



A modification of I-SOS: performance analysis to large scale functions

Sukanta Nama^{1,2}

Accepted: 23 September 2020 / Published online: 19 March 2021
© Springer Science+Business Media, LLC, part of Springer Nature 2021

Abstract

SOS is a global optimization algorithm, based on nature, and is utilized to execute the various complex hard optimization problems. Be that as it may, some basic highlights of SOS, for example, pitfall among neighborhood optima and weaker convergence zone should be upgraded to discover better answers for progressively intricate, nonlinear, many optimum solution type problems. To diminish these deficiencies, as of late, numerous analysts increase the exhibition of the SOS by designing up a few changed form of the SOS. This paper suggests an improved form of the SOS to build up an increasingly steady balance between discovery and activity cores. This technique uses three unique procedures called adjusted benefit factor, altered parasitism stage, and random weighted number-based search. The technique is referred to as mSOS and tested in a popular series of twenty classic benchmarks. The dimension of these problems is considered to be hundred to monitor the impact of the suggested technique on the versatility of the test problems. Also, some real-life optimization problems are solved with the help of the proposed mSOS. The results investigated based on three different way and these are statistical measures, convergence, and statistical analyses. The comparison of results of the mSOS with the standard SOS, SOS variants, and certain other cutting-edge algorithms shows its improved search performance.

Keywords SOS · Continuous unconstrained functions · Adaptive benefit factor · Modified method

1 Introduction

No unaccompanied optimization techniques are accessible to effectively solve every form of optimization problem [1, 2]. In this way, within the optimization community, certain concerning the most prominent challenges is after locate the appropriate optimization methods for fixing non-linear complicated optimization problems. With many conventional deterministic approaches, optimization has begun. If there is a discontinuity in the objective function problems, a gradient primarily-based algorithm will not perform well. In its case, gradient uninterrupted nature-based optimization algorithms are the superior alternative to resolve its kinds concerning optimization problems. In this case, gradient-free nature-based optimization algorithms are the leading elective to solve these types of optimization problems.

Moreover, a few prevalent gradient-free nature-based optimization algorithms found within the literature are GA [3]; PSO [4], given the rule of scavenging behavior of the swarm of winged

creatures; DE [5], in view of the Darwinian hypothesis of advancement; BBO [6], based on the scientific models of biogeography; HS [7], in view of the melodic procedure of looking for an ideal condition of amicability; GSA [8], in view of the law of gravity and mass intuitive; WCA [9], in view of the perception of water cycle procedure and how waterways and streams stream to the ocean in reality; BSA [10], given the three essential and well-known administrators of EA that are selection, mutation and crossover; SOS [11], based on the interaction relationship among the creature in the biological system, etc. By these algorithms, a huge measure of certifiable applications can be found in the writing over practically all parts of Humanities, Science, and Technology. Some different types of optimization problems on networks found in the literature can be seen in references [44–46]. The critical focal points in these algorithms are the framework for information-exchange instrument and collaborative quality that offer assistance to locate see the look space more competently and dodge the circumstance of skipping right solutions and getting to be stuck in nearby optima.

Cheng and Prayogo proposed the Symbiotic Organisms Search (SOS) [11] and it is a new robust and efficient metaheuristic algorithm for numerical optimization and engineering problems. SOS mimics the symbiotic interaction techniques that species have developed to outlive and engender within the environment [11].

✉ Sukanta Nama
sukanta1122@gmail.com

¹ Department of Mathematics, Ram Thakur College,
Agartala, Tripura 799003, India

² Department of Applied Mathematics, MBB University,
Agartala, Tripura 799003, India

As of late, SOS has been applied to tackle different real-world applications. Although, the utilization of SOS illustrates its adequate capacity in terms of investigation and absorption, but still in a few cases, it endures the issue of stagnation at nearby optima and improper adjust between investigation and absorption. In this manner, within the literature, a few endeavors have been performed to move forward the look component of SOS. Some are:

Umam and Santosa [12] proposed SOS-VNS strategy and in that paper talks about how to hybridize the SOS with variable neighborhood search so be able to keep utilized in imitation of solving the ATSP problem. The SOS-VNS is proposed dependent on the blend of SOS and VNS to move forward the convergence and computing time of SOS.

Ref. [13] proposes new chaos and a symbiotic search algorithm (A-CSOS) focused on global competitive rankings. The target of A-CSOS is to decrease the snare into nearby optima and improve the convergence of SOS to discover better answers for progressively unpredictable, nonlinear, and multi-modular optimization problems, for example, ORPD [13].

Ref. [14] proposes an improved variant of SOS (ISOS) dependent on the hypothesis of quasi oppositional based learning and a productive option for the parasitism phase. To direct the calculation perform a comprehensive look around the finest arrangement in endeavoring to assist move forward the look demonstrate of ISOS, a chaotic neighborhood look based on the piecewise straight chaotic outline is coupled into the calculation [14]. The objective of this strategy is to maintain a strategic distance from the over investigation issue of a unique parasitism phase that causes undesirable long time look within the second rate look space as the arrangement is as of now refined [14].

To estimate parameters of smooth and non-smooth fuel cost functions for improving the solution accuracy of economic dispatch problems, the improved symbiotic organisms search (R-SOS) Algorithm is proposed [15].

Using the SOS algorithm's multi-group coordination technique and quantum behavior, a new improved variant of the SOS algorithm is suggested, called the MQSOS algorithm [16]. MQSOS has speed and convergence capability and plays a strong role in multi-population on functional problems [16].

An improved variant of the metaheuristic optimization algorithm called Opposition-based Symbiotic Organisms Search (OSOS) is proposed to tackle the question of color image segmentation [17]. OSOS is used to overcome the multi-level image thresholding technique for the segmentation of color images. Opposition-based learning principles are used to improve the execution of standard SOS [17].

An Enhanced Symbiotic Species Search (ESOS) algorithm is suggested based on the Local Search Improvement Strategies to solve unrelated parallel machine manufacturing scheduling problem with setup times [18].

A new, complex-valued encoding symbiotic organism search (CSOS) algorithm is proposed by introducing the concept of high diploid coding [19].

A Modified SOS (MSOS) algorithm [20] is proposed to improve its search (exploitation) accuracy effectiveness along with exploration by adding an adaptive benefit factor and modified parasitism vector.

However, only a few efforts have been made to deal with the time-cost trade-off problem (TCTP) in the large-scale construction projects, and the existing optimization methods are slightly limited by the trouble of parameter tuning.

As time-cost trade-off problem (TCTP) [21] is known to be an NP-hard problem, a new variant of Symbiotic Organisms Search (SOS) algorithm is proposed that does not contain control parameters, called DSOS (Discrete Symbiotic Organisms Search) which generates the parasite organism using a heuristic rule based on the network levels.

Through combining the techniques Quasi-Opposition-Based Learning (QOBL) and Chaotic Local Search (CLS) with SOS, QOCSOS has been proposed for a higher quality solution and quicker convergence [22].

Nama et al. [23] proposed an Improved Symbiotic Organisms Search (ISOS) algorithm to boost the original algorithm's efficiency and add a random weighted reflection vector to enhance the SOS algorithm's search capability.

Nama et al. [24] incorporated Simple Quadratic Interpolation (SQI) into SOS to balance SQI exploration capability and SOS extraction capacity as well as enhance algorithm robustness.

In [25], HSOS [24] is used to find the seismic bearing potential of a shallow strip base under the pseudo-dynamic condition that is formulated by the method of limit analysis.

By introducing adaptive benefit factors in the basic SOS algorithm, three modified versions of the SOS algorithm are proposed to improve its efficiency of the basic SOS algorithm. This lay down a good balance between exploration and exploitation of the search space [26].

Therefore, with the aid of updated benefit factor, parasitism process, and random weight number search, the present research aims to improve the search component of the conventional SOS. The updated benefit factor is presented into the SOS to diminish the trap into neighborhood optima and to improve SOS convergence. The updated phase of parasitism is implemented to improve both the exploration and exploitation potential of the traditional SOS, consequently that an acceptable balance can be formed between exploration and exploitation. The random weight number is used to balance the capacity of exploitation and to increase the convergence rate. To the finest of our information, there's no variation comparable to what created in this paper. Within the proposed calculation, the covetous choice approach is additionally utilized inside the calculation to draw in the SOS toward the promising look spaces of domains space. The proposed algorithm is tested on twenty benchmark test problems with dimensions 100. The various success tests are also used to validate the importance of enhanced outcomes. The comparisons show that an expansion to a quicker meeting speed, the

suggested design accomplished the worldwide best with higher exactness. The effect of the suggested algorithm is also explored in the paper in several problems about engineering optimization. The findings on these issues also check the suggested algorithm’s superior search performance compared with the other comparative algorithms.

The remainder of this paper is composed as follows: Section 2, presents the traditional rendition of SOS and its inquiry component. In Section 3, the proposed mISOS is portrayed in detail. The experimentation and approval of the proposed calculation on benchmark test issues are acted in Section 4. The presentation of the proposed calculation about actual engineering problems is likewise talked about in Section 5. At long last, the conclusion of the work is furnished in Section 6 with some future research bearings.

2 Overview of basic SOS algorithm

Within nature, many species maintain an important relationship between survival and development. Often such partnerships may be helpful, or negative. In advantageous relationship species live respectively for shared advantage and endurance, which make it not quite the same as other normal connections. Considering the normal collaboration between the living being form for endurance in the biological system, Cheng and Prayogo [11] presented a convincing and unassuming meta-heuristic calculation called SOS which recreates the intelligent conduct among living being found in nature. In an ecosystem, a community of species is similar to the SOS algorithm population. Each organism speaks to one individual comparing to the optimization issue. That organism within the ecosystem has to do with an explicit fitness value, i.e. an objective function value that replicates the degree of adaptation to the defined goal. SOS implements the quest space into seven main components: initialization, Mutualism phase, selection-I, Commensalism phase, Selection-II, Parasitism phase, and selection-III.

Initialization The initial organism generates at random within the uniform search space at the initial level. According to Eq. (1), the initial organism is determined.

$$O_{i,d} = O_{lb,d} + rand(0, 1) * (O_{ub,d} - O_{lb,d}) \tag{1}$$

Here $O_{lb,d}$ and $O_{ub,d}$ are the lower and upper bound of the i^{th} organism respectively, d represents the dimension of the problem.

$$O_{p,d} = \begin{cases} O_{lb,d} + rand(0, 1) * (O_{ub,d} - O_{lb,d}) & \text{if } a < b \\ O_{i,d} & \text{Otherwise} \end{cases} ; a, b \in (0, 1) \tag{9}$$

Mutualism phase An organism, O_i interact with the randomly selected organism O_j from the ecosystem and all species communicate intending to extend shared survival capabilities within the environment in a mutualistic relationship according to Eqs. (2), (3), and (4).

$$O_{i,d}^{new} = O_{i,d} + rand(0, 1) * (O_{best} - M_V * BF1) \tag{2}$$

$$O_{j,d}^{new} = O_{j,d} + rand(0, 1) * (O_{best} - M_V * BF2) \tag{3}$$

$$M_V = 0.5 * (O_{i,d} + O_{j,d}) \tag{4}$$

Here O_{best} is the best organism in the ecosystem. The benefit factors (BF1 and BF2) are randomly calculated as either 1 or 2. Such variables reflect the degree of benefit for each organism, i.e. whether the interaction benefits an organism partially or entirely.

At the end of the mutualism phase, the selection-I operator is led by looking at the objective function value of the new aspirant organism with the corresponding old organism utilizing Eqs. (5) and (6)

$$O_i = \begin{cases} O_i^{new} & \text{if } f(O_i^{new}) < f(O_i) \\ O_i & \text{Otherwise} \end{cases} \tag{5}$$

$$O_j = \begin{cases} O_j^{new} & \text{if } f(O_j^{new}) < f(O_j) \\ O_j & \text{Otherwise} \end{cases} \tag{6}$$

Commensalism phase: Similar to mutualism phase, an organism, O_i interact with the randomly selected organism O_k from the ecosystem and only O_i takes the beneficial advantage within the environment in a commensalism relationship according to Eq. (7),

$$O_{i,d}^{new} = O_{i,d} + rand(-1, 1) * (O_{best} - O_{k,d}) \tag{7}$$

At the end of the commensalism phase, the selection-II operator is led by looking at the objective function value of the new aspirant organism with the corresponding old organism utilizing Eq. (8).

$$O_i = \begin{cases} O_i^{new} & \text{if } f(O_i^{new}) < f(O_i) \\ O_i & \text{Otherwise} \end{cases} \tag{8}$$

Parasitism phase By duplicating organism, O_i , an artificial parasite called $O_{p,d}$ is produced according to Eq. (9). At the end of production of $O_{p,j}$ using selection-III operator, $O_{p,j}$ led to

kill randomly selected organism $O_{i,d}$ utilizing Eq. (10).

$$O_i = \begin{cases} O_{p,d} & \text{if } f(O_{p,i}) < f(O_i) \\ O_{i,d} & \text{Otherwise} \end{cases} \quad (10)$$

The flowchart of the SOS algorithm is shown in Fig. 1.

3 The procedure of proposed mSOS algorithm

The search proficiency of the conventional SOS is overhauled in this segment by combining principles of the modified adaptive benefit factors, modification of parasitism phase, and random weighed reflection vector. Such alterations are portrayed and talked about in detail as underneath.

3.1 Modification of beneficial factor

In the fundamental SOS algorithm, the value of the benefit factor is considered as 1 or 2. This speaks that the interaction of the living being is halfway or completely benefited. Benefit factors are, however, the main control element in the SOS mutualism phase of the Mutual Vector. In the basic SOS algorithm, the beneficial factors (BF1 and BF2) are stochasti-

cally measured by either one or two, which shows whether the organism is enjoying the interaction partially or fully [20]. Such beneficial influences are heuristic in nature, because one organism may benefit partially or fully than another [11]. Such variables reflect the degree of the gain relationship, i.e., total or fractional. The new organism O_i^{new} and O_j^{new} enter the set of the organism, if the objective function value of these organisms are better than their corresponding pre-interaction organism O_i^{old} and O_j^{old} . This process is equivalent to greedy selection.

On the other hand, the organism O_i and O_j can get halfway and completely advantage from the mutual vector. It means that the algorithm quest will be successful with a small step if the value of the benefit factor is greater, but the convergence of the algorithm will diminish [26]. Similarly, if the greater value of the benefit factor is taken into account, the search accelerates the nearby value that decreases the ability of the algorithm to be used [26, 27]. Therefore, the benefit factor can vary within the range of 1 and 2. This rouses us for changing benefit factor (BF1 and BF2) adaptively which gives great union, predominant inquiry capacity, and congruity among investigation and misuse. The modification form of the benefit factors (aBF_1 , and aBF_2) are shown in Algorithm 1.

Algorithm 1. Modification of beneficial factor:

$$\text{if } (f_{best_{eco}} - f_{worst_{eco}} \neq 0)$$

$$aBF_1 = \text{round} \left(1 + \text{abs} \left(\frac{f_i - f_{worst_{eco}}}{f_{best_{eco}} - f_{worst_{eco}}} \right) \right), \quad (11)$$

else

$$aBF_1 = \text{round} \left(\frac{(3 + k)}{2} \right), k \in (0,1) \quad (12)$$

end

$$\text{if } (f_{best_{eco}} - f_{worst_{eco}} \neq 0)$$

$$aBF_2 = \text{round} \left(1 + \text{abs} \left(\frac{f_j - f_{worst_{eco}}}{f_{best_{eco}} - f_{worst_{eco}}} \right) \right), \quad (13)$$

else

$$aBF_2 = \text{round} \left(\frac{(3 + k)}{2} \right), k \in (0,1) \quad (14)$$

end

Here, f_i , f_j , $f_{worst_{eco}}$ and $f_{best_{eco}}$ are the objective function value of the i^{th} , j^{th} , worst, and best organisms respectively. Thus, during the execution process, the value of benefit

factors can be conserved repeatedly. In the mutualism phase, the term $M_V * aBF$ offers a decent harmony among investigation and taking advantage of the relationship trademark

between the living being O_i and O_j contrasted to those of starting BF1 and BF2, because of the adaptive benefits factors are implemented. So, the component of mutualism phase ($O_{best} - M_V * aBF$) can prompt great assorted variety with quicker convergence. Finally, the new organism O_i^{new} and O_j^{new} tried to minimize (for minimization problem) the fitness value of the optimization problem in the entire ecosystem. Accordingly, aBF drives the suggested method to investigate the untracked area in domain space when an organism ('I' or 'j') is left from the best organism and aids in increment the convergence rate. It demonstrates that the proposed method with aBF prompts the optimum solution globally, with a decent harmony among investigation and taking advantage of the proposed algorithm.

3.2 Modification of parasitism phase

A step of parasitism is necessary for the upgrade of the SOS exploration capability. Be that as it may, it is additionally encountered that over investigation brings about the higher computational expense. A significant number of new candidate solutions get dismissal because of second rate fitness value contrasted with the past one. In the parasitism period of essential SOS, the investigation rate is underprivileged, since the parasite vector produces in search space with a combination of structure variable with an arbitrary produced variable [20]. This solitary outcome progresses the existing outcome to increases this algorithm's exploitability. The primary purpose behind the change is to expel the disadvantage of the low exploitability of the parasitism stage. Numerous examinations show that the exploitability of the parasitism stage in SOS is impressively low when contrasted with the explorative

ability [20, 26, 27]. Expanding the quantity of FE prompts an expansion in the convergence time as well. Moreover, numerous look into a demonstration of the enhancement in the proficiency of the design algorithm with the change in the parasitism stage [20, 26–28]. Along these lines, this stage is upgraded with the alteration of the parasitism stage. Here it is attempted to progress exploitability of parasitism stage with keeping up global optimum also in domain space. In this manner, our thought process is to set an ideal harmony among the investigation and exploitability of the algorithm. Within the suggested algorithm, the investigation is empowered utilizing the best solution and one current solution. The procedure upgrades the assorted variety of populace and solution too. The suggested methodology permits the Algorithm to investigate various areas of the inquiry space simultaneously, keep away from the populace fixation in one district, and maintain a strategic distance from untimely intermingling. Algorithm 2 speaks to the alteration in the parasite vector.

In Algorithm 2, $O_{max,d}^{best}$ and $O_{min,d}^{best}$ are the maximum and minimum dimensions of the best organism respectively. From the algorithm, it is seen that, if a is less than b then the selected dimension is modified by search boundary otherwise the dimension modified with the help of the best organism. In this case, if the fitness value of the problem for the new calculated parasitism vector gives inferior (for minimization issue) than the past one, at that point the parasitism vector precedes the new location whereas taking out past living being. All things considered, the suggested algorithm can merge quicker while keeping up great assorted variety. Due to previously mentioned alteration in parasitism stage exploitability capacity increment with high intermingling rate and solidness of ideal solution.

Algorithm 2. Modification of Parasitism phase

$$\begin{aligned}
 & \text{if } a < b, \quad a, b \in (0,1) \\
 & P_V_d = O_{lb} + rand(0,1) * (O_{ub} - O_{lb}) \tag{15}
 \end{aligned}$$

$$\begin{aligned}
 & \text{else} \\
 & P_V_d = O_d + rand(0,1) * (O_{max,d}^{best} - O_{min,d}^{best}) \tag{16}
 \end{aligned}$$

end

3.3 Weighted random number

Within the conventional SOS, the living being stage causes the algorithm to continue by moving the mean of the living being towards its worldwide optima. To get another

arrangement of improved living being an arbitrary random weighted number is shaped and added to the current creature of the biological system. Also, in the mutualism stage, the algorithm continues by arbitrary cooperation among creatures to improve their relationship. To get another

arrangement of improved living being a random weighted number is shaped dependent on the mix of the random weighted differential vector [29] and the random weighted reflection vector [23]. This is found in algorithm 3.

Algorithm 3. Random weighted number:

if $a < b$

$$R = 1 - 0.5 * (1 + k) \quad (17)$$

else

$$R = 0.5 * (1 + k), k \in (0,1) \quad (18)$$

end

This random weighted number is proposed to upgrade the pursuit capacity of the design algorithm. Therefore, the strength of the algorithm depends on the assessment of standard deviation increments, and the calculation requests fewer quantities of investigations to combine a worldwide optimal solution. In the long run investigation capacity of stages is increment because of usage of random weighted number.

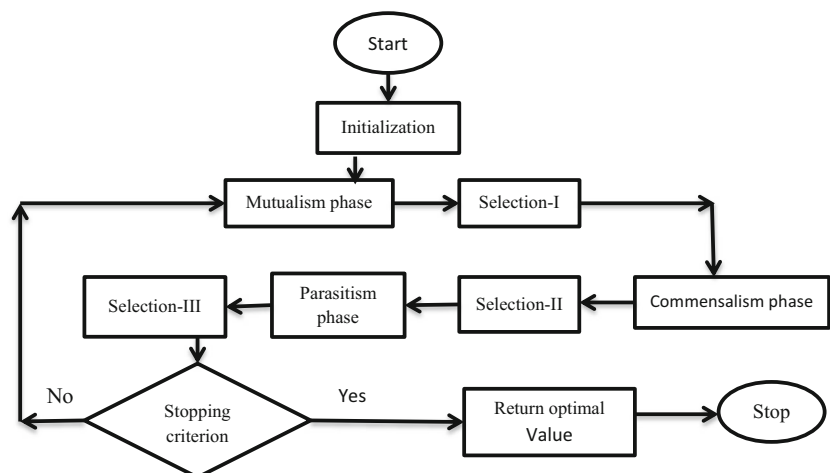
3.4 The framework of mISOS

In this subsection, the proposed mISOS architecture is introduced. It was detected after the literature that, because of the benefit factor in the mutualism process, the traditional SOS getting caught at neighborhood optima, less assorted variety, and slow convergence rate [20, 26].

Accordingly, the alteration of benefit factor is acquainted in SOS with keeping the living being from stagnation at nearby optima [26]. The parasitism process is used to discover and manipulate the domain space circa the finest candidate solution while increasing the convergence velocity of the original algorithm [20]. The updated process of parasitism is fundamentally centered around sparing computational time yet at the same time keep up the worldwide capacity of domain space. Integrating a random weighted number has improved the organism's exploration capacity, and this will be productive whereas fathoming the optimization issues having a huge number of nearby optima moreover [26].

The procedures which are connected within the mISOS can be summarized as takes after:

Fig. 1. Flowchart of the SOS algorithm



Algorithm 4. A modification based improved SOS (mISOS) algorithm

-
- (1) **Input:**
- (2) Eco-Size: The number of Organism in the ecosystem;
- (3) D: Number of variable of the optimization problem;
- (4) FES_max: Total number of function evaluations.
- (5) **Initialization:**
- (6) Initialized a group of Organisms Org_i ($i=1, 2, 3 \dots$ Eco-Size) randomly to the search space;
- (7) Evaluate the objective function values of each organism;
- (8) FES= Eco-Size; % Count the number of objective function values evaluations %;
- (9) **Main loop**
- (10) **While** FES < FES_max
- (11) Identify the best Organism according to their objective function value;
- (12) For each organism Org_i
- (13) **Apply Mutualism Phase**
- (14) Randomly select one Organism Org_j ($i \neq j$);
- (15) Evaluate benefit factors (aBF1 and aBF2), random weighted no. according to algorithm 1 and 3.
- (16) Determine new organism Org_i^{new} and Org_j^{new} by Eqn. (2, 3)
- (17) Calculate the fitness value of the new Organisms Org_i^{new} and Org_j^{new} and update according to Eqn. (5) and (6);
- (18) FE = FE +2; % Count the number of objective function values evaluations %;
- (19) **Apply Commensalism Phase**
- (20) Randomly select one Organism Org_j ($i \neq j$);
- (21) Determine new organism Org_i^{new} according to Eqn. (7);
- (22) Calculate the fitness value of the new Organism and update Organism Org_i^{new} with the succor of Organism Org_j according to Eqn. (8);
- (23) FE = FE +1; % Count the number of objective function values evaluations %;
- (24) **Apply Parasitism Phase**
- (25) Randomly select one Organism Org_j ($i \neq j$);
- (26) Create a Parasite (Parasite_Vector) from Organism Org_i according to algorithm 2.
- (27) Calculate the fitness value of the Parasite_Vector. If the fitness value of Parasite_Vector is not better than Org_j will have immunity from the parasite and the Parasite_Vector will no longer be able to live in that ecosystem.
- (28) FE = FE +1; % Count the number of objective function values evaluations %;
- (29) **End**
- Output:** The best Organism with the minimum fitness function value in the ecosystem(for minimization problem);
-

1. The modification of benefit factor is utilized within the mISOS to hop out from neighborhood optima and to quicken the look prepare.

2. The parasitism phase is presented to upgrade the investigation of look space amid the look prepare, moreover, when the current organism is absent from the obscure optimal organism.

3. The random weight number is utilized to travel the current organism form to other more promising look space.

4. After each phase i.e. mutualism phases, commensalism phases, and parasitism phases, the selection operator is used between two continuous emphases of the algorithm maintains a strategic distance from the living being to wander from promising pursuit areas and controlled the progression of decent variety inside the algorithm.

The investigation, exploitability, and the harmony between them are fundamental segments of any nature-enlivened algorithm. In the mISOS, these parts have attempted to improve

dependent on modified benefit factor, modified parasite factor, and random weighted number. The steps of the algorithm are exhibited in Algorithm 4 and the flowchart is introduced in Fig. 2.

4 Performance analysis on 20 test function experiments and comparison of results

4.1 Benchmark set analysis of results and parameter setting

The proposed mISOS is evaluated on a set of twenty notable and traditional benchmark problems [30]. These functions are scalable, and thus the scalability of the mISOS is also evaluated on these metrics, taking into account the dimension one hundred. Table 1 presents the names, mathematical form, quest range, and optimum values (Fmin) of traditional benchmark test functions. On these test functions, every algorithm is executed thirty times independently. The parameter settings of all the algorithms are presented in Table 2. Tables 3, 4, 5, 6, and 7 report the statistical parameters obtained, such as mean, and standard deviation of benchmark problem values.

A comparison of the algorithms is important for selecting the appropriate algorithm for an optimization problem from several optimizers. This is as often as possible done by utilizing optimum results from exact examinations on benchmarks [39] because the theorem of non-free lunch [2] infers that there is no all-inclusive best calculation. The algorithms compared based on the quality of the solution achieved for such benchmark function and the essential computational necessities. In this work, the output findings are compared with some SOS variant and these are SOS [11], I-SOS [23], SOS-ABF1 [27], SOS-ABF2 [27] and SOS-ABF1&2 [27]. Also, some other optimization algorithm BSA [10], ABSA [31], CLPSO [32],

CPSO-H [33], FDR-PSO [34], FI-PS [35], UPSO [36], EPSDE [37], TSDE [38], CPI-DE [39], ACoS-PSO [40], HBSA [30] and DSOS variants [47]. The reason behind the chosen algorithm for comparison is that the algorithms are widely used for solving the different complex problems from a different branch of science and engineering. The common control.

The parameter value of all the algorithms is thirty independent trials, fifty organisms, and $D \times 100$ function evaluations. Also, the parameter values of all algorithm-specific common control are taken the same as given in their original manuscript. For every algorithm, the maximum number of function evaluation is considered as the stopping criteria and every algorithm run in Matlab R2010a with Lenovo Intel(R) Core(TM) i5-8250 U, 8th generation CPU @1.60GHz 1.80GHz, 8GB RAM Windows 10 home and an $\times 64$ -based processor.

The outcomes are listed in the “Mean \pm STD” format. Mean “and “STD “reflect the benchmark problems values average and standard deviation respectively. Boldface represents the best outcome among the algorithms compared in this study. F: Function, S: Search space.

In the table, the signs $-/+/\approx$ will be used to show that the output of the corresponding competitor in terms of numerical results is worse than, better than, and identical to that of mISOS.

4.2 Comparison with conventional SOS and some improved SOS variant

The proposed mISOS are compared in this section to the conventional SOS and some modified SOS variant on twenty test problems. Table 3 provides a comparison of the tests for dimension 100. In these tables, the test problem values in terms of mean and standard deviation are specified. The output

Fig. 2. Flowchart of mISOS Algorithm

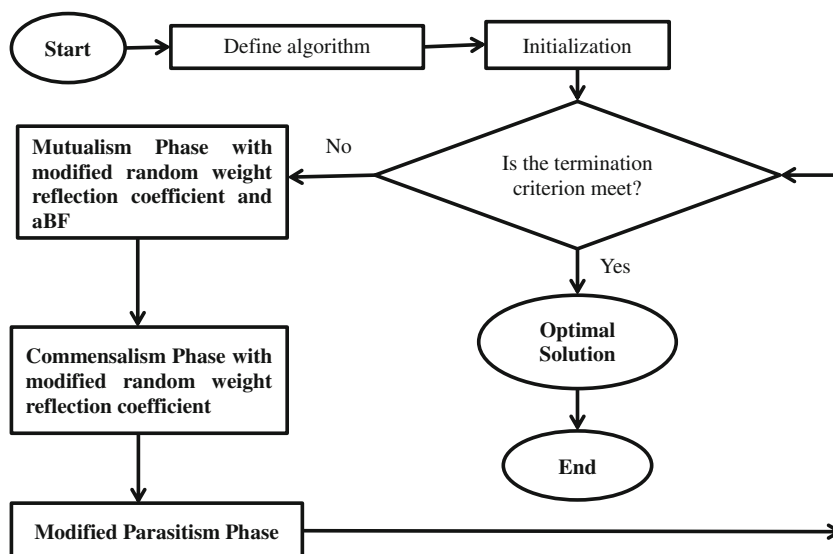


Table 1 Benchmark function applied for the validation of the proposed method ($F_{\min} = 0$, S =Search space)

Functions	Formulation	S
F1.Sphere	$f(x) = \sum_{i=1}^D x_i^2$	[-100, 100]
F2.Schwefel2.22	$f(x) = \sum_{i=1}^D x_i + \prod_{i=1}^D x_i $	[-10, 10]
F3.Schwefel1.2	$f(x) = \sum_{i=1}^D \sum_{j=1}^i x_j^2$	[-100, 100]
F4.Schwefel2.21	$f(x) = \max \{ x_i , 1 \leq i \leq D\}$	[-100, 100]
F5.Rosenbrock	$f(x) = \sum_{i=1}^D [100(x_{i+1}-x_i^2)^2 + (x_i-1)^2]$	[-30, 30]
F6.Step	$f(x) = \sum_{i=1}^D (x_i + 0.5)^2$	[-100, 100]
F7.Quartic	$f(x) = \sum_{i=1}^D ix_i^4 + random(0, 1)$	[-1.28, 1.28]
F8.Schwefel	$f(x) = 418 : 9829n - \sum x_i \sin(\sqrt{ x_i })$	[-500, 500]
F9.Rastrigin	$f(x) = 10D + \sum_{i=1}^D [x_i^2 - 10\cos(2\pi x_i)]$	[-5.12, 5.12]
F10.Ackley	$f(x) = 20 + e - 20e^{\frac{1}{D}(\sum_{i=1}^D \sqrt{\frac{D}{\pi} x_i^2})} - e^{\frac{1}{D}(\sum_{i=1}^D \cos(2\pi x_i))}$	[-32, 32]
F11.Griewank	$f(x) = \sum_{i=1}^D \frac{x_i^2}{4000} - \prod_{i=1}^D \cos(\frac{x_i}{\sqrt{i}}) - 1$	[-600, 600]
F12.Penalized1	$f(x) = \frac{\pi}{D} \left\{ 10\sin^2(\pi y_i) + \sum_{i=1}^{D-1} (y_i-1)^2 [1 + 10\sin^2(3\pi y_{i+1})] \right\} + \sum_{i=1}^D u(x_i, 10, 100, 4)$ Where $y_i = 1 + \frac{1}{4}(x_i + 1)$, $u(x_i, a, k, m) = \begin{cases} k(x_i-a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i-a)^m & x_i < -a \end{cases}$	[-50, 50]
F13.Penalized2	$f(x) = 0.1 \left\{ 10\sin^2(\pi x_i) + \sum_{i=1}^{D-1} (x_i-1)^2 [1 + 10\sin^2(3\pi x_{i+1})] \right\} + \sum_{i=1}^D u(x_i, 5, 100, 4)$	[-50, 50]
F14.Salomon	$f(x) = 1 - \cos\left(2\pi\sqrt{\frac{D}{\sum_{i=1}^D x_i^2}}\right) + 0.1\sqrt{\sum_{i=1}^D x_i^2}$	[-100, 100]
F15.Zakharov	$f(x) = \sum_{i=1}^D x_i^2 + \left(\sum_{i=1}^D \frac{ix_i}{2}\right)^2 + \left(\sum_{i=1}^D \frac{ix_i}{2}\right)^2$	[-5.12, 5.12]
F16.Axis parallel hyper ellipsoid	$f(x) = \sum_{i=1}^D ix_i^2$	[-5.12, 5.12]
F17.Ellipsoidal	$f(x) = \sum_{i=1}^D (x_i-i)^2$	[-100, 100]
F18.Cigar	$f(x) = x_1^2 + 100000 \sum_{i=2}^D x_i^2$	[-10, 10]
F19.Exponential	$f(x) = 1 - \exp\left(-0.5 \sum_{i=1}^D x_i^2\right)$	[-1, 1]
F20.Cosine mixture	$f(x) = 00.1D + \sum_{i=1}^D x_i^2 - 0.1 \sum_{i=1}^D \cos(5\pi x_i)$	[-1, 1]

findings are compared with SOS[11], I-SOS[23], SOS-ABF1[27], SOS-ABF2[27] and SOS-ABF1&2[27] at dimension (D) 100, respectively, after achieving D*100 FES of 20 test functions over 30 test runs with a population size of 50.

Only one optimum known as global optima is present in these problems which are for F6. As shown in Table 3, on fifteen, thirteen, thirteen, and nineteen test functions, mISOS performs better than SOS, I-SOS, SOS-ABF1, SOS-ABF1, and

Table 2 Parameter values of all the algorithm specific control parameters

Algorithm	Parameter values
SOS	BF1 = round(1 + rand); BF2 = round(1 + rand);
I-SOS	BF1 = round(1 + rand); BF2 = round(1 + rand);
SOS-ABF1	BF2 = round(1 + rand);
SOS-ABF2	BF1 = round(1 + rand);
SOS-ABF1&2	–
DSOS	w = 0.5, BF1 = round(1 + rand); BF2 = round(1 + rand);
mISOS	–
BSA	F = 3*rndn, where, rndn $\in N(0, 1)$; mix rate(M) = 1,
ABSA	$k_1 = 0.9, k_2 = 0.9, k_3 = 1.0,$ and $k_4 = 1.0,$
CLPSO	$w_0 = 0.9, w_1 = 0.4, c = 1.49445, m = 7, P_c = 0.05$ to 0.5,
CPSO-H	$w_{max} = 0.9, w_{min} = 0.5, c_1 = 1.49, c_2 = 1.49,$
FDR-PSO	$w_{max} = 0.9, w_{min} = 0.5, c_1 = 1, c_2 = 1, c_3 = 2,$
FI-PS	$w_{max} = 0.9, w_{min} = 0.5, c_1 = 2, c_2 = 2,$
UPSO	$c_1 = 1.49445, c_2 = 1.49445; w = 0.729; u = 0.1; \mu = 0; \sigma = 0.01;$
EPSDE	F = [0.4; 0.5; 0.6; 0.7; 0.8; 0.9]; CR = [0.1; 0.2; 0.3; 0.4; 0.5; 0.6; 0.7; 0.8; 0.9];
TSDE	The parameter candidate pool: [F = 1, CR = 0.1], [F = 1, CR = 0.9], and [F = 0.8, CR = 0.2].
CPI-DE	F = 0.9; CR = 0.5; $\delta^g = 1,$
ACoS-PSO	$c_1 = 1.4962; c_2 = 1.4962; w = 0.7298; \sigma = 0.05,$
HBSA	F = 0.9; DIM_RATE = 1;
DE	F = 0.5; CR = 0.9;

SOS-ABF1&2 respectively. In the charts, the bold font shows better results. Overall, as shown in Table 3, except for the F5, F 8, F9, F13, and F17 functions, mISOS obtained better results than other algorithms compared.

4.3 Comparison with other optimizers

The execution of the suggested mISOS is contrasted in this paragraph with other optimizers in terms of the numerical value of the test functions which are used in the previous section. These recent optimizers are BSA [10], ABSA [31], CLPSO [32], CPSO-H [33], FDR-PSO [34], FI-PS [35], UPSO [36], EPSDE [37], TSDE [38], CPI-DE [39], ACoS-PSO [40] and HBSA [30], and used to evaluate the mISOS performance. We held the same population size for a reasonable comparison, and a similar introductory population is picked for every algorithm for a specific run. Algorithm parameter settings are equivalent to utilize in their main papers. The examination is acted in Tables 4 and 5 which demonstrations the mISOS 'search effectiveness and solution accuracy. In a large portion of the test issues, either the proposed mISOS has accomplished the optima, beat, or serious versus different calculations.

The suggested mISOS either reached the optima, outflanked, or exceptionally competitive with other algorithms. In Table 6, the signs – + \approx are utilized to speak to that the execution of the comparing competitor in terms of numerical results is worse than, better than, and

comparable to that of mISOS. As shown in Table 6, mISOS performs better than BSA, ABSA, CLPSO, CPSO-H, FDR-PSO, FI-PS, UPSO, EPSDE, TSDE, CPI-DE, ACoS-PSO and HBSA on 18/20, 19/20, 19/20, 19/20, 18/20, 18/20, 18/20, 18/20, 19/20, 18/20, and 18/20 test functions, respectively.

4.4 Comparison with some recent algorithms

Table 7 the execution of the suggested mISOS with some recent optimizers in terms of the numerical value of the test functions which are used in the previous section. These recent optimizers are SOS and its variants called DSOS [47]. We held the same population size for a reasonable comparison, and a similar introductory population is picked for every algorithm for a specific run. Algorithm parameter settings are equivalent to utilize in their main papers [47]. The examination presented in Table 7, demonstrates the mISOS 'search effectiveness and solution accuracy. The suggested mISOS either reached the optima, outflanked, or exceptionally competitive with other algorithms. In Table 7, the signs – + \approx are utilized to speak to that the execution of the comparing competitor in terms of numerical results is worse than, better than, and comparable to that of mISOS. As shown in Table 7, mISOS performs better than SOS, DOS1, DOS2, DSOS3, DSOS4, DSOS5 and DSOS6 on 11/5/1, 11/5/1, 12/7/0, 12/7/0, 12/7/0, 12/7/0, 12/7/0 test functions, respectively.

Table 3 Performance results of SOS [11], I-SOS [23], SOS-ABF1 [26], SOS-ABF2 [26], SOS-ABF1&2 [26] and mSOS at dimension (D) 100 after reaching D*100 FEs of 20 test functions over 30 runs with 50 population size

F	SOS	I-SOS	SOS-ABF1	SOS-ABF2	SOS-ABF1&2	mSOS
F1	1.06e-007 ± 6.61e-008	6.97e-004 ± 1.40e-003	1.76e-004 ± 6.91e-005	1.47e-005 ± 5.62e-006	3.54e+002 ± 2.37e+002	3.78e-010 ± 1.58e-010
F2	1.57e-004 ± 5.35e-005	1.09e-002 ± 1.16e-002	3.87e-004 ± 1.32e-004	2.50e-004 ± 7.56e-005	1.46e+026 ± 7.85e+026	1.92e-005 ± 1.01e-005
F3	6.10e-006 ± 5.68e-006	6.79e-003 ± 7.90e-003	8.88e-003 ± 3.13e-003	6.99e-004 ± 2.79e-004	1.60e+004 ± 1.31e+004	2.00e-008 ± 1.05e-008
F4	4.29e-003 ± 1.76e-003	2.25e-002 ± 2.69e-002	2.35e-003 ± 7.92e-004	2.11e-003 ± 5.42e-004	1.76e-003 ± 6.68e-004	9.66e-005 ± 2.79e-005
F5	9.81e+001 ± 3.55e-001	3.73e+001 ± 3.41e+001	9.84e+001 ± 1.23e-001	9.82e+001 ± 2.60e-001	2.88e+006 ± 2.98e+006	9.87e+001 ± 4.85e-002
F6	0.00e+000 ± 0.00e+000	0.00e+000 ± 0.00e+000	0.00e+000 ± 0.00e+000	0.00e+000 ± 0.00e+000	3.20e+002 ± 2.72e+002	0.00e+000 ± 0.00e+000
F7	8.67e-003 ± 3.12e-003	6.69e-003 ± 3.28e-003	1.48e-002 ± 5.57e-003	1.09e-002 ± 4.25e-003	5.06e+000 ± 4.85e+000	4.72e-003 ± 1.82e-003
F8	3.90e+004 ± 2.98e+003	1.59e+001 ± 1.44e+001	3.07e+004 ± 7.88e+002	3.08e+004 ± 7.16e+002	3.07e+004 ± 9.00e+002	3.11e+004 ± 5.19e+002
F9	3.03e-005 ± 6.25e-005	1.56e-001 ± 7.15e-001	1.02e-002 ± 3.50e-002	2.03e-003 ± 3.59e-003	7.45e+002 ± 7.42e+001	3.28e+001 ± 5.08e+001
F10	3.35e-005 ± 1.02e-005	3.53e-003 ± 6.43e-003	2.97e-005 ± 8.90e-006	2.45e-005 ± 7.27e-006	2.11e-004 ± 1.41e-004	1.85e-005 ± 7.18e-006
F11	1.26e-007 ± 3.61e-007	2.00e-003 ± 7.60e-003	1.19e-005 ± 6.40e-006	2.56e-006 ± 1.75e-006	4.65e+000 ± 4.22e+000	2.70e-009 ± 7.03e-009
F12	1.90e-001 ± 1.94e-003	1.59e-002 ± 5.31e-003	1.71e-001 ± 2.55e-003	1.69e-001 ± 2.62e-003	7.85e+006 ± 8.16e+006	1.88e-001 ± 3.18e-003
F13	6.11e+000 ± 1.08e+000	7.36e-001 ± 2.62e-001	6.84e+000 ± 7.13e-001	6.51e+000 ± 7.58e-001	1.04e+007 ± 1.19e+007	1.03e+001 ± 4.86e-001
F14	2.96e-001 ± 3.98e-002	2.87e-001 ± 7.11e-002	2.76e-001 ± 3.45e-002	2.88e-001 ± 4.07e-002	2.91e-001 ± 4.40e-002	2.68e-001 ± 4.65e-002
F15	2.58e+002 ± 8.03e+001	1.68e+002 ± 3.15e+001	4.81e+002 ± 1.53e+002	4.64e+002 ± 1.29e+002	6.96e+002 ± 1.60e+002	6.00e+001 ± 1.28e+001
F16	1.09e-008 ± 7.39e-009	1.07e-004 ± 2.16e-004	2.28e-005 ± 9.39e-006	1.86e-006 ± 7.30e-007	4.30e+001 ± 6.52e+001	5.34e-011 ± 2.00e-011
F17	4.27e+004 ± 1.01e+004	5.59e+004 ± 4.61e+003	2.90e+004 ± 5.05e+003	2.66e+004 ± 4.47e+003	3.04e+004 ± 2.95e+003	1.61e+005 ± 9.85e+003
F18	6.17e-005 ± 3.28e-005	3.65e+000 ± 1.10e+001	1.75e-001 ± 6.07e-002	1.52e-002 ± 4.98e-003	2.21e+005 ± 1.84e+005	4.65e-007 ± 2.06e-007
F19	4.96e-012 ± 3.15e-012	4.00e-008 ± 1.23e-007	3.96e-009 ± 1.98e-009	4.47e-010 ± 2.48e-010	1.43e-003 ± 1.61e-003	2.36e-013 ± 1.13e-013
F20	1.13e-010 ± 8.87e-011	6.85e-007 ± 1.66e-006	7.83e-008 ± 3.52e-008	9.34e-009 ± 3.88e-009	2.15e-002 ± 2.75e-002	1.90e-012 ± 1.09e-012
-	15/20	13/20	13/20	13/20	19/20	
+	4/20	6/20	6/20	6/20	1/20	
≈	1/20	1/20	1/20	1/20	0/20	

Table 4 Performance results of ten function (F1-F10) at dimension (D) 100 after reaching D*100 FEs over 30 runs with 50 population size

Function	F1	F2	F3	F4	F5
miSOS	3.78e-010 ± 1.58e-010	1.92e-005 ± 1.01e-005	2.00e-008 ± 1.05e-008	9.66e-005 ± 2.79e-005	9.87e+001 ± 4.85e-002
BSA	1.86e+004 ± 1.48e+004	1.00e+002 ± 2.29e+001	8.22e+005 ± 5.59e+005	4.26e+001 ± 8.46e+000	8.02e+006 ± 1.38e+007
ABSA	1.66e+005 ± 1.37e+004	4.04e+036 ± 1.54e+037	7.78e+006 ± 7.96e+005	9.16e+001 ± 1.93e+000	5.67e+008 ± 7.97e+007
CLPSO	5.48e+004 ± 4.12e+003	2.85e+002 ± 1.61e+002	2.43e+006 ± 2.53e+005	7.71e+001 ± 3.87e+000	7.75e+007 ± 1.29e+007
CPSO-H	7.08e+002 ± 1.29e+002	1.97e+001 ± 1.87e+000	3.99e+004 ± 1.15e+004	9.39e+001 ± 1.36e+000	1.08e+006 ± 2.13e+005
FDR-PSO	5.34e+002 ± 1.55e+002	1.38e+001 ± 2.46e+000	2.55e+004 ± 8.15e+003	3.02e+001 ± 3.00e+000	9.01e+004 ± 3.95e+004
FLPS	7.07e+004 ± 5.43e+003	2.30e+010 ± 8.74e+010	3.05e+006 ± 2.49e+005	7.85e+001 ± 2.67e+000	1.36e+008 ± 2.06e+007
UPSO	8.93e+003 ± 1.20e+003	1.99e+002 ± 4.36e+001	3.63e+005 ± 5.60e+004	8.59e+001 ± 3.64e+000	8.80e+006 ± 2.03e+006
EPSDE	7.08e+002 ± 2.01e+002	3.93e+001 ± 3.79e+001	3.19e+004 ± 9.25e+003	8.57e+001 ± 1.72e+001	9.59e+005 ± 1.85e+006
TSDE	1.83e+004 ± 3.22e+003	1.34e+002 ± 1.75e+001	8.77e+005 ± 1.19e+005	5.01e+001 ± 4.30e+000	9.25e+006 ± 3.05e+006
CPI-DE	2.75e+005 ± 1.14e+004	2.88e+048 ± 5.44e+048	1.30e+007 ± 7.53e+005	9.58e+001 ± 1.11e+000	1.12e+009 ± 9.61e+007
ACoS-PSO	1.05e+004 ± 2.86e+003	3.08e+002 ± 2.44e+001	6.64e+005 ± 1.73e+005	6.39e+001 ± 9.06e+000	7.09e+006 ± 7.46e+006
HBSA	8.83e+002 ± 1.60e+002	1.52e+001 ± 2.78e+000	3.58e+004 ± 8.22e+003	1.78e+001 ± 1.81e+000	6.27e+004 ± 2.74e+004

Function	F6	F7	F8	F9	F10
miSOS	0.00e+000 ± 0.00e+000	4.72e-003 ± 1.82e-003	3.11e+004 ± 5.19e+002	3.28e+001 ± 5.08e+001	1.85e-005 ± 7.18e-006
BSA	1.37e+004 ± 8.08e+003	1.48e+001 ± 1.75e+001	2.68e+004 ± 6.80e+002	8.22e+002 ± 6.80e+001	1.55e+001 ± 2.18e+000
ABSA	1.65e+005 ± 1.29e+004	9.05e+002 ± 1.42e+002	2.68e+004 ± 6.70e+002	1.20e+003 ± 4.84e+001	2.01e+001 ± 2.54e-001
CLPSO	5.29e+004 ± 4.00e+003	1.12e+002 ± 2.31e+001	2.41e+004 ± 1.09e+003	9.65e+002 ± 3.71e+001	1.79e+001 ± 2.27e-001
CPSO-H	7.88e+002 ± 1.82e+002	2.93e+000 ± 4.23e-001	2.06e+003 ± 3.35e+002	9.12e+001 ± 8.54e+000	9.37e+000 ± 4.98e-001
FDR-PSO	6.67e+002 ± 1.52e+002	1.11e+000 ± 2.20e-001	1.84e+004 ± 3.15e+003	2.81e+002 ± 4.02e+001	5.15e+000 ± 5.07e-001
FLPS	6.92e+004 ± 3.57e+003	2.06e+002 ± 3.02e+001	2.88e+004 ± 1.14e+003	1.13e+003 ± 3.50e+001	1.81e+001 ± 2.03e-001
UPSO	8.79e+003 ± 1.45e+003	1.18e+001 ± 2.83e+000	1.68e+001 ± 7.89e+001	7.37e+002 ± 5.05e+001	1.18e+001 ± 5.03e-001
EPSDE	7.78e+002 ± 1.92e+002	1.23e+000 ± 1.06e+000	2.91e+000 ± 2.79e+000	7.21e+002 ± 4.52e+001	5.07e+000 ± 5.14e-001
TSDE	1.91e+004 ± 4.67e+003	1.46e+001 ± 5.52e+000	2.83e+004 ± 6.82e+002	9.77e+002 ± 4.84e+001	1.35e+001 ± 8.61e-001
CPI-DE	2.66e+005 ± 1.14e+004	1.79e+003 ± 1.76e+002	3.76e+004 ± 1.04e+003	1.62e+003 ± 5.17e+001	2.09e+001 ± 7.17e-002
ACoS-PSO	1.26e+004 ± 5.18e+003	1.31e+001 ± 8.55e+000	2.13e+004 ± 2.59e+003	1.07e+003 ± 1.13e+002	1.27e+001 ± 1.66e+000
HBSA	8.55e+002 ± 1.95e+002	6.66e-001 ± 2.14e-001	4.52e+001 ± 5.06e+001	6.51e+002 ± 4.65e+001	5.23e+000 ± 4.38e-001

Table 5 Performance results of ten function (F11-F20) at dimension (D) 100 after reaching D*100 FEs over 30 runs with 50 population size

Function	F11	F12	F13	F14	F15
miSOS	2.70e-009 ± 7.03e-009	1.90e-001 ± 3.87e-003	1.03e+001 ± 4.86e-001	2.68e-001 ± 4.65e-002	6.00e+001 ± 1.28e+001
BSA	1.25e+002 ± 5.68e+001	2.18e+006 ± 7.31e+006	1.90e+007 ± 6.62e+007	1.46e+001 ± 3.42e+000	4.62e+002 ± 1.19e+002
ABSA	1.51e+003 ± 1.20e+002	1.22e+009 ± 2.43e+008	2.42e+009 ± 3.05e+008	4.70e+001 ± 2.07e+000	7.79e+002 ± 5.92e+001
CLPSO	4.89e+002 ± 4.36e+001	7.81e+007 ± 2.21e+007	2.27e+008 ± 4.48e+007	2.49e+001 ± 1.17e+000	8.26e+002 ± 6.64e+001
CPSO-H	7.81e+000 ± 1.66e+000	2.22e+000 ± 3.51e+000	2.64e+001 ± 6.74e+000	1.88e+001 ± 1.06e+000	8.86e+002 ± 7.55e+001
FDR-PSO	5.33e+000 ± 1.17e+000	1.31e+002 ± 2.58e+002	3.60e+002 ± 7.42e+001	6.11e+000 ± 6.21e-001	2.25e+002 ± 5.70e+001
FLPS	6.24e+002 ± 6.11e+001	2.00e+008 ± 4.19e+007	5.25e+008 ± 6.64e+007	2.74e+001 ± 1.14e+000	2.09e+003 ± 3.72e+002
UPSO	7.83e+001 ± 9.73e+000	4.33e+006 ± 2.21e+006	1.74e+007 ± 7.67e+006	1.44e+001 ± 1.04e+000	8.43e+002 ± 1.16e+002
EPSDE	7.82e+000 ± 2.24e+000	2.36e+006 ± 6.48e+006	8.64e+005 ± 2.51e+006	6.52e+000 ± 6.67e-001	1.61e+003 ± 6.42e+002
TSDE	1.74e+002 ± 4.17e+001	1.72e+006 ± 1.65e+006	1.39e+007 ± 8.02e+006	1.47e+001 ± 1.27e+000	1.13e+002 ± 3.06e+001
CPI-DE	2.39e+003 ± 1.26e+002	2.80e+009 ± 2.25e+008	5.19e+009 ± 3.92e+008	5.24e+001 ± 1.21e+000	3.89e+006 ± 1.01e+007
ACoS-PSO	1.14e+002 ± 4.63e+001	2.45e+006 ± 2.72e+006	1.22e+007 ± 9.19e+006	1.70e+001 ± 1.69e+000	6.52e+002 ± 1.18e+002
HBSA	8.33e+000 ± 1.94e+000	9.62e+000 ± 2.66e+000	1.12e+002 ± 3.19e+001	6.46e+000 ± 7.30e-001	5.64e+001 ± 1.29e+001

Function	F16	F17	F18	F19	F20
miSOS	5.34e-011 ± 2.00e-011	1.61e+005 ± 9.85e+003	4.65e-007 ± 2.06e-007	2.36e-013 ± 1.13e-013	1.90e-012 ± 1.09e-012
BSA	2.21e+003 ± 1.45e+003	1.16e+005 ± 1.90e+004	1.72e+007 ± 1.02e+007	4.93e-001 ± 1.40e-001	9.56e+000 ± 2.08e+000
ABSA	2.02e+004 ± 1.94e+003	1.88e+005 ± 2.15e+004	1.65e+008 ± 1.55e+007	9.69e-001 ± 1.75e-002	2.60e+001 ± 1.48e+000
CLPSO	6.52e+003 ± 6.95e+002	1.67e+005 ± 1.59e+004	5.31e+007 ± 5.19e+006	9.35e-001 ± 1.01e-002	1.43e+001 ± 6.27e-001
CPSO-H	1.03e+002 ± 2.16e+001	8.25e+002 ± 2.14e+002	7.99e+005 ± 1.73e+005	3.87e-002 ± 8.41e-003	9.24e-001 ± 1.47e-001
FDR-PSO	6.54e+001 ± 2.18e+001	6.30e+002 ± 2.08e+002	4.74e+005 ± 1.72e+005	2.38e-002 ± 5.68e-003	2.65e+000 ± 5.58e-001
FLPS	8.20e+003 ± 6.10e+002	8.26e+004 ± 5.48e+003	6.84e+007 ± 3.97e+006	9.67e-001 ± 9.64e-003	1.67e+001 ± 5.26e-001
UPSO	9.20e+002 ± 1.45e+002	2.64e+004 ± 3.57e+003	8.78e+006 ± 1.28e+006	3.52e-001 ± 3.41e-002	7.89e+000 ± 6.30e-001
EPSDE	9.26e+001 ± 2.16e+001	5.64e+003 ± 1.48e+003	7.52e+005 ± 1.79e+005	3.53e-002 ± 1.29e-002	9.18e-001 ± 3.37e-001
TSDE	2.12e+003 ± 3.57e+002	5.96e+004 ± 7.45e+003	2.00e+007 ± 4.15e+006	6.03e-001 ± 7.48e-002	1.08e+001 ± 5.69e-001
CPI-DE	3.38e+004 ± 2.34e+003	5.06e+005 ± 3.64e+004	2.65e+008 ± 1.05e+007	1.00e+000 ± 1.07e-006	3.68e+001 ± 1.22e+000
ACoS-PSO	1.64e+003 ± 4.99e+002	4.51e+003 ± 1.59e+003	1.05e+007 ± 3.37e+006	4.38e-001 ± 1.14e-001	1.11e+001 ± 1.39e+000
HBSA	8.75e+001 ± 2.24e+001	2.11e+004 ± 3.30e+003	8.87e+005 ± 2.43e+005	3.55e-002 ± 7.16e-003	1.30e+000 ± 3.51e-001

Table 6 Performance of the corresponding competitor is worse than, better than, and similar to that of mISOS

	BSA	ABSA	CLPSO	CPSO-H	FDR-PSO	FI-PS	UPSO	EPSDE	TSDE	CPI-DE	ACoS-PSO	HBSA
–	18/20	19/20	19/20	19/20	18/20	18/20	18/20	18/20	18/20	19/20	18/20	18/20
+	1/20	1/20	1/20	1/20	2/20	2/20	2/20	2/20	2/20	2/20	2/20	2/20
≈	0/20	0/20	0/20	0/20	0/20	0/20	0/20	0/20	0/20	0/20	0/20	0/20

4.5 Statistical analysis

In addition, simultaneously the multiple-problem Wilcoxon test [41] and Friedman's test were executed using SPSS software to evaluate the substantial outperformance of the mISOS on the test functions. In this case, the Bonferroni-Dunn technique [42] was chosen as the post-hoc technique for Friedman's test. With regard to the 100D test functions, the average objective function values over 30 runs, the results of the Wilcoxon multi-problem test, and Friedman tests are condensed in Tables 8, 9, 10, 11, 12 and 13 separately.

In Tables 8, 10, and 12, all R+ values are greater than R- values, showing that mISOS efficiency is higher than other competitors' efficiency. Also, mISOS accomplish the first rank in the Friedman's test shown in Tables 9, 11, and 13. Consequently, the test results show that mISOS is outperforming the competitors with 30D and 100D on the classical test functions.

On the basis of the statistical analysis, we can observe that the rank of mISOS is one, while the other approaches are the following techniques, separately. In this way, the suggested mISOS can be viewed as a superior analyzer than different algorithms beneath thought for contrast to find the result of the optimization problem with high precision of the solution.

4.6 Convergence rate comparison

The convergence curves are illustrated in Figs. 3. were plotted using the objective function values of the respective functions. Those curves were plotted by setting dimensional size 100. The convergence curves are plotted for the test function are F1.Sphere; F3.Schwefel1.2; F4.Schwefel2.21; F6.Step; F11.Griewank; F14.Salomon; F15.Zakharov; F16.Axis parallel hyper ellipsoid; F18.Cigar; F19.Exponential; F20.Cosine mixture. It very well may be effectively seen from the figures that the mISOS converges rapidly towards the worldwide optima and accomplishes high exactness contrasted with other enhancement algorithms. On the other hand, the other advancement algorithms effectively fall into neighborhood optima, and along these lines, the convergence speed is moderate when different peers continue to switch worldwide optima. The global optima for the F14 and F15 functions lie in a narrow valley, and converging to that result is tough for optimization algorithms. Nonetheless, the suggested mISOS accomplishes this global optimum of up to one hundred and

propositions a result similar to global optima for the dimension superior to one hundred. Therefore, the suggested mISOS reveal its high precision and higher convergence rate associated with other methods of optimization.

Consequently, the general investigation of the outcomes on the test functions shows the substantial improvement in the pursuit technique of the mISOS when contrasted with the SOS variations and other comparative algorithms far as accomplishing the better convergence rate, upgrading, and adjusting the investigation and exploitability quality of the algorithm. Thus, the suggested mISOS can be viewed as a superior optimizer than others to solve the test problems with a large number of variables and high solution accuracy.

5 Applications of mISOS on engineering benchmark problems

In this subsection, the suggested modified version of SOS is executed on five well-known engineering test problems [43], namely, frequency modulation sounds parameter identification problem, Lennard-jones potential problem, tersoff potential function minimization problem, and speed reducer design. These problems are taken from ref. [43]. A summary of the numerical research engineering test problems is as follows:

5.1 Frequency modulation sounds parameter identification problem [42]

Frequency-Modulated (FM) sound wave is a global optimization problem and it has a significant role in several modern music systems. FM amalgamation is a proficient, however not in every case effectively controlled procedure for producing intriguing sounds. A minor number of parameters can be playing an important role to deliver a wide scope of sound tones. This can consequently produce sounds that are analogous to the target sounds. Utilizing mISOS in FM synthesis to find the optimized parameter values in which the FM matching synthesis technique is presented. Initially, a set of parameters value is determined using mISOS and the FM synthesizer generates the corresponding sounds. In this problem, the distances of features between the target sound and synthesized sounds are considered as the fitness value (or objective function value). This engineering benchmark problem is a highly complex multimodal problem. The dimension

Table 7 Comparison of mean error and standard deviation in objective function value for 30 dimension of classical test problems

F	SOS	DSOS-1	DSOS-2	DSOS-3	DSOS-4	DSOS-5	DSOS-6	miSOS
F1	1.17E-37(1.12E-37)	4.18E-68(2.93E-67)	1.50E-39(1.41E-39)	1.10E-39(1.00E-39)	3.04E-40(2.38E-40)	2.50E-40(2.18E-40)	3.34E-40(4.31E-40)	1.12e-71 ± 2.32e-71
F2	1.48E-19(9.09E-20)	7.57E-38(3.69E-37)	3.51E-23(2.02E-23)	4.18E-23(3.24E-23)	3.69E-23(4.06E-23)	3.38E-23(3.36E-23)	3.12E-23(2.65E-23)	3.34e-36 ± 3.11e-36
F3	5.84E-13(1.08E-12)	1.84E-57(9.86E-57)	4.15E-12(9.53E-12)	1.10E-11(3.39E-11)	1.77E-12(3.63E-12)	1.46E-12(2.48E-12)	2.16E-12(5.46E-12)	9.58e-71 ± 7.46e-71
F4	2.79E-15(1.85E-15)	7.28E-32(2.76E-31)	2.90E-16(1.96E-16)	2.15E-16(1.46E-16)	1.20E-16(7.59E-17)	9.90E-17(6.08E-17)	1.24E-16(7.87E-17)	7.08e-33 ± 4.73e-33
F5	2.35E+01(3.99E-01)	2.42E+01(4.68E-01)	2.27E+01(3.98E-01)	2.26E+01(3.79E-01)	2.28E+01(5.29E-01)	2.28E+01(4.21E-01)	2.29E+01(5.01E-01)	2.17e+01 ± 1.08e-01
F6	8.45E-10(9.87E-10)	1.63E-06(2.13E-06)	1.52E-13(1.83E-13)	4.38E-13(1.22E-12)	1.81E-13(2.48E-13)	1.66E-13(2.81E-13)	1.92E-13(2.19E-13)	0.00e+00 ± 0.00e+00
F7	3.86E-73(1.26E-72)	1.82E-134(1.10E-133)	5.27E-74(1.25E-73)	1.84E-74(4.14E-74)	3.01E-75(5.24E-75)	5.01E-75(3.11E-74)	1.02E-75(1.47E-75)	7.33e-04 ± 2.93e-04
F8	4.88E+03(4.71E+02)	4.39E+03(4.72E+02)	7.06E+03(4.63E+02)	5.34E+03(3.67E+02)	6.96E+03(5.00E+02)	5.69E+03(4.37E+02)	5.49E+03(6.37E+02)	1.77e+00 ± 1.92e+00
F9	0.00E+00(0.00E+00)	0.00E+00(0.00E+00)	7.04E+00(2.41E+01)	7.65E+00(2.36E+01)	5.85E+00(2.44E+01)	7.22E+00(2.67E+01)	3.81E+00(1.81E+01)	1.44e+01 ± 3.33e+01
F10	1.97E+01(2.81E+00)	1.96E+00(6.01E+00)	2.00E+01(9.12E-02)	2.01E+01(9.70E-02)	2.01E+01(1.44E-01)	2.01E+01(1.45E-01)	2.01E+01(1.67E-01)	4.58e-15 ± 6.96e-16
F11	0.00E+00(0.00E+00)	0.00E+00(0.00E+00)	1.32E-03(4.45E-03)	9.93E-04(4.03E-03)	3.32E-04(1.66E-03)	1.25E-03(3.94E-03)	2.01E-04(1.43E-03)	0.00E+00(0.00E+00)
F12	4.14E-11(6.26E-11)	6.35E-08(1.24E-07)	2.15E-13(7.49E-13)	8.96E-14(2.57E-13)	5.71E-14(1.73E-13)	1.11E-13(4.40E-13)	2.00E-13(4.15E-13)	1.25e-01 ± 4.64e-02
F13	8.75E-02(8.57E-02)	3.30E-01(1.67E-01)	4.06E-02(5.83E-02)	4.49E-02(5.98E-02)	3.14E-02(4.13E-02)	5.25E-02(8.04E-02)	4.42E-02(6.31E-02)	1.42e-01 ± 2.64e-01
F14	9.99E-02(3.76E-08)	9.60E-02(1.96E-02)	9.99E-02(6.59E-08)	9.99E-02(6.34E-08)	9.99E-02(9.79E-08)	9.99E-02(6.68E-08)	9.99E-02(5.19E-08)	1.47e-01 ± 5.03e-02
F15	7.34E-05(9.28E-05)	2.41E-35(1.51E-34)	1.31E-03(2.78E-03)	1.06E-03(1.78E-03)	7.75E-04(1.09E-03)	6.91E-04(7.63E-04)	6.31E-04(8.20E-04)	1.03e-17 ± 1.78e-17
F18	9.45E-31(1.31E-30)	7.64E-63(3.98E-62)	5.76E-33(5.84E-33)	4.51E-33(4.66E-33)	1.78E-33(2.24E-33)	1.39E-33(2.54E-33)	1.87E-33(2.66E-33)	6.37e-69 ± 5.41e-69
F19	7.18E-66(4.08E-78)	7.18E-66(3.19E-81)	7.18E-66(3.19E-81)	7.18E-66(3.19E-81)	7.18E-66(3.19E-81)	7.18E-66(3.19E-81)	7.18E-66(3.19E-81)	0.00e+00 ± 0.00e+00
11/5/1	11/5/1	11/5/1	12/7/0	12/7/0	12/7/0	12/7/0	12/7/0	→+/≈

Table 8 Results of the multiple-problem based Wilcoxon’s test for mISOS and some selected SOS variants on 20 test functions with 100D from ($\alpha = 0.05$)

mISOS vs. Algorithm	p Value	R+	R-	Winner
SOS	0.212	126	64	mISOS
I-SOS	0.968	96	94	mISOS
SOS-ABF1	0.936	97	93	mISOS
SOS-ABF2	0.968	96	94	mISOS
SOS-ABF1&2	0.003	184	26	mISOS

of this problem is six of the sound wave. The objective function is defined as follows:

$$\text{Minimize } f(y) = \sum_{t=0}^{100} (y(t) - y_0(t))^2 \tag{19}$$

Where the primary sound wave and target sound is given by Eqs. (20) and (21).

$$y(t) = a_1 \sin(\omega_1 t \theta) + a_2 \sin(\omega_2 t \theta + a_3 \sin(\omega_3 t \theta)) \tag{20}$$

$$y_0(t) = 1.0 * \sin(5.0 * t \theta + 1.5 * \sin(4.8 * t \theta + 2.0 * \sin(4.9 * t \theta))) \tag{21}$$

Six variables are $y = \{a_1, \omega_1, a_2, \omega_2, a_3, \omega_3\}$ and the domain of these variables is $[-6.4, 6.35]$. Here $\theta = 2\pi/100$.

5.2 Lennard-Jones potential problem [42]

Lennard-jones (LJ) potential problem is the minimization of molecular potential energy associated with pure LJ cluster. The lattice structure of the LJ cluster has an icosahedral core and a combination of surface lattice points. The algorithm mISOS is executed over this capacity for its ability to accommodate molecular structure, where the atoms are composed so that the particle has the least energy. This is a potential energy minimization problem of multimodal optimization comprising an infinite number of local minima. In LJ-problem, the cluster configuration of ‘N’ atoms is given by the Cartesian coordinates

$$\vec{p}_i = \{ \vec{x}_i, \vec{y}_i, \vec{z}_i \}, i = 1, 2, 3, \dots, N \tag{22}$$

The mathematical formation of Lennard-Jones pair potential for ‘N’ atoms is given as follows:

Table 9 Ranking of mISOS and some selected SOS variants by the Friedman’s test on 20 test functions with 100D

Algorithm	Mean Rank	Final rank
SOS	3.00	3
ISOS	3.70	5
SOSABF1	3.62	4
SOSABF2	2.85	2
SOSABF12	5.38	6
mISOS	2.45	1

Table 10 Results of the multiple-problem based Wilcoxon’s test for mISOS and some selected algorithms variants on 20 test functions with 100D from ($\alpha = 0.05$)

mISOS vs. Algorithm	p Value	R+	R-	Winner
BSA	0.004	183	27	mISOS
ABSA	0.000	200	10	mISOS
CLPSO	0.001	197	13	mISOS
CPSO-H	0.008	176	34	mISOS
FDR-PSO	0.009	175	35	mISOS
FI-PS	0.002	186	24	mISOS
UPSO	0.005	181	29	mISOS
EPSDE	0.005	180	30	mISOS
TSDE	0.004	183	27	mISOS
CPIDE	0.000	190	0	mISOS
ACoS-PSO	0.004	183	27	mISOS
HBSA	0.014	171	39	mISOS

$$V_N(p) = \sum_{i=1}^{N-1} \sum_{j=i+1}^N (r_{i,j}^{-12} - 2 \cdot r_{i,j}^{-6}) \tag{23}$$

Where

$$r_{i,j} = \left\| \vec{p}_j - \vec{p}_i \right\|_2 \tag{24}$$

And the gradient is

$$\begin{aligned} \nabla_j V_N(p) &= -12 \sum_{i=1, i \neq j}^{N-1} (r_{i,j}^{-14} - 2 \cdot r_{i,j}^{-8}) \cdot (\vec{p}_j - \vec{p}_i), j \\ &= 1, 2, 3, \dots, N \end{aligned} \tag{25}$$

In this case, 10 atom problems have been considered, and then the total number of variables is 30. Any permutation and translation of the ordering of $\{p_j\}$ results in an equivalent

Table 11 Ranking of mISOS and some selected others algorithm variants by the Friedman’s test on 20 test functions with 100D

Algorithm	Mean Rank	Final Rank
mISOS	2.11	1
BSA	7.61	8
ABSA	11.50	12
CLPSO	9.53	10
CPSOH	4.76	5
FDR-PSO	2.84	2
FI-PS	10.74	11
UPSO	6.47	6
EPSDE	4.29	4
TSDE	7.74	9
CPIDE	13.00	13
ACoS-PSO	6.89	7
HBSA	3.53	3

Table 12 Results of the multiple-problem based Wilcoxon’s test for mISOS and recent SOS variants on 17 test functions with 30D from ($\alpha = 0.05$)

mISOS vs. Algorithm	p Value	R+	R-	Winner
SOS	0.535	80	56	mISOS
DSOS1	0.569	79	57	mISOS
DSOS2	0.407	94	59	mISOS
DSOS3	0.407	94	59	mISOS
DSOS4	0.435	93	60	mISOS
DSOS5	0.435	93	60	mISOS
DSOS6	0.463	92	61	mISOS

function value since $V_N(p)$ depends only on the pair distances. The domain of first three variables are $[0, 4], [0, 4], [0, \pi]$ and for other variables are: $[-4 - \frac{1}{4} \lfloor \frac{t-4}{3} \rfloor, 4 + \frac{1}{4} \lfloor \frac{t-4}{3} \rfloor]$. Where $\lfloor r \rfloor$ is the nearest least integer w.r.t. r.

5.3 Tersoff potential function minimization problem for model Si (B)

The total Tersoff potential energy function of the system is the sum of individual potentials of atoms and defined as

$$f(\vec{X}_1, \vec{X}_2, \vec{X}_3, \dots, \vec{X}_N) = E_1(\vec{X}_1, \vec{X}_2, \vec{X}_3, \dots, \vec{X}_N) + \dots + E_N(\vec{X}_1, \vec{X}_2, \vec{X}_3, \dots, \vec{X}_N) \tag{26}$$

The Tersoff potential [11] of individual atoms can be formally defined as follows:

$$E_i = \frac{1}{2} \sum_{i \neq j} f_c(r_{i,j}) (V_R(r_{i,j}) - B_{i,j} V_A(r_{i,j})), \quad \forall i \tag{27}$$

Here $r_{i,j}$ is the distance between atoms i and j, V_R is a repulsive term, V_A is an attractive term, $f_c(r_{i,j})$ is a switching function, $B_{i,j}$ is a many-body term that depends on the positions of atoms i and j and the neighbors of atom i. The term $B_{i,j}$ is given by

Table 13 Ranking of mISOS and some recent algorithm variants by the Friedman’s test on 17 test functions with 30D

Algorithm	Mean rank	Final rank
SOS	5.29	7
DSOS1	3.26	2
DOSO2	5.50	8
DSOS3	5.41	5
DSOS4	4.38	3
DSOS5	4.50	6
DSOS6	4.47	4
mISOS	3.18	1

$$B_{i,j} = \left(1 + \gamma^n \xi_{i,j}^m\right)^{-1/2n_1} \tag{28}$$

In Eq. (28), n_1 and γ are known fitted parameters and the term $\xi_{i,j}$ for atoms, i and j (i.e., for bond ij) are given by:

$$\xi_{i,j} = \sum_{k \neq i} f_c(r_{i,j}) g(\theta_{ijk}) \exp\left(\lambda_3^3 - (r_{ij} - r_{ik})^3\right) \tag{29}$$

In Eq. (29), the term θ_{ijk} is the bond angle between bonds ij and ik, and the function g is given by:

$$g(\theta_{ijk}) = 1 + c^2/d^2 - c^2 / \left(d^2 + (h - \cos(\theta_{ijk}))^2\right) \tag{30}$$

The quantities λ_3 , and c, d, h which appear in Eqs. (29) and (30) are also known as fitted parameters. The terms $V_R(r_{i,j})$, $V_A(r_{i,j})$ and the switching function $f_c(r_{i,j})$ which appears in Eq. (27) are given by:

$$V_R(r_{i,j}) = A e^{-\lambda_1 r_{ij}} \tag{31}$$

$$V_A(r_{i,j}) = B e^{-\lambda_2 r_{ij}} \tag{32}$$

$$f_c(r_{i,j}) = \begin{cases} 1 & , r_{i,j} \leq R - D \\ \frac{1}{2} - \frac{1}{2} \sin\left(\frac{\pi(r_{i,j} - R)}{D}\right) & , R - D < r_{i,j} < R + D \\ 0 & , r_{i,j} \geq R + D \end{cases} \tag{33}$$

Where A, B, λ_1 R, D and λ_2 are given fitted parameters. Thus the cost function now can be redefined as:

$$f(x) = E_1(x) + E_2(x) + \dots + E_N(x), \quad x \in \Omega \tag{34}$$

In general, for the system of N atoms, the no. of unknown variables is $n = 3 \times N - 6$. Now the cluster X of N atoms can be redefined as

$$x = \{x_1, x_2, x_3, \dots, x_n\}, x \in IR^{3N-6} \tag{35}$$

The search region for Si (B) model of Tersoff potential is

$$\Omega = \{ \{x_1, x_2, x_3, \dots, x_n\} : -4.25 \leq x_1, x_i \leq 4.25; \tag{36}$$

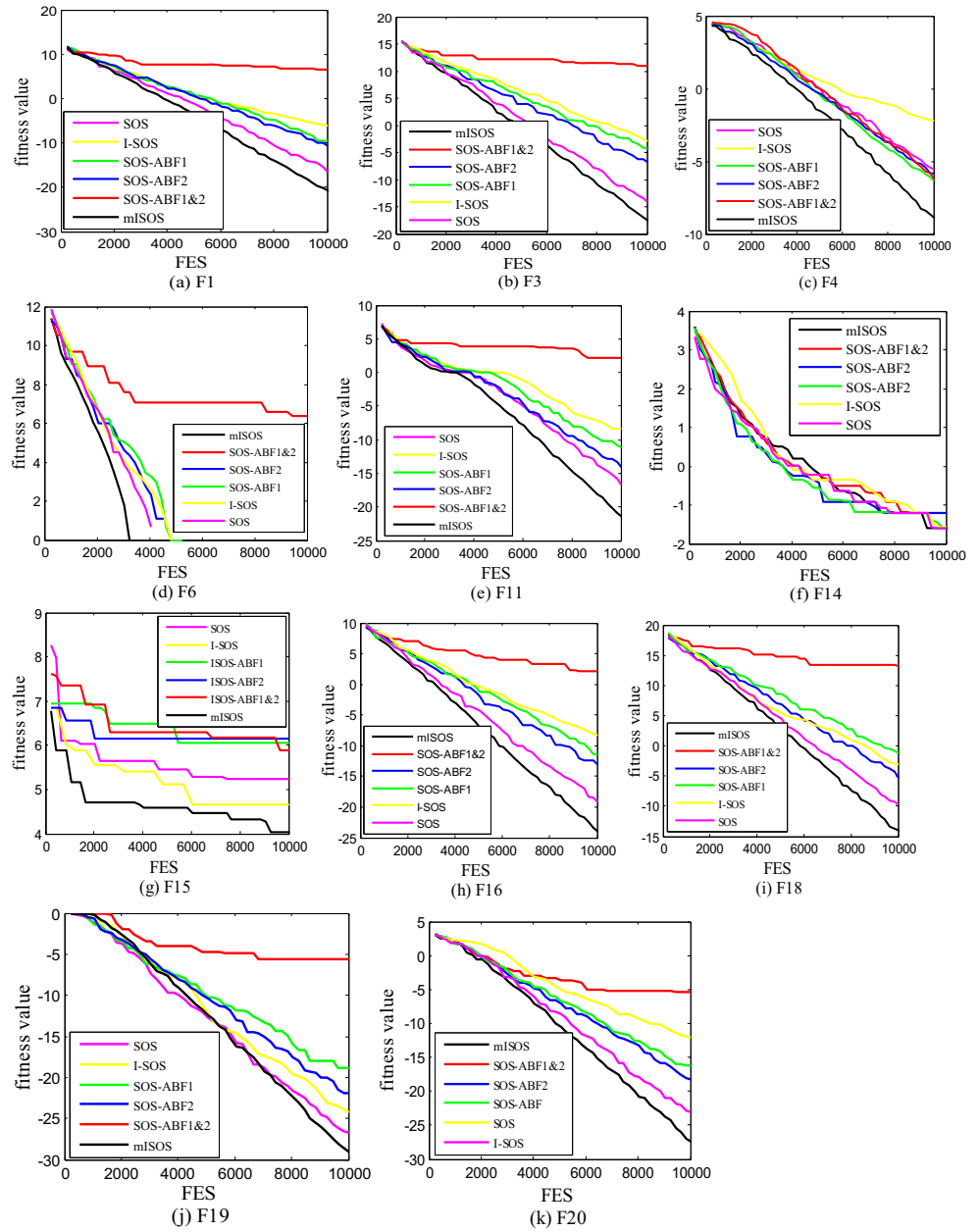
$$i = 4, 5, \dots, n; 0 \leq x_2 \leq 4; 0 \leq x_3 \leq \pi \}$$

In this case, the number of atoms is considered as 10 so that the total no of variables is 30. The detailed description of this problem can be seen in ref. [43].

5.4 Spread spectrum radar polyphase code design problem [43]

Spread spectrum radar polyphase code design problem is one of the most popular engineering design optimization problems for the application of global optimization algorithms. When designing a radar system that uses pulse compression, great

Fig. 3 Convergence curve of eleven test functions (F1, F3, F4, F6, F11, F14, F15, F16, F18, F19 and F20) with $D = 100$ and 10,000 FEs over objective function value vs. fitness evaluation (FES)



attention must be given to the choice of the appropriate waveform. Many methods of radar pulse modulation that make pulse compression possible are known. Polyphase codes are attractive as they offer lower side-lobes in the compressed signal and easier implementation of digital processing techniques. This polyphase pulse compression code synthesis is based on the properties of the aperiodic autocorrelation function and the assumption of coherent radar pulse processing in the receiver. The problem has continuous variables and with numerous local optimizations as a min-max non-linear non-convex optimization problem. The mathematical formation of this problem is as follows:

$$\text{Minimize } f(X) = \max\{\varphi_1(X), \varphi_2(X), \dots, \varphi_{2m}(X)\} \quad (37)$$

Where $X = \{(x_1, x_2, x_3, \dots, x_D) \in R^D \mid 0 \leq x_j \leq 2\pi, j = 1, 2, 3, \dots, D\}$ and $m = 2D - 1$, with

$$\phi_{2i-1}(X) = \sum_{j=i}^D \cos\left(\sum_{k=|2i-j|+1}^j X_k\right), i = 1, 2, 3, \dots, D \quad (38)$$

$$\phi_{2i}(X) = 0.5 + \sum_{j=i+1}^D \cos\left(\sum_{k=|2i-j|+1}^j X_k\right), i = 1, 2, 3, \dots, D-1 \quad (39)$$

$$\phi_{m+i}(X) = -\phi_i(X), i = 1, 2, 3, \dots, m \quad (40)$$

Here the goal is to optimized (minimize) the module of the greatest among the autocorrelation function that is associated with the compressed radar pulse at the optimal receiver output whereas the variables represent symmetrized phase

Table 14 Parameter Estimation for Frequency-Modulated (FM) Sound Waves

Algorithm	Best	Medians	Worst	Mean	SD
mISOS	1.97e+001	2.60e+001	2.90e+001	2.58e+001	2.06e+000
ISOS	2.41e+001	2.72e+001	2.92e+001	2.71e+001	1.28e+000
SOS	2.44e+001	2.62e+001	2.79e+001	2.64e+001	9.89e-001
ASOS_BF1	2.39e+001	2.64e+001	2.78e+001	2.63e+001	1.11e+000
ASOS_BF2	1.95e+001	2.64e+001	2.82e+001	2.61e+001	1.93e+000
ASOS_BF1&2	2.15e+001	2.61e+001	2.85e+001	2.61e+001	1.41e+000
BSA	2.37e+001	2.82e+001	2.99e+001	2.79e+001	1.45e+000
ABSA	2.56e+001	2.83e+001	3.05e+001	2.82e+001	1.42e+000
DE	2.54e+001	2.65e+001	2.87e+001	2.65e+001	8.92e-001

differences. It is an NP-hard optimization problem. The problem can be categorized as a class of continuous min-max global optimization problems because of the objective function is piecewise smooth. The problem has twenty variables and the domain of the variable is $[0, 2\pi]$.

5.5 Result and discussion

The mISOS solved the above problems and the optimum solution is described in Tables 14, 15, 16, and 17. The function evaluations are 1000 function evaluations with eco-size 50 and 25 runs used to solve the problem.

For Parameter estimation for frequency modulated (FM) sound waves problem, the obtained results are compared with ISOS, SOS, ASOS_BF1, ASOS_BF2, ASOS_BF1&2, BSA, ABSA, and DE. The performance results of this problem are presented in Table 14, which includes the Best, Median, Worst, Mean, and S.D. It is seen from Table 14 that the best solution is obtained by ASOS_BF2, Worst solution obtained by ABSA but Medians and Mean results are obtained by the mISOS algorithm. So from the results presented in Table 14, one can easily summarize that the performance of the mISOS algorithm on this optimization problem is the best minimized compared to other algorithms.

For Lennard-Jones Potential Problem, the obtained results are also compared with ISOS, SOS, ASOS_BF1, ASOS_BF2,

ASOS_BF1&2, BSA, ABSA, and DE. The performance results of this problem are presented in Table 15, which includes the Best, Median, Worst, Mean, and S.D. It is seen from Table 15 that the best, medians and mean results are obtained by mISOS. So from the results presented in Table 15, one can easily summarize that the performance of the mISOS algorithm on this optimization problem is the best minimized compared to other algorithms.

For Tersoff potential for model Si(B), the obtained results are compared with various other algorithms such as ISOS, SOS, ASOS_BF1, ASOS_BF2, ASOS_BF1&2, BSA, ABSA, and DE. The performance results of this problem are presented in Table 16, which includes the Best, Median, Worst, Mean, and S.D. It is seen from Table 16 that the best solution is obtained by BSA, the worst solution is obtained by ISOS but medians and mean results are obtained by mISOS. From the reported results, we observe that analyzed the mISOS variant has shown a better solution compared to SOS and other algorithms.

For the spread spectrum radar poly phase problem, the obtained results are compared with ISOS, SOS, ASOS_BF1, ASOS_BF2, ASOS_BF1&2, BSA, ABSA, and DE. It is seen from Table 17 that the best solution is obtained by BSA, the worst solution is obtained by SOS but medians and mean results are obtained by mISOS. So from the observation of Table 17, one can easily show that the mISOS algorithm

Table 15 Lennard-Jones Potential Problem

Algorithm	Best	Medians	Worst	Mean	SD
mISOS	-6.04e+000	-4.23e+000	-2.93e+000	-4.37e+000	7.76e-001
ISOS	-6.28e+000	-3.40e+000	-2.11e+000	-3.48e+000	9.39e-001
SOS	-6.42e+000	-4.03e+000	-2.47e+000	-4.12e+000	8.74e-001
ASOS_BF1	-4.41e+000	-3.29e+000	-2.43e+000	-3.29e+000	5.40e-001
ASOS_BF2	-5.42e+000	-3.31e+000	-2.30e+000	-3.42e+000	7.56e-001
ASOS_BF1&2	-3.46e+000	-2.56e+000	-2.04e+000	-2.59e+000	3.53e-001
BSA	-4.37e+000	-2.51e+000	-2.03e+000	-2.71e+000	5.81e-001
ABSA	-4.37e+000	-2.51e+000	-2.03e+000	-2.71e+000	5.81e-001
DE	-3.91e+000	-2.78e+000	-2.14e+000	-2.84e+000	4.28e-001

Table 16 Tersoff Potential Function Minimization Problem

Algorithm	Best	Medians	Worst	Mean	SD
mISOS	-1.93e+001	-1.36e+001	-1.02e+001	-1.40e+001	2.22e+000
ISOS	-1.59e+001	-1.21e+001	-8.77e+000	-1.28e+001	1.70e+000
SOS	-1.48e+001	-1.19e+001	-8.56e+000	-1.18e+001	1.69e+000
ASOS_BF1	-1.77e+001	-1.16e+001	-4.66e+000	-1.15e+001	2.48e+000
ASOS_BF2	-1.71e+001	-1.12e+001	-8.39e+000	-1.15e+001	1.92e+000
ASOS_BF1&2	-1.60e+001	-1.11e+001	-8.35e+000	-1.14e+001	1.89e+000
BSA	-2.03e+001	-1.32e+001	-1.01e+001	-1.33e+001	2.09e+000
ABSA	-1.85e+001	-1.35e+001	-1.10e+001	-1.37e+001	1.73e+000
DE					

performs better compared to other algorithms. This indicates the superiority of mISOS in solving this problem in approximating the global. Thus the above discussion concludes that the performance of the SaISOS algorithm in solving this problem is acceptable.

From the above discussion, results, and evaluations, it is clear that mISOS has improved quality, efficiency, and robustness to solve real-life problems. Also, mISOS may be reflected as an additional optimizer for real-life problems.

6 Conclusions and future directions

A modified version of SOS called modification based I-SOS (called mISOS) algorithm is established to astounded the weaknesses of the SOS algorithm. The three altered search approaches called adjusted benefit factor, altered parasitism phase, and a random weighted number-based search are engaged in the suggested mISOS algorithm. An altered parasite vector is suggested here for advancement in the exploitability of the parasitism stage in the fundamental SOS algorithm. To improve the effectiveness of complex structures of basic SOS, an adaptive benefit factor is introduced. The random weight number is used to travel the current

form of the organism into other more promising space for the search. The suggested mISOS have been executed on twenty test functions. The strength of the mISOS has been tested by considering dimension 100. The results obtained by mISOS algorithm is compared with a various existing algorithm such as SOS, I-SOS, SOS-ABF1, SOS-ABF2, SOS-ABF1&2 BSA, ABSA, CLPSO, CPSO-H, FDR-PSO, FI-PS, UPSO, EPSDE, TSDE, CPI-DE, ACoS-PSO and HBSA. The examination of the outcomes through numerically, statistically and convergence behavior guarantees the prevalent exhibition of the mISOS on a large number of variable test problems when contrasted with basic SOS, some improve version of SOS, and some other optimizer. The execution comparison on a few designing engineering test problems illustrates the appropriateness of the mISOS to real-world engineering applications.

The analysis of the results through convergence behavior and statistical validation ensures the superior performance of the mISOS on high-dimensional optimization problems as compared to conventional SOS, variants of SOS, and some other recent optimization algorithms. The performance comparison on several engineering optimization test cases demonstrates the applicability of the mISOS on real-world problems.

Table 17 Spread spectrum radar polyphase problem

Algorithm	Best	Medians	Worst	Mean	SD
mISOS	1.76e+000	2.24e+000	2.59e+000	2.25e+000	2.27e-001
ISOS	2.17e+000	2.39e+000	2.59e+000	2.39e+000	1.19e-001
SOS	2.07e+000	2.36e+000	2.65e+000	2.36e+000	1.63e-001
ASOS_BF1	2.04e+000	2.33e+000	2.59e+000	2.31e+000	1.44e-001
ASOS_BF2	2.02e+000	2.34e+000	2.54e+000	2.32e+000	1.49e-001
ASOS_BF1&2	1.90e+000	2.33e+000	2.66e+000	2.35e+000	1.70e-001
BSA	1.81e+000	2.33e+000	2.58e+000	2.27e+000	2.16e-001
ABSA	1.90e+000	2.32e+000	2.60e+000	2.30e+000	1.64e-001
DE	2.01e+000	2.33e+000	2.59e+000	2.31e+000	1.63e-001

As the suggested mISOS illustrate its productivity and un-wavering quality in solving the problem with large number of variable, it can subsequently be utilized within the future to execute complex non-linear problem. Later on, the mISOS can likewise be extended by some fundamental alterations to solve the discrete, constrain, and multi-objective test problem.

Compliance with ethical standards

Conflict of interest The author declares that he has no conflict of interest regarding the publication of this paper.

Nomenclature SOS, Symbiotic Organisms Search; mISOS, Modification Based Improved Symbiotic Organisms Search; GA, Genetic Algorithm; PSO, Particle Swarm Optimization; DE, Differential Evolution; BBO, Biogeography-Based Optimization; HS, Harmony Search; GSA, Gravitational Search Algorithm; WCA, Water Cycle Algorithm; BSA, Backtracking Search Optimization Algorithm; EA, Evolutionary Algorithm; SOS-VNS, Hybrid Symbiotic Organisms Search Algorithm With Variable Neighbourhood Search; ATSP, Asymmetric Traveling Salesman Problem; VNS, Variable Neighbourhood Search; A-CSOS, Competitive Ranking-Based Symbiotic Organisms Search Algorithm; ORPD, Optimal Reactive Power Dispatch; ISOS, Improved Symbiotic Organisms Search Algorithm; SASOS, Simulated Annealing Based Symbiotic Organism Search; DOCR, Directional Overcurrent Relay (DOCR) Problems; MQSOS, Symbiotic Organism Search Algorithm With Multi-Group Quantum-Behavior Communication Scheme; OSOS, Oppositional Symbiotic Organisms Search Optimization; ESOS, Enhanced Symbiotic Organisms Search Algorithm; CSOS, Complex-Valued Encoding Symbiotic Organisms Search Algorithm; MSOS, Modified Symbiotic Organisms Search; QOBL, Quasi-Opposition-Based Learning; CLS, Chaotic Local Search; QOCSOS, Quasi-Oppositional-Chaotic Symbiotic Organisms Search Algorithm; SQI, Simple Quadratic Interpolation; HSOS, Hybrid Symbiosis Organisms Search; BF1 and BF2, Benefit Factors; SOS-ABF1,2,1&2, Adaptive Symbiotic Organisms Search; I-SOS, Improved Symbiotic Organisms Search Algorithm; ABSA, Adaptive Backtracking Search Algorithm; CLPSO, Comprehensive Learning Particle Swarm Optimizer; CPSO-H, Cooperative Approach To Particle Swarm Optimization; FDR-PSO, Fitness-Distance-Ratio Based Particle Swarm Optimization; FI-PS, Fully Informed Particle Swarm; UPSO, A Unified Particle Swarm Optimization Scheme; EPSDE, Differential Evolution Algorithm with Ensemble Of Parameters And Mutation Strategies; TSDE, Differential Evolution with A Two-Stage Optimization Mechanism; CPI-DE, Cumulative Population Distribution Information In Differential Evolution; ACOS-PSO, An Adaptive Framework To Tune The Coordinate Systems In Particle Swarm Optimization Algorithms; HBSA, Hybrid Backtracking Search Optimization Algorithm

References

- Rao SS (2019) Engineering optimization: theory and practice. John Wiley & Sons
- Wolpert DH, Macready WG (1997) No free lunch theorems for optimization. *IEEE Trans Evol Comput* 1(1):67–82
- Holland JH (1992). *Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence*. MIT press
- Shi Y, Eberhart R (1998) A modified particle swarm optimizer. In 1998 IEEE international conference on evolutionary computation proceedings. IEEE world congress on computational intelligence (cat. No. 98TH8360) (pp 69-73). IEEE
- Storn R, Price K (1997) Differential evolution—a simple and efficient heuristic for global optimization over continuous spaces. *J Glob Optim* 11(4):341–359
- Simon D (2008) Biogeography-based optimization. *IEEE Trans Evol Comput* 12(6):702–713
- Lee KS, Geem ZW (2005) A new meta-heuristic algorithm for continuous engineering optimization: harmony search theory and practice. *Comput Methods Appl Mech Eng* 194(36–38):3902–3933
- Rashedi E, Nezamabadi-Pour H, Saryazdi S (2009) GSA: a gravitational search algorithm. *Inf Sci* 179(13):2232–2248
- Eskandar H, Sadollah A, Bahreininejad A, Hamdi M (2012) Water cycle algorithm—a novel metaheuristic optimization method for solving constrained engineering optimization problems. *Comput Struct* 110:151–166
- Civicioglu P (2013) Backtracking search optimization algorithm for numerical optimization problems. *Appl Math Comput* 219(15): 8121–8144
- Cheng MY, Prayogo D (2014) Symbiotic organisms search: a new metaheuristic optimization algorithm. *Comput Struct* 139:98–112
- Umam MIH, Santosa B (2018) A hybrid symbiotic organisms search algorithm with variable neighbourhood search for solving symmetric and asymmetric traveling salesman problem. In: IOP conference series: materials science and engineering, vol 337 no 1. IOP publishing, p 012005
- Yalçın E, Çam E, Taplamacıoğlu MC (2019) A new chaos and global competitive ranking-based symbiotic organisms search algorithm for solving reactive power dispatch problem with discrete and continuous control variable. *Electric Eng*:1-18
- Çelik E (2020) A powerful variant of symbiotic organisms search algorithm for global optimization. *Eng Appl Artif Intell* 87:103294
- Sönmez Y, Unal M (2020) Estimation of smooth and non-smooth fuel cost function parameters using improved symbiotic organisms search algorithm. *J Electric Eng Technol* 15(1):13–25
- Chu SC, Du ZG, Pan JS (2020) Symbiotic organism search algorithm with multi-group quantum-behavior communication scheme applied in wireless sensor networks. *Appl Sci* 10(3):930
- Chakraborty F, Nandi D, Roy PK (2019) Oppositional symbiotic organisms search optimization for multilevel thresholding of color image. *Appl Soft Comput* 82:105577
- Ezugwu AE (2019) Enhanced symbiotic organisms search algorithm for unrelated parallel machines manufacturing scheduling with setup times. *Knowl-Based Syst* 172:15–32
- Miao F, Zhou Y, Luo Q (2019) Complex-valued encoding symbiotic organisms search algorithm for global optimization. *Knowl Inf Syst* 58(1):209–248
- Kumar S, Tejani GG, Mirjalili S (2019) Modified symbiotic organisms search for structural optimization. *Eng Comput* 35(4):1269–1296
- Liu D, Li H, Wang H, Qi C, Rose T (2020) Discrete symbiotic organisms search method for solving large-scale time-cost trade-off problem in construction scheduling. *Expert Syst Appl* 148: 113230
- Truong KH, Nallagownden P, Baharudin Z, Vo DN (2019) A quasi-oppositional-chaotic symbiotic organisms search algorithm for global optimization problems. *Appl Soft Comput* 77:567–583
- Nama S, Saha A, Ghosh S (2016) Improved symbiotic organisms search algorithm for solving unconstrained function optimization. *Decision Sci Lett* 5(3):361–380

24. Nama S, Saha AK, Ghosh S (2017) A hybrid symbiosis organisms search algorithm and its application to real world problems. *Memetic Comput* 9(3):261–280
25. Saha A, Nama S, Ghosh S (2019) Application of HSOS algorithm on pseudo-dynamic bearing capacity of shallow strip footing along with numerical analysis. *Int J Geotech Eng*:1–14
26. Nama S, Saha AK, Sharma S (2020) A novel improved symbiotic organisms search algorithm. *Comput Intell*. <https://doi.org/10.1111/coin.12290>
27. Tejani GG, Savsani VJ, Patel VK (2016) Adaptive symbiotic organisms search (SOS) algorithm for structural design optimization. *J Comput Design Eng* 3(3):226–249
28. Prayogo D, Cheng MY, Wong FT, Tjandra D, Tran DH (2018) Optimization model for construction project resource leveling using a novel modified symbiotic organisms search. *Asian J Civil Eng* 19(5):625–638
29. Satapathy S, Naik A (2013) Improved teaching learning based optimization for global function optimization. *Decision Sci Lett* 2(1):23–34
30. Nama S, Saha A (2019) A novel hybrid backtracking search optimization algorithm for continuous function optimization. *Decision Sci Lett* 8(2):163–174
31. Duan H, Luo Q (2014) Adaptive backtracking search algorithm for induction magnetometer optimization. *IEEE Trans Magn* 50(12):1–6
32. Liang JJ, Qin AK, Suganthan PN, Baskar S (2006) Comprehensive learning particle swarm optimizer for global optimization of multimodal functions. *IEEE Trans Evol Comput* 10(3):281–295
33. Van den Bergh F, Engelbrecht AP (2004) A cooperative approach to particle swarm optimization. *IEEE Trans Evol Comput* 8(3):225–239
34. Peram T, Veeramachaneni K, Mohan CK (2003) Fitness-distance-ratio based particle swarm optimization. In proceedings of the 2003 IEEE swarm intelligence symposium. SIS'03 (cat. No. 03EX706). IEEE, pp 174–181
35. Mendes R, Kennedy J, Neves J (2004) The fully informed particle swarm: simpler, maybe better. *IEEE Trans Evol Comput* 8(3):204–210
36. Parsopoulos KE (2004) UPSO: a unified particle swarm optimization scheme. *Lecture Series Comput Comput Sci* 1: 868–873
37. Mallipeddi R, Suganthan PN, Pan QK, Tasgetiren MF (2011) Differential evolution algorithm with ensemble of parameters and mutation strategies. *Appl Soft Comput* 11(2):1679–1696
38. Liu ZZ, Wang Y, Yang S, Cai Z (2016) Differential evolution with a two-stage optimization mechanism for numerical optimization. In 2016 IEEE congress on evolutionary computation (CEC). IEEE, pp 3170–3177
39. Wang Y, Liu ZZ, Li J, Li HX, Yen GG (2016) Utilizing cumulative population distribution information in differential evolution. *Appl Soft Comput* 48:329–346
40. Liu ZZ, Wang Y, Yang S, Tang K (2018) An adaptive framework to tune the coordinate systems in nature-inspired optimization algorithms. *IEEE Trans Cybernet* 49(4):1403–1416
41. Derrac J, García S, Molina D, Herrera F (2011) A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. *Swarm Evolution Comput* 1(1):3–18
42. Demšar J (2006) Statistical comparisons of classifiers over multiple data sets. *J Mach Learn Res* 7(Jan):1–30
43. Das S, Suganthan PN (2010) Problem definitions and evaluation criteria for CEC 2011 competition on testing evolutionary algorithms on real world optimization problems. Jadavpur University, Nanyang Technological University, Kolkata, pp 341–359
44. Niu B, Liu Y, Zhou W, Li H, Duan P, Li J (2019a) Multiple Lyapunov functions for adaptive neural tracking control of switched nonlinear nonlower-triangular systems. *IEEE Transactions on Cybernetics*
45. Niu B, Wang D, Liu M, Song X, Wang H, Duan P (2019b) Adaptive neural output-feedback controller Design of Switched Nonlower Triangular Nonlinear Systems with Time Delays. *IEEE Trans Neural Netw Learn Syst*
46. Niu B, Wang D, Alotaibi ND, Alsaadi FE (2018) Adaptive neural state-feedback tracking control of stochastic nonlinear switched systems: an average dwell-time method. *IEEE Trans Neural Netw Learn Syst* 30(4):1076–1087
47. Kahraman HT, Aras S, Gedikli E (2020) Fitness-distance balance (FDB): a new selection method for meta-heuristic search algorithms. *Knowl-Based Syst* 190:105169

Publisher's note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Sukanta Nama received his M.Sc. and Ph.D. degree in Mathematics from the National Institute of Technology Agartala, India in 2013 and 2018 respectively. Currently, he is working as a guest teacher at the Department of Applied Mathematics, MBB University, Agartala. His current research interest area includes operation research, evolutionary optimization, and application of evolutionary optimization in different fields such as geotechnical problems, image segmentation.