.52E+03	5.88E+03	**3.70E+03**
	SD	1.08E+03	6.96E+02	6.50E+02	8.05E+02	5.66E+02	7.72E+02	5.74E+02
	p-values	**4.55E−09**	5.92E−01	**5.10E−08**	**2.56E−05**	1.07E−01	**1.05E−11**	N/A
CECF11	mean	1.37E+03	1.37E+03	3.11E+03	2.83E+03	1.24E+03	1.95E+03	**1.24E+03**
	SD	2.03E+02	1.59E+02	2.01E+03	2.40E+03	3.47E+01	8.68E+02	5.12E+01
	p-values	**2.20E−08**	6.68E−01	**3.71E−18**	**2.57E−16**	2.21E−01	**1.27E−17**	N/A
CECF12	mean	4.05E+07	4.07E+07	1.50E+08	1.55E+08	1.01E+06	5.71E+07	**9.54E+05**
	SD	4.92E+07	5.53E+07	2.29E+08	2.05E+08	7.98E+05	6.45E+07	6.57E+05
	p-values	**2.13E−17**	8.99E−01	**3.94E−18**	**2.06E−16**	9.36E−01	**3.30E−18**	N/A
CECF13	mean	1.58E+07	9.84E+05	2.00E+07	6.24E+06	3.01E+04	1.31E+05	**3.01E+04**
	SD	5.84E+07	4.19E+06	3.07E+07	1.93E+07	1.56E+04	7.27E+04	1.64E+04
	p-values	**3.14E−13**	7.89E−01	**4.47E−17**	**7.09E−11**	8.46E−01	**4.01E−16**	N/A
CECF14	mean	7.24E+04	8.07E+04	1.01E+05	1.08E+05	1.63E+03	3.52E+05	**1.63E+03**
	SD	1.12E+05	1.45E+05	1.94E+05	1.54E+05	6.13E+01	5.28E+05	6.66E+01
	p-values	**3.30E−18**	8.20E−01	**3.29E−18**	**3.30E−18**	7.94E−01	**3.30E−18**	N/A
CECF15	mean	1.26E+05	2.67E+05	1.21E+05	3.81E+04	8.94E+03	4.19E+05	**8.04E+03**
	SD	4.06E+05	5.85E+05	7.50E+04	2.79E+04	3.98E+03	1.22E+06	3.03E+03
	p-values	**2.13E−17**	3.35E−01	**3.30E−18**	**1.91E−13**	4.45E−01	**3.30E−18**	N/A

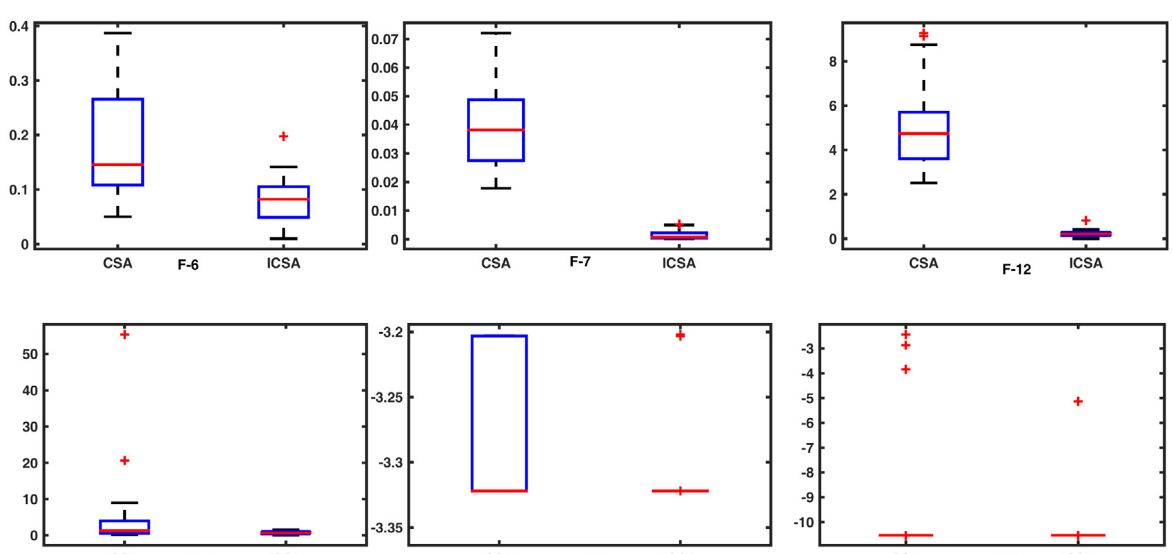

Fig. 4. Box plot analysis.

Please cite this article as: S. Shekhawat and A. Saxena, Development and applications of an intelligent crow search algorithm based on opposition based learning. ISA Transactions (2019), https://doi.org/10.1016/j.isatra.2019.09.004.

Table 5
Comparison of ICSA with other algorithms on (BFS-II) CEC-2017-II.

Function	Parameters	β-GWO [44]	GWO [36]	GOA [45]	MFO [35]	CSA [24]	WOA [34]	ICSA
CECF16	mean	2.24E+03	2.28E+03	3.02E+03	3.05E+03	2.55E+03	3.38E+03	**2.22E+03**
	SD	2.47E+02	2.60E+02	3.90E+02	3.39E+02	2.53E+02	3.60E+02	2.53E+02
	p-values	**1.59E−11**	**3.88E−01**	**3.40E−07**	**1.68E−08**	1.34E−01	**1.02E−15**	N/A
CECF17	mean	1.97E+03	1.95E+04	2.31E+03	2.39E+03	2.08E+03	2.40E+03	**1.95E+03**
	SD	1.41E+02	1.39E+02	2.31E+02	2.37E+02	1.80E+02	2.27E+02	1.54E+02
	p-values	**2.96E−03**	4.26E−01	**1.50E−08**	**2.75E−11**	6.88E−01	**8.30E−12**	N/A
CECF18	mean	4.61E+05	6.94E+05	1.06E+06	3.76E+06	2.52E+04	3.04E+06	**2.52E+04**
	SD	5.02E+05	1.22E+06	1.87E+06	6.95E+06	1.12E+04	3.34E+06	1.34E+04
	p-values	**1.20E−17**	8.62E−01	**4.99E−18**	**6.68E−18**	9.73E−01	**3.30E−18**	N/A
CECF19	mean	3.12E+05	3.72E+05	9.69E+06	3.68E+06	8.57E+03	1.65E+06	**7.19E+03**
	SD	8.22E+05	6.39E+05	1.99E+07	1.49E+07	6.78E+03	1.43E+06	6.30E+03
	p-values	**2.43E−16**	5.43E−01	**3.30E−18**	**8.26E−15**	2.26E−01	**4.18E−18**	N/A
CECF20	mean	2.31E+03	2.31E+03	2.66E+03	2.59E+03	2.41E+03	2.67E+03	**2.31E+03**
	SD	1.18E+02	1.21E+02	1.67E+02	2.08E+02	1.19E+02	2.03E+02	9.77E+01
	p-values	**4.09E−03**	8.46E−01	**9.18E−15**	**2.68E−09**	1.95E−02	**2.21E−13**	N/A
CECF21	mean	2.37E+03	2.38E+03	2.48E+03	2.47E+03	2.42E+03	2.53E+03	2.41E+03
	SD	2.22E+01	1.86E+01	4.25E+01	4.07E+01	2.96E+01	4.25E+01	4.13E+01
	p-values	**5.96E−13**	4.12E−02	1.91E−12	**2.58E−09**	3.39E−01	**3.01E−17**	N/A
CECF22	mean	4.17E+03	4.17E+03	6.37E+03	6.30E+03	2.30E+03	6.13E+03	**2.30E+03**
	SD	1.37E+03	1.31E+03	1.54E+03	1.49E+03	1.39E+00	2.16E+03	1.40E+00
	p-values	**3.30E−18**	9.79E−01	**3.30E−18**	**3.30E−18**	6.68E−01	**3.30E−18**	N/A
CECF23	mean	2.73E+03	2.74E+03	2.89E+03	2.81E+03	2.93E+03	2.97E+03	**2.62E+03**
	SD	3.07E+01	3.48E+01	8.31E+01	3.34E+01	6.61E+01	8.32E+01	6.32E+01
	p-values	**5.95E−18**	1.83E−01	**1.43E−02**	**4.37E−15**	6.93E−01	**6.26E−04**	N/A
CECF24	mean	2.91E+03	2.91E+03	3.04E+03	2.97E+03	3.04E+03	3.10E+03	3.05E+03
	SD	5.03E+01	6.06E+01	5.28E+01	3.61E+01	6.09E+01	6.38E+01	6.93E+01
	p-values	**1.63E−14**	4.86E−01	**4.26E−01**	**3.57E−09**	4.62E−01	**3.33E−05**	N/A
CECF25	mean	2.95E+03	2.95E+03	2.98E+03	3.10E+03	2.92E+03	3.02E+03	**2.91E+03**
	SD	3.38E+01	2.74E+01	1.10E+02	1.80E+02	2.36E+01	6.87E+01	4.28E+01
	p-values	**2.65E−07**	7.08E−01	**1.94E−03**	**6.52E−10**	4.86E−01	**2.48E−14**	N/A
CECF26	mean	4.37E+03	4.41E+03	5.67E+03	5.44E+03	3.78E+03	6.90E+03	**3.73E+03**
	SD	2.64E+02	3.28E+02	8.94E+02	3.85E+02	1.17E+03	8.66E+02	1.50E+03
	p-values	**2.01E−05**	9.41E−01	**4.55E−08**	**2.30E−06**	6.57E−02	**1.63E−14**	N/A
CECF27	mean	3.23E+03	3.23E+03	3.28E+03	3.24E+03	3.42E+03	3.35E+03	3.42E+03
	SD	1.15E+01	1.64E+01	4.90E+01	1.72E+01	9.55E+01	9.08E+01	8.51E+01
	p-values	**3.30E−18**	2.33E−02	**9.41E−14**	**1.34E−17**	3.52E−01	**5.01E−06**	N/A
CECF28	mean	3.33E+03	3.34E+03	3.48E+03	3.85E+03	3.22E+03	3.41E+03	**3.22E+03**
	SD	4.03E+01	5.61E+01	1.70E+02	6.23E+02	2.22E+01	8.38E+01	2.57E+01
	p-values	**5.29E−18**	6.59E−01	**4.99E−18**	**3.30E−18**	6.78E−01	**3.94E−18**	N/A
CECF29	mean	3.65E+03	3.66E+03	4.09E+03	4.05E+03	4.07E+03	4.65E+03	4.03E+03
	SD	1.36E+02	1.31E+02	2.60E+02	2.53E+02	2.16E+02	3.77E+02	2.46E+02
	p-values	**2.33E−13**	6.16E−01	**1.96E−01**	**6.35E−01**	3.22E−01	**3.83E−13**	N/A
CECF29	mean	5.22E+06	3.68E+06	9.72E+06	9.41E+05	2.09E+05	8.36E+06	**1.36E+05**
	SD	5.14E+06	2.43E+06	1.76E+07	2.49E+06	2.72E+05	5.56E+06	1.17E+05
	p-values	4.99E−18	1.74E−01	3.30E−18	1.02E−02	3.25E−01	4.70E−18	N/A

- For BFS-I, a few boxplots are shown in Fig. 4. We have included six box plots, two from each category. From these boxplots, it is evident to judge that interquartile range for ICSA is very low when it is compared with the CSA. For function F-6, F-7, F-12, F-13 and F-20 the box plots for ICSA are super narrow. For F-23 the data distribution is uneven for CSA. This analysis shows that the ICSA can produce results in a narrow range, which indicates the better solution quality.
- For BFS-II, we have plotted the box plots for CEC17 benchmark functions in Fig. 5. From this analysis we have observed that box plot for proposed ICSA are super narrow and mean values are also optimal for ICSA for most of the cases. Narrow box plots with optimal mean is indicator of high optimization performance. To read this diagram we have indicated numbers on x-axis where no.7 indicates proposed variant. The entries 1–6 reflects the results of β-GWO [44] GWO [36] GOA [45] MFO [35] CSA [24] and WOA [34] respectively.

5.3. Trajectory analysis

A good comparison between our newly developed variant ICSA and original CSA through trajectory analysis is presented in Fig. 6. To judge the performance of ICSA, we choose three unimodal functions (F-1, F-2, F-3), three multimodal functions (F-8, F-11, F-13) and three fixed dimension multimodal functions (F-15, F-16, F-23) from BFS-I. Trajectory analysis presents a comparison of position of first crow with the number of iterations. Fig. 6 clearly indicates that a combined approach of cosine function and OBL enhanced exploration factor at early stages and also ensure the steady behavior of optimal position at a later stage. In most of the exhibited cases we observe that the trajectories are steady and lower for the proposed variant. In case of F-11 it is observed that high perturbations are found in the trajectory of first crow of CSA as compared with the proposed variant, which indicates that the search process adopted in ICSA is more swift and robust. Similarly, the trajectories settle earlier for ICSA. The position of crow also provides the details of the improvement in convergence.

5.4. Iterative execution time analysis

We have drawn the execution times for each iteration for functions shown in Fig. 7. The analysis is shown for F-1 marked as (a), F-2 (b), F-3 (c), F-8 (d), F-11 (e), F-13 (f), F-15 (g), F-20 (h) and F-23 (i) of BFS-I functions. We have observed that the execution time per iteration is optimal for almost all the cases for proposed variant. However, in some cases it is equivalent to the original algorithm. This analysis is important as we have added two computational steps in the original algorithm. It is observed that with the inculcation of these two steps, execution time is not increased. On the other hand, in some cases it is decreased, which is aligned with our claim that the proposed variant is accelerated.

5.5. Convergence property analysis

Convergence curves of functions F-1, F-4, F-7 and F-13 are shown in Fig. 8. From the curves, it can be concluded that the convergence properties of the CSA are much enhanced with the proposed modifications. A clear accelerating effect can be seen in the last few iterations of function F-1 and F-13.

5.6. Scalability test

To validate the performance of proposed ICSA, we have performed the numerical experiments on unimodal and multimodal benchmark functions of 50 and 100 dimensions of BFS-I functions. From these benchmarking results, which are shown in Table 6, it can be concluded that the proposed variant ICSA performed well on almost all the functions. These results show that proposed variant is able to solve higher dimensional problems as the values of different statistical attributes are optimal for the proposed variant.

After conducting these tests, and observing positive implications of proposed modifications in CSA over standard test bench mark problems, the applicability of variant is checked on various real world engineering problem in following section.

6. Case studies

In this section, we applied our improved version of CSA on some real optimization problems. The result measures the impact of opposition based learning and acceleration factor in enhancing the exploration and exploitation properties of CSA.

6.1. Design problem I:Three truss bar

This problem is a well-known non-linear constrained problem and has been widely used for benchmarking of many problems [19,24] and [45]. Fig. 9 shows the schematic diagram of the problem. The objective of this problem is to minimize the volume (X) of a statically loaded 3-bar truss subject to stress, buckling and deflection constraints. The expression for the volume is given as:

Let $\vec{X} = [x_1, x_2]$

$$\text{Min} f\left(\vec{X}\right) = \left(2\sqrt{2}x_1 + x_2\right) l \tag{7}$$

subject to:

$$f_1\left(\vec{X}\right) = \frac{2\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2} P - \sigma \le 0 \tag{8}$$

$$f_2\left(\vec{X}\right) = \frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2} P - \sigma \le 0 \tag{9}$$

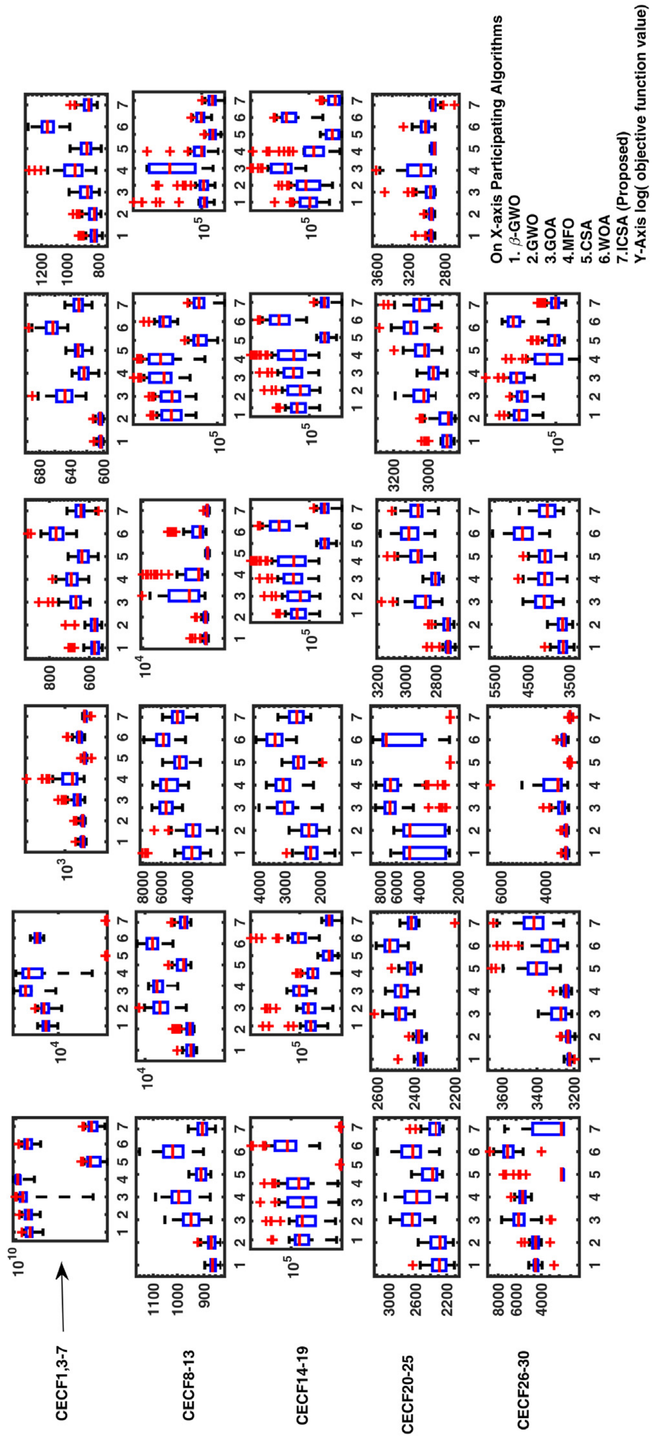

Fig. 5. Boxplot analysis on CEC-2017.

$$f_3\left(\vec{X}\right) = \frac{1}{\sqrt{2}x_2 + x_1} P - \sigma \le 0 \tag{10}$$

where $l = 100$ cm, $P = 2$ kN/cm^2, $\sigma = 2$ kN/cm^2 with variable range $0 \le x_1, x_2 \le 1$.

For solving this optimization problem, no. of search agents (30) and maximum iteration count (500) are considered and kept constant for all the variants. The results of the ICSA are aligned with some of the well-known meta heuristic approaches,

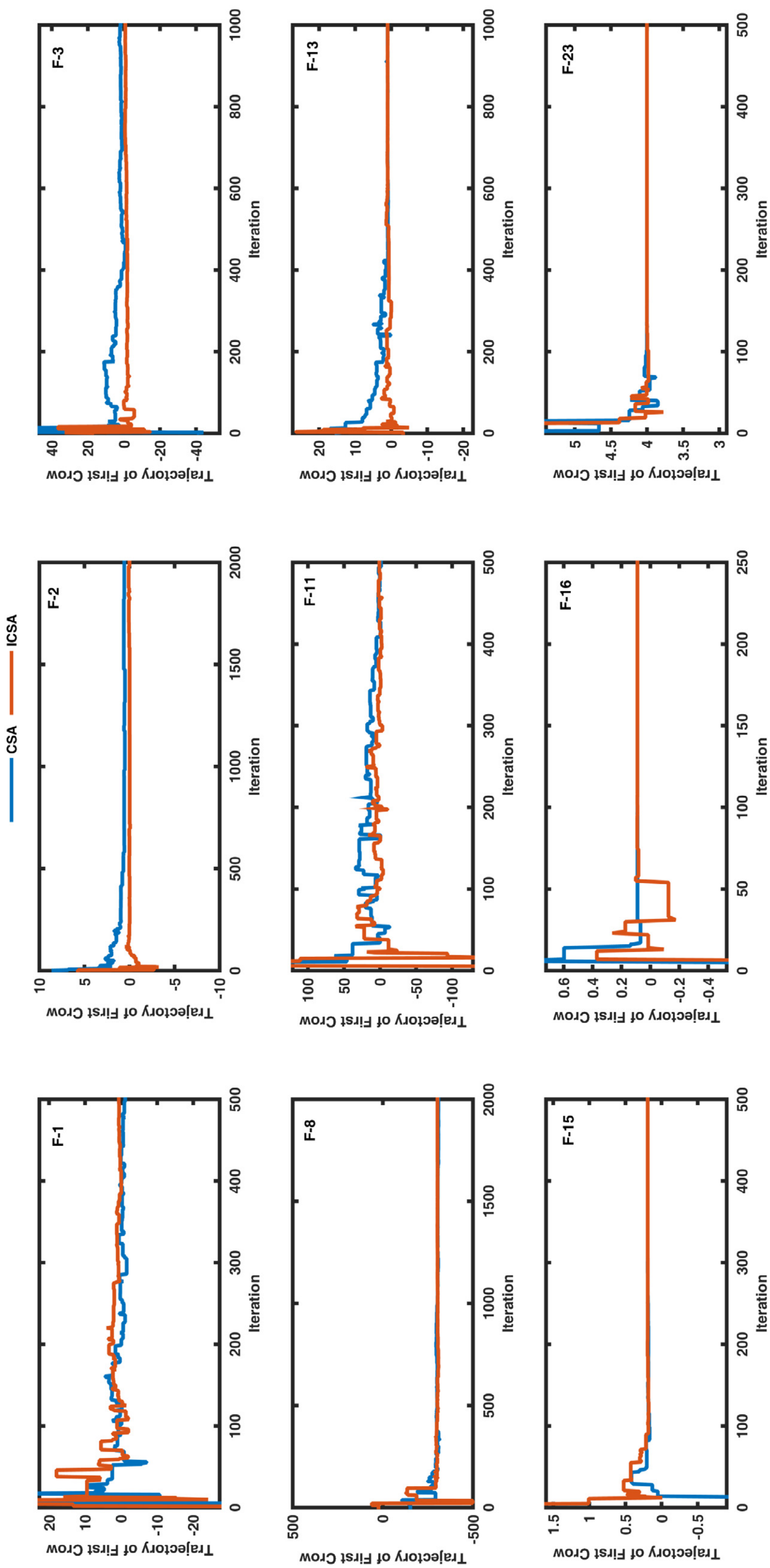

Fig. 6. Trajectory analysis.

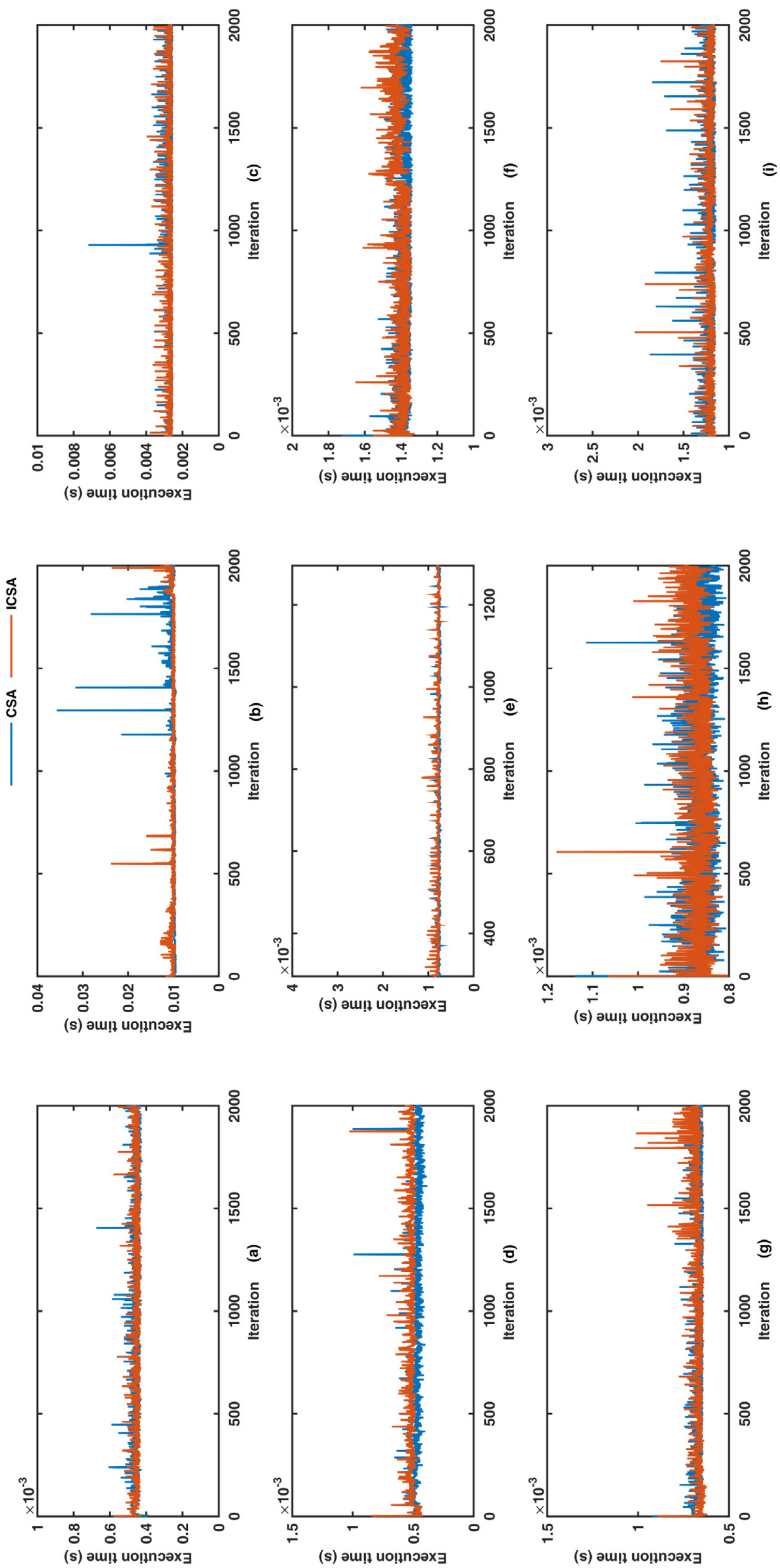

Fig. 7. Execution time analysis.

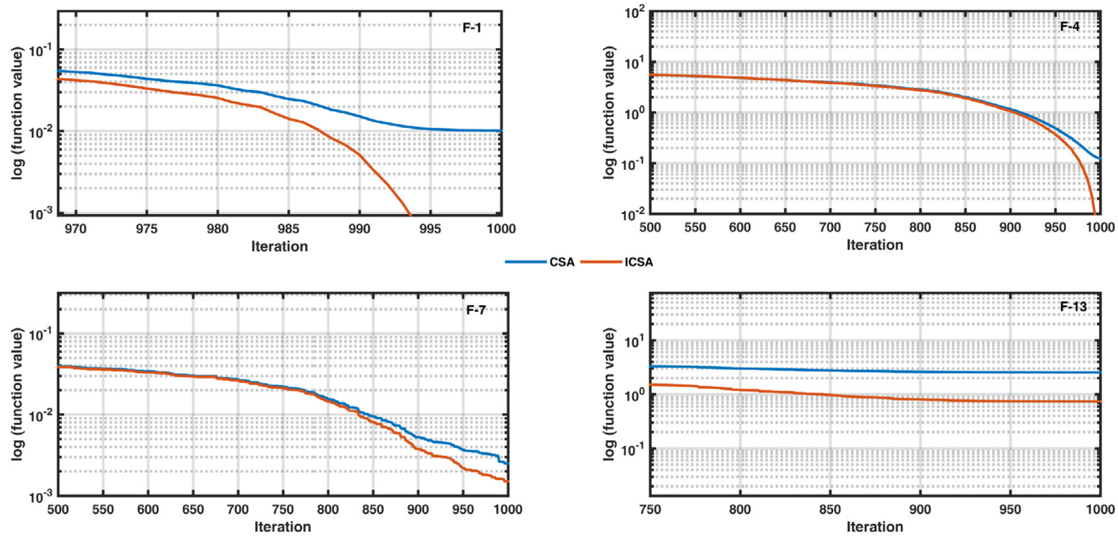

Fig. 8. Comparison of convergence characteristic of ICSA.

Table 6
Results of scalability test (BFS-I)

Function	CSA				ICSA			
	Results of 50 dim Problem							
	Min	Max	Mean	SD	Min	Max	Mean	SD
F-1	1.156E−02	1.109E−01	3.935E−02	2.056E−02	**1.541E−60**	**2.978E−60**	**2.087E−60**	**2.810E−61**
F-3	7.662E+01	2.165E+02	1.495E+02	3.673E+01	**4.506E−60**	**1.536E−07**	**5.118E−09**	**2.803E−08**
F-4	3.819E+00	8.893E+00	5.676E+00	1.072E+00	**3.501E−31**	**3.728E−31**	**3.617E−31**	**6.159E−33**
F-5	4.878E+01	2.778E+02	1.020E+02	5.289E+01	**4.642E+01**	**4.894E+01**	**4.868E+01**	**4.603E−01**
F-6	1.820E−02	6.512E−02	4.029E−02	1.399E−02	**1.314E−03**	**3.200E−02**	**1.607E−02**	**7.531E−03**
F-7	1.998E−02	8.694E−02	5.105E−02	1.542E−02	**5.391E−05**	**1.493E−03**	**6.154E−04**	**4.413E−04**
F-8	−1.242E+04	−7.445E+03	−1.115E+04	1.110E+03	**−1.315E+04**	**−9.009E+03**	**−1.096E+04**	**1.033E+03**
F-9	1.893E+01	8.264E+01	4.500E+01	1.540E+01	**0.000E+00**	**2.737E−08**	**9.123E−10**	**4.997E−09**
F-10	2.988E+00	5.947E+00	4.418E+00	7.899E−01	**8.882E−16**	**8.882E−16**	**8.882E−16**	**0.000E+00**
F-11	7.237E−02	2.300E−01	1.455E−01	3.741E−02	**0.000E+00**	**6.183E−08**	**2.061E−09**	**1.129E−08**
F-12	1.705E+00	8.432E+00	4.612E+00	1.456E+00	**3.935E−04**	**5.207E−01**	**2.076E−01**	**1.188E−01**
F-13	2.146E+00	7.624E+01	3.125E+01	2.623E+01	**1.632E+00**	**4.202E+00**	**2.655E+00**	**6.561E−01**
Function	Results of 100 dim Problem							
	Min	Max	Mean	SD	Min	Max	Mean	SD
F-1	5.163E+00	1.115E+01	7.893E+00	1.490E+00	**3.871E−60**	**1.185E−07**	**1.543E−08**	**4.001E−08**
F-3	8.173E+02	1.594E+03	1.230E+03	2.142E+02	**1.348E−59**	**4.585E−58**	**8.982E−59**	**1.032E−58**
F-4	7.702E+00	1.164E+01	9.101E+00	8.831E−01	**3.538E−31**	**3.748E−31**	**3.681E−31**	**4.860E−33**
F-5	4.165E+02	8.268E+02	6.072E+02	1.285E+02	**9.869E+01**	**9.887E+01**	**9.879E+01**	**4.835E−02**
F-6	3.709E+00	1.138E+01	7.173E+00	1.902E+00	**2.184E+00**	**4.938E+00**	**3.245E+00**	**6.002E−01**
F-7	9.890E−02	2.199E−01	1.437E−01	2.802E−02	**6.508E−05**	**2.008E−03**	**8.660E−04**	**5.212E−04**
F-8	−2.356E+04	−9.477E+03	−1.925E+04	2.872E+03	**−2.056E+04**	**−1.188E+04**	**−1.760E+04**	**2.139E+03**
F-9	4.714E+01	1.317E+02	9.427E+01	2.324E+01	**0.000E+00**	**6.363E−08**	**2.121E−09**	**1.162E−08**
F-10	3.962E+00	7.009E+00	5.060E+00	8.356E−01	**8.882E−16**	**4.526E−05**	**2.928E−06**	**1.115E−05**
F-11	1.014E+00	1.108E+00	1.066E+00	2.365E−02	**0.000E+00**	**0.000E+00**	**0.000E+00**	**0.000E+00**
F-12	3.280E+00	7.495E+00	5.039E+00	1.240E+00	**5.574E−02**	**6.643E−01**	**3.971E−01**	**1.211E−01**
F-13	3.513E+01	1.584E+02	1.067E+02	2.608E+01	**7.081E+00**	**9.986E+00**	**9.212E+00**	**8.582E−01**

namely Society and Civilization (SC) algorithm, Hybrid Particle Swarm Optimization and Differential Evolution (PSO-DE), Mine Blast Algorithm (MBA) and Differential Evolution with Dynamic Stochastic Selection(DSS-MDE). Each algorithm has been executed for 20 times and the results are exhibited in Table 7. It can be noted here that the value of SD is optimal for ICSA as compared with other opponents.

Table 7
Comparison of results on design problem I: Three truss bar.

Algorithm	Max	Mean	Min	SD
ICSA	263.8958434	263.18958434	263.8958434	**1.01E−10**
CSA	263.8958434	263.8958434	263.8958434	2.842170943
SC [48]	263.969756	263.903356	263.895846	1.30E−02
PSO-DE [49]	263.895843	263.895843	263.895843	4.50E−10
DSS-MDE [50]	263.895849	263.895843	263.895843	9.72E−07
MBA [51]	263.915983	263.897996	263.895852	3.93E−03
RL-BA [52]	263.924700	263.9003	263.89584	6.06E−03

6.2. Design problem II: Estimating parameters for Frequency Modulated (FM) sound waves

Frequency-Modulated (FM) sound wave synthesis is an important process of modern music systems. Parameter estimation of frequency modulated synthesizer is a six dimensional optimization problem. The objective of this problem is to generate sound signals which are close replica to the target sound waves. The nature of the problem is complex and multimodal. The minima of the objective function is at zero. The vector to be optimized

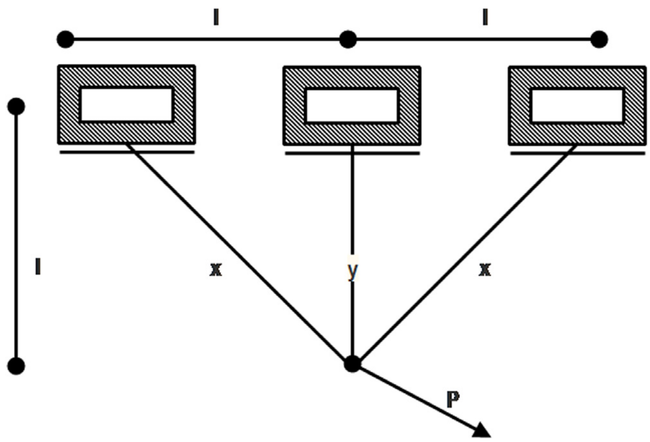

Fig. 9. Three truss bar design.

Table 8
Comparison of results on frequency modulation problem.

Algorithms	Mean	SD	Max	Min	p-values
SCA [53]	1.95E+01	4.24E+00	2.42E+01	1.06E+01	**4.23E−02**
GWO [36]	1.94E+01	2.97E+00	2.52E+01	1.16E+01	**3.97E−02**
WOA [34]	2.18E+01	4.07E+00	2.51E+01	1.15E+01	**3.94E−02**
HHO [54]	2.01E+01	4.54E+00	2.51E+01	1.17E+01	2.24E−01
CSA [24]	1.87E+01	7.01E+00	2.49E+01	1.13E−02	8.51E−01
CCSA [25]	2.47E+01	1.21E+00	2.61E+01	2.15E+01	**2.30E−06**
ImCSA [30]	2.33E+01	2.91E+00	2.64E+01	1.49E+01	**9.26E−04**
ICSA	**1.84E+01**	4.07E+00	**2.37E+01**	1.13E+01	N/A

has 6 parameters as per equation

$$\vec{Z} = (\alpha_1, \delta_1, \alpha_2, \delta_2, \alpha_3, \delta_3) \tag{11}$$

The equations for target and estimated sound waves are as follows:

$$x\,(u) = \alpha_1 \sin\left(\delta_1 u\phi + \alpha_2 \sin\left(\delta_2 u\phi + \alpha_3 \sin\left(\delta_3 u\phi\right)\right)\right) \tag{12}$$

$$x_0\,(u) = (1.0) \sin\left((5.0)\, u\phi - (1.5) \sin\left((4.8)\, u\phi + (2.0) \sin\left((4.9)\, u\phi\right)\right)\right) \tag{13}$$

$$Min\ f(\vec{Z}) = \sum_{u=0}^{100} (x(u) - x_0(u))^2 \tag{14}$$

Table 8 depicts the results of this problem. From the table, it is observed that Max and Mean values of error in objective functions are optimal for ICSA. The comparison with contemporary optimizers indicates that the proposed version of CSA handles the optimization process more efficiently as compared to other opponents. It can be seen and concluded that ICSA exhibits better results.

The last column of Table 8 shows the p-values obtained from Wilcoxon ranksum analysis. The p-values less than 0.05 indicate that there is a significant difference between the proposed ICSA performance and other evolutionary algorithms. We observed that p-values for SCA, GWO, WOA, CCSA and ImCSA are less than 0.05 hence the results of ICSA are statistically different. However, in some cases the performance of ICSA is competitive. For further investigation, we have included box plot analysis. Fig. 10 shows the box plot analysis of different contemporary optimizers. The statistics related to this plot is shown in Table 9. From the boxplots, it is evident that proposed variant shows superior characteristics. As the Interquartile range (IQR) is optimal for the case of ICSA as compared to any other optimizer.

6.3. Design problem III: Model Order Reduction (MOR)

MOR is a concept of systems and control engineering which is used here for approximation of the higher-order linear time invariant systems (LTI). MOR helps in simplifying the dynamical system without affecting the input, output behavior of system. MOR can be defined as

Let $H\,(s) : v \rightarrow z$ be transfer function of order n then MOR finds a reduced order transfer function $H\,(s) : v \rightarrow \tilde{z}$ of order $\tilde{n}$ such that for the same input $v\,(t)$ the output is $\tilde{z}\,(t) \approx z\,(t)$. Here H(s) must be rational and stable.

The latest trend is to use optimization techniques to get a reduced order model H(S) in the following way :

$$\text{minimize ISE} = \int_0^\infty \left[z\,(t) - \tilde{z}\,(t)\right]^2 dt \tag{15}$$

where ISE $< \epsilon$, ϵ is error. For validation of the proposed variant, we have chosen two higher order functions and perform the MOR.

Table 9
Statistical analysis of frequency modulation.

Parameters	SCA [53]	GWO [36]	WOA [34]	CSA [24]	HHO [54]	CCSA [25]	ImCSA [30]	ICSA
Min	1.06E+01	1.16E+01	1.15E+01	1.13E−02	1.17E+01	2.15E+01	1.49E+01	1.13E+01
Q1	1.72E+01	1.87E+01	2.14E+01	1.40E+01	1.74E+01	2.42E+01	2.31E+01	1.54E+01
Median	2.08E+01	1.90E+01	2.31E+01	2.24E+01	2.05E+01	2.50E+01	2.41E+01	**1.89E+01**
Q3	2.29E+01	2.14E+01	2.47E+01	2.36E+01	2.41E+01	2.56E+01	2.46E+01	2.17E+01
Max	2.42E+01	2.52E+01	2.51E+01	2.49E+01	2.51E+01	2.61E+01	2.64E+01	**2.37E+01**
Box1 (hidden)	1.72E+01	1.87E+01	2.14E+01	1.40E+01	1.74E+01	2.42E+01	2.31E+01	1.54E+01
Box2 (lower)	3.57E+00	3.71E−01	1.68E+00	8.35E+00	3.06E+00	8.23E−01	1.02E+00	3.52E+00
Box3 (upper)	2.08E+00	2.37E+00	1.56E+00	1.27E+00	3.67E+00	5.79E−01	4.50E−01	2.75E+00
Whisker Top	1.28E+00	3.76E+00	4.33E−01	1.24E+00	9.60E−01	5.18E−01	1.80E+00	2.08E+00
Whisker Bottom	6.63E+00	7.09E+00	9.97E+00	1.40E+01	5.67E+00	2.66E+00	8.18E+00	4.07E+00

Table 10
Comparison of results on MOR for function1.

Algorithm	ISE	A_1	A_0	B_2	B_1	B_0
ICSA	**2.32E−04**	0.229766723	2.6431063	0.603914029	3.15305627	2.643501819
CSA (studied) [24]	2.34E−04	0.213686	2.485470552	0.565550182	2.962837294	2.48583343
Birader and Saxena [55]	1.15E−03	0.2838	1.0004	0.3986	1.3744	1.0000
Sikander and Prasad [56]	2.78E−02	0.6997	0.6997	1.0000	1.4577	0.6997
Desai and Prasad [57]	2.84E−03	0.8058	0.7944	1.0000	1.6500	0.7944
Routh Hurwitz [58]	9.74E−02	0.2057	24.0000	30.0000	42.0000	24.0000

Please cite this article as: S. Shekhawat and A. Saxena, Development and applications of an intelligent crow search algorithm based on opposition based learning. ISA Transactions (2019), https://doi.org/10.1016/j.isatra.2019.09.004.

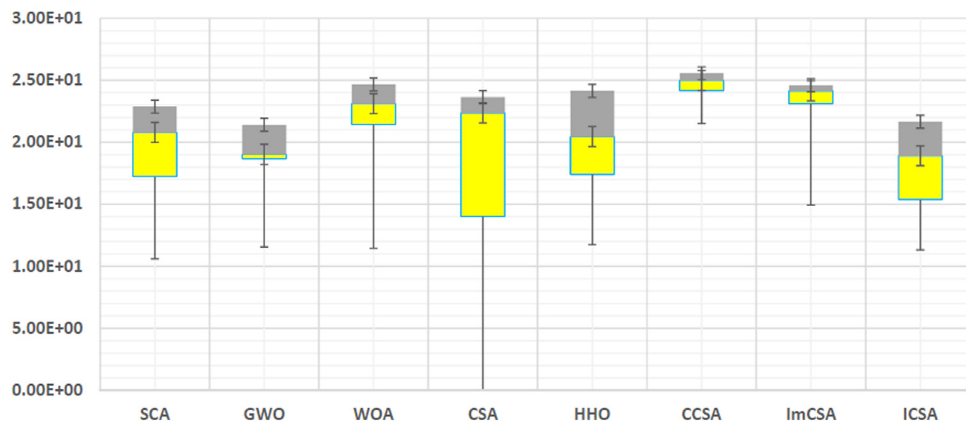

Fig. 10. Box plot analysis of FM problem.

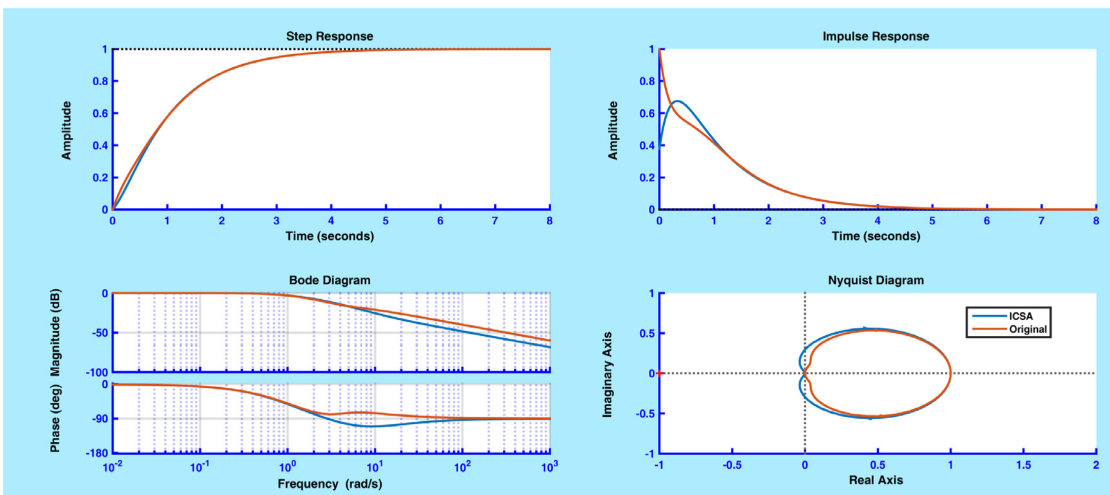

Fig. 11. Properties of proposed reduced order model-Function1.

Table 11
Comparison of results on MOR for function2.

Algorithm	ISE	A_1	A_0	B_2	B_1	B_0
ICSA	**1.1E−06**	0.0174	0.0652	1.2474	3.3696	2.4479
CSA (studied) [24]	1.6E−07	0.0505	0.0656	1.3134	3.4886	2.4686
GOA (studied) [45]	1.90E−01	0.6863	0.1682	2.3151	2.1642	2.8499
OB-ac-ALO [59]	2.61E−04	0.0050	0.0050	0.1866	0.4519	0.1871

- Function 1: For this analysis, we have chosen 4th order transfer function, which is given by:

$$T(s) = \frac{(s^3 + 7s^2 + 24s + 24)}{(s^4 + 10s^3 + 35s^2 + 50s + 24)} \tag{16}$$

In this analysis, the function represented by (16) is reduced as per following equation

$$T(s)' = \frac{A_1 s + A_0}{B_2 s^2 + B_1 s + B_0} \tag{17}$$

The objective is to find out the parameters of Eq. (17) while solving the optimization (15).

- Function 2: The second problem is chosen as

$$T(s) = \frac{(s + 4)}{(s^4 + 19s^3 + 113s^2 + 245s + 150)} \tag{18}$$

The comparison of ISE values obtained by using ICSA and other reported techniques is represented in Tables 10 and 11 for function 1 and 2. The values are more optimal for ICSA than other mentioned techniques which clearly indicate a better approximation of Model order reduction than CSA and other well-known examples from literature. Figs. 11 and 12 show the analysis of the reduced model order function with the original function, CSA and ICSA. Inspecting the diagrams, it can be easily concluded that step as well as impulse responses are aligned with the original system responses. Further, to clarify the proposed approach bode diagram and Nyquist plots of the original function and the reduced model by ICSA have also been exhibited. The efficient performance of our proposed variant also justifies that the integration of opposition based theory and cosine based acceleration factor are successful in improving the results. Following points are emerged from this study.

- The statistical results of MOR problem are depicted in Table 12. In this table optimization results of studied algorithms namely Sine Cosine Algorithm (SCA) [53], GWO [36], WOA [34], recently published Harris Hawk Optimizer (HHO) [54], CCSA [25], ImCSA [30] along with CSA [24] are presented for both functions. the results are presented in terms of min, max, mean and SD of the error in objective function.
- The p-values of rank sum test are also shown with this analysis in the last column of this table. From the results reported in Table 12 we observe that all statistical attributes for ICSA are optimal for both functions. These values are shown in bold face, along with that, Wilcoxon Ranksum test

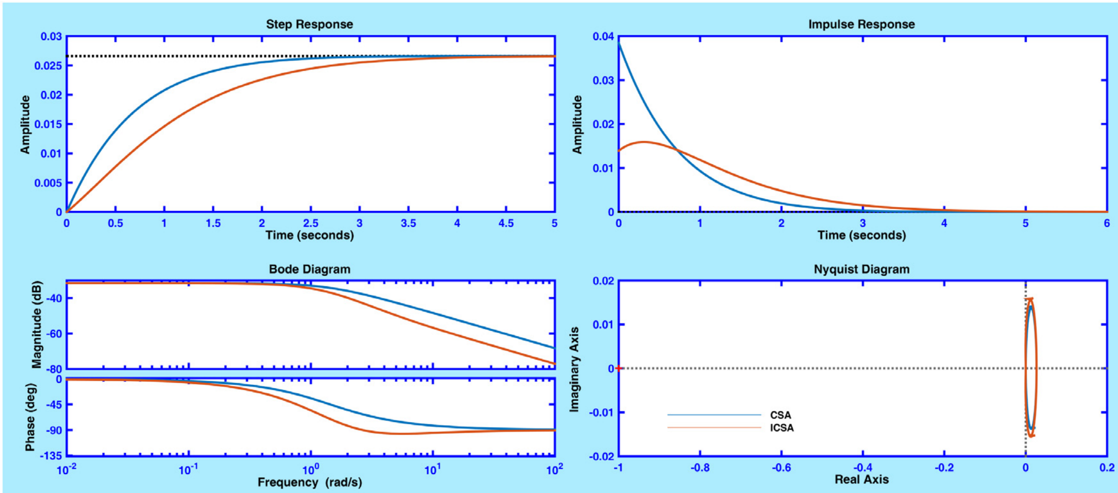

Fig. 12. Properties of proposed reduced order model-Function2.

also give positive results as obtained p-values are less than 0.05.

- From these application results, it is evident that performance of ICSA is superior to other contemporary optimizers for MOR problems.
- Further, the performance of ICSA is tested through convergence property analysis. Convergence curves for MOR optimization processes are shown in Figs. 13 and 14 for function 1 and 2 respectively. From these figures it is evident that optimization process is effectively handled by ICSA. Positive implications of modifications in CSA are quite evident in these applications.
- Further, to test statistical significance, boxplots for these two problems of MOR are plotted and depicted in Figs. 15 and 16 for function 1 and 2 respectively. By observing these plots, we can analyze that the mean values are optimal as compared with other opponents. In case of function1, box plot is super narrow and median values are very less. In case of function 2 median values are optimal and there is a good data agreement between the solution obtained.

From these results, it is evident that proposed ICSA shows better performance as compared with native CSA and some contemporary optimizers. Following section presents conclusion of this research work.

7. Conclusion

In the present work, we have proposed two major modifications in CSA. The first one is application of OBL, which is applied in initialization phase of the CSA and the second one is a cosine function driven acceleration factor for improvement in position update mechanism and further for enhancing the exploitation virtues of CSA. The implications of these modifications on the existing CSA are proven positive when it has been tested over two sets of benchmark functions. First is the set of 23 conventional benchmark function and second is latest benchmark function set CEC-2017, which both include functions of different kind and nature. Further, the applicability of the proposed variant is proven on three engineering applications. Following are the major conclusions of this study:

- The performance of the proposed variant is analyzed with the help of several statistical tests including Wilcoxon rank sum and box plot analysis and other performance measures namely trajectory analysis and iteration execution

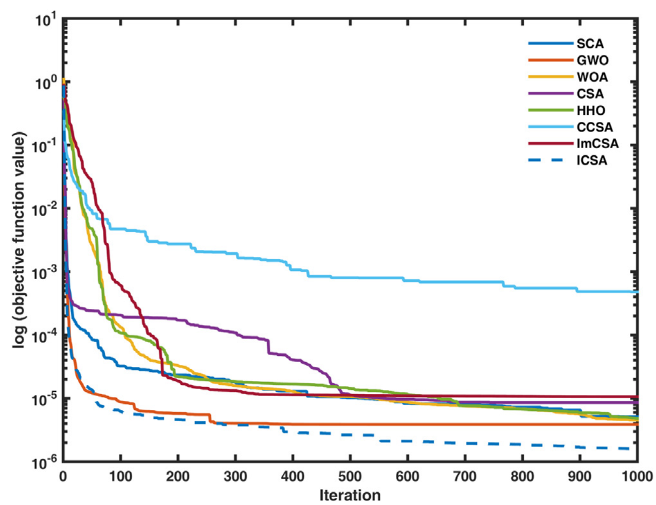

Fig. 13. Convergence properties analysis of proposed reduced order model-Function1.

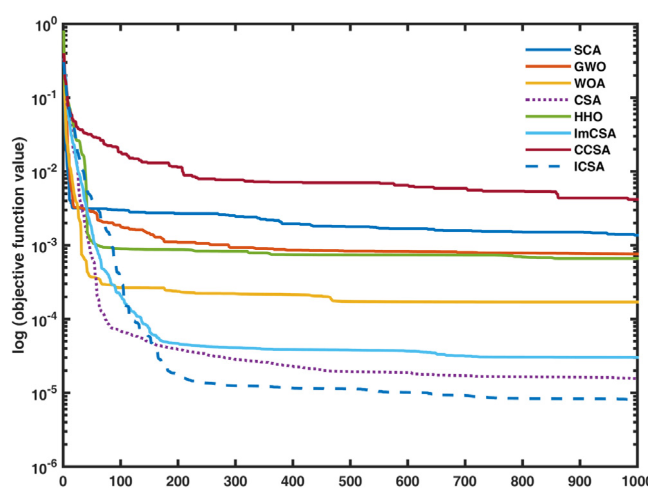

Fig. 14. Convergence properties analysis of proposed reduced order model-Function2.

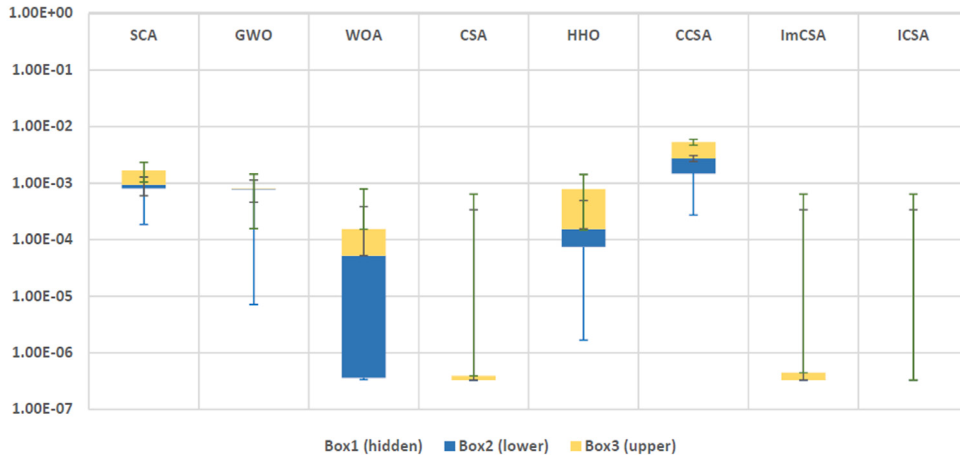

Fig. 15. Box plot analysis of MOR problem Function1.

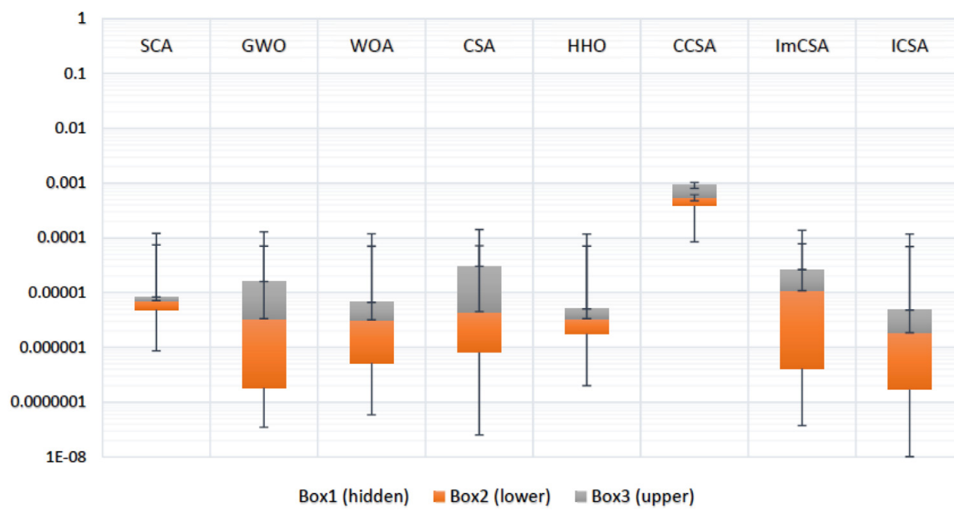

Fig. 16. Box plot analysis of MOR problem Function2.

Table 12
Comparative analysis of the performance of ICSA with modern optimizers on MOR problem.

Function1					
Algorithms	Mean	SD	Max	Min	p-values
SCA [53]	1.38E−03	9.81E−04	3.22E−03	1.85E−04	**5.00E−06**
GWO [36]	7.16E−04	2.47E−04	9.35E−04	7.14E−06	**6.98E−11**
WOA [34]	1.35E−04	2.28E−04	7.68E−04	3.34E−07	**1.63E−02**
CSA [24]	3.37E−06	1.23E−05	5.56E−05	3.28E−07	2.90E−01
HHO [54]	8.22E−04	1.94E−03	8.61E−03	1.66E−06	**4.29E−02**
CCSA [25]	3.46E−03	2.92E−03	1.25E−02	2.73E−04	**4.23E−05**
ImCSA [30]	2.74E−06	5.65E−06	2.09E−05	3.28E−07	**4.58E−02**
ICSA	**3.63E−07**	**1.57E−07**	**1.03E−06**	**3.28E−07**	N/A
Function2					
SCA [53]	6.63E−06	3.00E−06	1.28E−05	8.75E−07	**9.09E−04**
GWO [36]	1.26E−05	2.22E−05	9.24E−05	3.48E−08	**5.48E−02**
WOA [34]	3.15E−06	2.71E−06	6.72E−06	5.88E−08	4.12E−01
CSA [24]	1.87E−05	2.48E−05	8.82E−05	2.54E−08	**6.36E−03**
HHO [54]	9.01E−06	1.32E−05	3.97E−05	2.00E−07	**4.94E−02**
CCSA [25]	6.50E−04	4.16E−04	1.79E−03	8.49E−05	**1.23E−06**
ImCSA [30]	1.60E−05	1.91E−05	5.50E−05	3.73E−08	**5.97E−03**
ICSA	**2.44E−06**	**2.48E−06**	**6.37E−06**	**1.02E−08**	N/A

time analysis and scalability test. It is observed that the proposed variant shows satisfactory performance when it has been compared with contemporary optimizers. Further, the convergence properties of proposed variant have also been enhanced with the inculcation of these two modifications.

- The proposed variant has also been implemented on three real world applications namely Three truss bar design problem, Frequency Modulated Sound wave parameter estimation problem and Model Order reduction of higher order transfer functions. The results of these applications are aligning with CSA and in some cases the variant provides optimal results. The results of proposed ICSA has been compared with recent optimizers and it has been observed that ICSA provides optimal results. Also, it is statistically different from other optimizers.
- The performance of the proposed variant has been compared with some of the contemporary, proven meta heuristics on mathematical functions and real applications. We observe that the proposed variant exhibits satisfactory performance. Further, comparison of the performance of proposed variant with some well-known proven meta heuristics over real engineering problems also establishes the efficacy of the proposed variant.

Applicability of proposed ICSA can be tested over more challenging problems like protein structure minimization, controller design for power system, feature selection, knapsack and filter design problem.

Table 13
Details of BFS-I.

Function	Dim	Range	Minima
Unimodal benchmark function			
$F-1(z)=\sum_{i=1}^{n} z_i^2$	30	[−100,100]	0
$F-2(z)=\sum_{i=1}^{n}\lvert z_i\rvert+\prod_{i=1}^{n}\lvert z_i\rvert$	30	[−10,10]	0
$F-3(z)=\sum_{i=1}^{n}\left(\sum_{j-1}^{i} z_j\right)^2$	30	[−100,100]	0
$F-4(z)=\max_i\{\lvert z_i\rvert \quad 1\le i\le n$	30	[−100,100]	0
$F-5(z)=\sum_{i=1}^{n-1}[100(z_{i+1}-z_i^2)^2+(z_i-1)^2]$	30	[−30,30]	0
$F-6(z)=\sum_{i=1}^{n-1}([z_i+0.5])^2$	30	[−100,100]	0
$F-7(z)=\sum_{i=1}^{n-1} iz_i^4+random[0,1]$	30	[−1.28,1.28]	0
Multimodal benchmark function			
$F-8(z)=\sum_{i=1}^{n}-z_i\sin\left(\sqrt{\lvert z_i\rvert}\right)$	30	[−500,500]	−418.9829 ×D
$F-9(z)=\sum_{i=1}^{n}\left[{z_i}^2-10\ \cos(2\pi z_i)+10\right]$	30	[−5.12,5.12]	0
$F-10(z)=-20\ \exp\left(-0.2\sqrt{\frac{1}{n}\sum_{i=1}^{n}{z_i}^2}\right)-\exp\left(\frac{1}{n}\sum_{i=1}^{n}\cos(2\pi z_i)\right)+20+e$	30	[−32,32]	0
$F-11(z)=\frac{1}{4000}\sum_{i=1}^{n}{z_i}^2-\prod_{i=1}^{n}\cos\left(\frac{z_i}{\sqrt{i}}\right)+1$	30	[−600,600]	0
$F-12(z)=\frac{\pi}{n}\left\{10\sin(\pi y_1)+\sum_{i=1}^{n-1}(y_i-1)^2[1+10\sin^2(\pi y_{i+1})]+(y_n-1)^2\right\}+\sum_{i=1}^{n}u(z_i,10,100,4)$ $y_i=1+\frac{z_{i+1}}{4}$ $u(z_i,a,k,m)=\begin{cases}k(z_i-a)^m & z_i>a\\ 0 & -a<z_i<a\\ k(-z_i-a)^m & z_i<-a\end{cases}$	30	[−50,50]	0
$F-13(z)=0.1\{sin^2(3\pi z_1)+\sum_{i=1}^{n}(z_i-1)^2[1+\sin^2(3\pi z_i+1)]+(z_n-1)^2[1+\sin^2(2\pi z_n)]\}+\sum_{i=1}^{n}u(z_i,5,100,4)$	30	[−50,50]	0
Fixed-dimension multimodal benchmark function			
$F-14(z)=\left(\frac{1}{500}+\sum_{j=1}^{25}\frac{1}{j+\sum_{i=1}^{2}(z_i-a_{ij})^6}\right)^{-1}$	2	[−65,65]	1
$F-15(z)=\sum_{j=1}^{11}\left[a_i-\frac{z_1\left(b_i^2+b_iz_2\right)}{b_i^2+b_iz_3+z_4}\right]^2$	4	[−5,5]	0.00030
$F-16(z)=4z_1^2-2.1z_1^4+\frac{1}{3}z_1^6+z_1z_2-4z_2^2+4z_2^4$	2	[−5,5]	−1.0316
$F-17(z)=\left(z_2-\frac{5.1}{4\pi^2}z_1^2+\frac{5}{\pi}z_1-6\right)^2+10\left(1-\frac{1}{8\pi}\right)\cos z_1+10$	2	[−5,5]	0.398
$F-18(z)=A(z)\times B(z)$ $A(z)=1+(z_1+z_2+1)^2(19-14z_1+3z_1^2-14z_2+6z_1z_2+3z_2^2)$ $B(z)=30+(2z_1-3z_2)^2\times(18-32z_1+12z_1^2+48z_2-36z_1z_2+27z_2^2)$	2	[−2,2]	3
$F-19(z)=-\sum_{i=1}^{4}c_i\exp\left(-\sum_{j=1}^{3}a_{ij}(z_j-p_{ij})^2\right)$	3	[1,3]	−3.86
$F-20(z)=-\sum_{i=1}^{4}c_i\exp\left(-\sum_{j=1}^{6}a_{ij}(z_j-p_{ij})^2\right)$	6	[0,1]	−3.32
$F-21(z)=-\sum_{i=1}^{5}\left[(Z-a_i)(Z-a_i)^T+c_i\right]^{-1}$	4	[0,10]	−10.1532
$F-22(z)=-\sum_{i=1}^{7}\left[(Z-a_i)(Z-a_i)^T+c_i\right]^{-1}$	4	[0,10]	−10.4028
$F-23(z)=-\sum_{i=1}^{10}\left[(Z-a_i)(Z-a_i)^T+c_i\right]^{-1}$	4	[0,10]	−10.5363

Please cite this article as: S. Shekhawat and A. Saxena, Development and applications of an intelligent crow search algorithm based on opposition based learning. ISA Transactions (2019), https://doi.org/10.1016/j.isatra.2019.09.004.

Table 14
Details of CEC-2017.

Function name	Optima
Unimodal functions	
Shifted and Rotated Bent Cigar function (CECF1)	100
Shifted and Rotated Zakharov function (CECF3)	300
Simple multimodal functions	
Shifted and Rotated Rosenbrock's function (CECF4)	400
Shifted and Rotated Rastrigin's function (CECF5)	500
Shifted and Rotated Expanded Scaffer's function (CECF6)	600
Shifted and Rotated Lunacek Bi Rastrigin function (CECF7)	700
Shifted and Rotated Non-continuous Rastrigin function (CECF8)	800
Shifted and Rotated Levy function (CECF9)	900
Shifted and Rotated Schwefel's function (CECF10)	1000
Hybrid functions	
Hybrid function 1 (N = 3) (CECF11)	1100
Hybrid function 2 (N = 3) (CECF12)	1200
Hybrid function 3 (N = 3) (CECF13)	1300
Hybrid function 4 (N = 4) (CECF14)	1400
Hybrid function 5 (N = 4) (CECF15)	1500
Hybrid function 6 (N = 4) (CECF16)	1600
Hybrid function 7 (N = 5) (CECF17)	1700
Hybrid function 8 (N = 5) (CECF18)	1800
Hybrid function 9 (N = 5) (CECF19)	1900
Hybrid function 10 (N = 6) (CECF20)	2000
Composite functions	
Composition function 1 (N = 3) (CECF21)	2100
Composition function 2 (N = 3) (CECF22)	2200
Composition function 3 (N = 4) (CECF23)	2300
Composition function 4 (N = 4) (CECF24)	2400
Composition function 5 (N = 5) (CECF25)	2500
Composition function 6 (N = 5) (CECF26)	2600
Composition function 7 (N = 6) (CECF27)	2700
Composition function 8 (N = 6) (CECF28)	2800
Composition function 9 (N = 3) (CECF29)	2900
Composition function 10 (N = 3) (CECF30)	3000

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Appendix

See Tables 13 and 14.

References

[1] Holland JH. Genetic algorithms. Sci Am 1992;267(1):66–73.
[2] Koza JR. Genetic programming II, automatic discovery of reusable subprograms. Cambridge, MA: MIT Press; 1992.
[3] Simon D. Biogeography-based optimization. IEEE Trans Evol Comput 2008;12(6):702–13.
[4] Dasgupta D, Michalewicz Z. Evolutionary algorithms in engineering applications. Springer Science & Business Media; 2013.
[5] Erol OK, Eksin I. A new optimization method: big bang–big crunch. Adv Eng Softw 2006;37(2):106–11.
[6] Hatamlou A. Black hole: A new heuristic optimization approach for data clustering. Inform Sci 2013;222:175–84.
[7] Kaveh A, Khayatazad M. A new meta-heuristic method: ray optimization. Comput Struct 2012;112:283–94.
[8] Du H, Wu X, Zhuang J. Small-world optimization algorithm for function optimization. In: International conference on natural computation. Springer; 2006, p. 264–73.
[9] Moghaddam FF, Moghaddam RF, Cheriet M. Curved space optimization: A random search based on general relativity theory, arXiv preprint arXiv:1208.2214, 2012.
[10] Formato R. Central force optimization: a new metaheuristic with applications in applied electromagnetics. prog electromagn res (pier), Vol. 77. 2007, p. 425–91.
[11] Rashedi E, Nezamabadi-Pour H, Saryazdi S. Gsa: a gravitational search algorithm. Inform Sci 2009;179(13):2232–48.
[12] Kennedy J. Particle swarm optimization. In: Encyclopedia of machine learning. Springer; 2011, p. 760–6.
[13] Dorigo M, Stützle T. The ant colony optimization metaheuristic: Algorithms, applications, and advances. In: Handbook of metaheuristics. 2003, p. 250–85.
[14] Basturk B. An artificial bee colony (abc) algorithm for numeric function optimization. In: IEEE swarm intelligence symposium. Indianapolis, in, USA, 2006; 2006.
[15] Yang X-S. A new metaheuristic bat-inspired algorithm. In: Nature inspired cooperative strategies for optimization (NICSO 2010). Springer; 2010, p. 65–74.
[16] Gandomi AH, Alavi AH. Krill herd: a new bio-inspired optimization algorithm. Commun Nonlinear Sci Numer Simul 2012;17(12):4831–45.
[17] Yang X-S. Firefly algorithm, stochastic test functions and design optimisation. Int J Bio-Inspired Comput 2010;2(2):78–84.
[18] Saxena A, Shekhawat S. Ambient air quality classification by grey wolf optimizer based support vector machine. J Environ Publ Health 2017;2017.
[19] Saxena A, Shekhawat S, Kumar R. Application and development of enhanced chaotic grasshopper optimization algorithms. Model Simul Eng 2018;2018.
[20] Mahdavi M, Fesanghary M, Damangir E. An improved harmony search algorithm for solving optimization problems. Appl Math Comput 2007;188(2):1567–79.
[21] He S, Wu QH, Saunders J. Group search optimizer: an optimization algorithm inspired by animal searching behavior. IEEE Trans Evol Comput 2009;13(5):973–90.
[22] Eita M, Fahmy M. Group counseling optimization. Appl Soft Comput 2014;22:585–604.
[23] Wolpert DH, Macready WG, et al. No free lunch theorems for search, Tech. rep., Technical Report SFI-TR-95-02-010, Santa Fe Institute (1995).
[24] Askarzadeh A. A novel metaheuristic method for solving constrained engineering optimization problems: crow search algorithm. Comput Struct 2016;169:1–12.
[25] Sayed GI, Hassanien AE, Azar AT. Feature selection via a novel chaotic crow search algorithm. Neural Comput Appl 2017;1–18.
[26] Mohammadi F, Abdi H. A modified crow search algorithm (mcsa) for solving economic load dispatch problem. Appl Soft Comput 2018;71:51–65.
[27] Gupta D, Rodrigues JJ, Sundaram S, Khanna A, Korotaev V, de Albuquerque VHC. Usability feature extraction using modified crow search algorithm: a novel approach. Neural Comput Appl 2018;1–11.
[28] Gupta D, Sundaram S, Khanna A, Hassanien AE, De Albuquerque VHC. Improved diagnosis of parkinson's disease using optimized crow search algorithm. Comput Electr Eng 2018;68:412–24.
[29] Abdelaziz AY, Fathy A. A novel approach based on crow search algorithm for optimal selection of conductor size in radial distribution networks. Eng Sci Technol Int J 2017;20(2):391–402.
[30] Díaz P, Pérez-Cisneros M, Cuevas E, Avalos O, Gálvez J, Hinojosa S, Zaldivar D. An improved crow search algorithm applied to energy problems. Energies 2018;11(3):571.
[31] Rincon P. Science/nature– crows and jays top bird iq scale, BBC News.
[32] Prior H, Schwarz A, Güntürkün O. Mirror-induced behavior in the magpie (pica pica): evidence of self-recognition. PLoS Biol 2008;6(8). e202.
[33] Tizhoosh HR. Opposition-based learning: a new scheme for machine intelligence. In: Computational intelligence for modelling, control and automation, 2005 and international conference on intelligent agents, web technologies and internet commerce, international conference on, Vol. 1. IEEE; 2005, p. 695–701.
[34] Mirjalili S, Lewis A. The whale optimization algorithm. Adv Eng Softw 2016;95:51–67.
[35] Mirjalili S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. Knowl-Based Syst 2015;89:228–49.
[36] Mirjalili S, Mirjalili SM, Lewis A. Grey wolf optimizer. Adv Eng Softw 2014;69:46–61.
[37] Saxena A, Soni BP, Kumar R, Gupta V. Intelligent grey wolf optimizer–development and application for strategic bidding in uniform price spot energy market. Appl Soft Comput 2018;69:1–13.
[38] Xu Q, Wang L, Wang N, Hei X, Zhao L. A review of opposition-based learning from 2005 to 2012. Eng Appl Artif Intell 2014;29:1–12.
[39] Wang G-G, Deb S, Gandomi AH, Alavi AH. Opposition-based krill herd algorithm with cauchy mutation and position clamping. Neurocomputing 2016;177:147–57.
[40] Shan X, Liu K, Sun P-L. Modified bat algorithm based on lévy flight and opposition based learning. Sci Program 2016;2016.
[41] Sarkhel R, Chowdhury TM, Das M, Das N, Nasipuri M. A novel harmony search algorithm embedded with metaheuristic opposition based learning. J Intell Fuzzy Systems 2017;32(4):3189–99.
[42] Alamri HS, Alsariera YA, Zamli KZ. Opposition-based whale optimization algorithm. Adv Sci Lett 2018;24(10):7461–4.
[43] Jadoun VK, Gupta N, Niazi K, Swarnkar A. Modulated particle swarm optimization for economic emission dispatch. Int J Electr Power Energy Syst 2015;73:80–8.

[44] Saxena A, Kumar R, Das S. β-Chaotic map enabled grey wolf optimizer. Appl Soft Comput 2019;75:84–105.

[45] Saremi S, Mirjalili S, Lewis A. Grasshopper optimisation algorithm: Theory and application. Adv Eng Softw 2017;105:30–47.

[46] Awad N, Ali M, Liang J, Qu B, Suganthan P. Problem definitions and evaluation criteria for the cec 2017 special session and competition on single objective bound constrained real-parameter numerical optimization. In: Technical Report. Nanyang Technological University Singapore; 2016.

[47] Wilcoxon F. Individual comparisons by ranking methods. Biom Bull 1945;1(6):80–3.

[48] Ray T, Liew KM. Society and civilization: An optimization algorithm based on the simulation of social behavior. IEEE Trans Evol Comput 2003;7(4):386–96.

[49] Liu H, Cai Z, Wang Y. Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization. Appl Soft Comput 2010;10(2):629–40.

[50] Zhang M, Luo W, Wang X. Differential evolution with dynamic stochastic selection for constrained optimization. Inform Sci 2008;178(15):3043–74.

[51] Sadollah A, Bahreininejad A, Eskandar H, Hamdi M. Mine blast algorithm: A new population based algorithm for solving constrained engineering optimization problems. Appl Soft Comput 2013;13(5):2592–612.

[52] Meng X-B, Li H-X, Gao X-Z. An adaptive reinforcement learning-based bat algorithm for structural design problems.

[53] Mirjalili S. Sca: a sine cosine algorithm for solving optimization problems. Knowl-Based Syst 2016;96:120–33.

[54] Heidari AA, Mirjalili S, Faris H, Aljarah I, Mafarja M, Chen H. Harris hawks optimization: Algorithm and applications. Future Gener Comput Syst 2019;97:849–72.

[55] Biradar S, Hote YV, Saxena S. Reduced-order modeling of linear time invariant systems using big bang big crunch optimization and time moment matching method. Appl Math Model 2016;40(15–16):7225–44.

[56] Sikander A, Prasad R. Linear time-invariant system reduction using a mixed methods approach. Appl Math Model 2015;39(16):4848–58.

[57] Desai S, Prasad R. A novel order diminution of lti systems using big bang big crunch optimization and routh approximation. Appl Math Model 2013;37(16–17):8016–28.

[58] Krishnamurthy V, Seshadri V. Model reduction using the routh stability criterion. IEEE Trans Automat Control 1978;23(4):729–31.

[59] Dinkar SK, Deep K. Accelerated opposition-based antlion optimizer with application to order reduction of linear time-invariant systems. Arab J Sci Eng 2018;1–29.