RESEARCH PAPER

Hybridizing salp swarm algorithm with particle swarm optimization algorithm for recent optimization functions

Narinder Singh[1] · S. B. Singh[1] · Essam H. Houssein[2]

Abstract
The salp swarm algorithm (SSA) has shown its fast search speed in several challenging problems. Research shows that not every nature-inspired approach is suitable for all applications and functions. Additionally, it does not provide the best exploration and exploitation for each function during the search process. Therefore, there were several researches attempts to improve the exploration and exploitation of the meta-heuristics by developing the newly hybrid approaches. This inspired our current research and therefore, we developed a newly hybrid approach called hybrid salp swarm algorithm with particle swarm optimization for searching the superior quality of optimal solutions of the standard and engineering functions. The hybrid variant integrates the advantages of SSA and PSO to eliminate many disadvantages such as the trapping in local optima and the unbalanced exploitation. We have used the velocity phase of the PSO approach in salp swarm approach in order to avoid the premature convergence of the optimal solutions in the search space, escape from ignoring in local minima and improve the exploitation tendencies. The new approach has been verified on different dimensions of the given functions. Additionally, the proposed technique has been compared with a wide range of algorithms in order to confirm its efficiency in solving standard CEC 2005, CEC 2017 test suits and engineering problems. The simulation results show that the proposed hybrid approach provides competitive, often superior results as compared to other existing algorithms in the research community.

Keywords Standard functions · Heuristic hybridization · Salp swarm algorithm · Particle swarm optimization algorithm · Exploration and exploitation

1 Introduction

The last few decades witnessed the introduction of many robust population-based meta-heuristics, such as swarm intelligence optimization and evolutionary algorithms for the purpose of finding the best and possible optimal solutions to many real-life applications. Although such algorithms have been useful in tackling many real life problems, the extensive literature review reveals that there is no single algorithm which works well in all the applications. Consequently, researchers have been developing newly, modified and hybrid techniques for resolving and eliminate many of the disadvantages of the existing algorithms such as: differential evolution (DE) [1, 2], genetic algorithm (GA) [3, 4], hybrid genetic algorithm (HGA) [5], biogeography based optimization algorithm (BBO) [6], gravitational search algorithm (GSA) [7], particle swarm optimization (PSO) [8], Tabu search (TS) [9], harmony search algorithm (HAS) [10], artificial neural network (ANN) [11], dragonfly algorithm (DA) [12], grey wolf optimization (GWO) [13], krill herd algorithm (KHA) [14], black-hole-based optimization (BHBO) [15], robust optimization (RO) [16], ant lion optimizer (ALO) [17], One half personal best position particle swarm optimizations (OHGBPPSO) [18], EPO [19], SHO [20], EGWO [21], AGWO [22], PSOGWO [23], MFOA [79], RSO [80] and many other algorithms [24–38].

The particle swarm optimization is a swarm intelligence approach, which is inspired by the social behaviors of animals like bird flocking and fish schooling [39, 40]. As a

✉ Narinder Singh
narindersinghgoria@gmail.com

S. B. Singh
sbsingh69@yahoo.com

Essam H. Houssein
essam.halim@mu.edu.eg

1 Department of Mathematics, Punjabi University, Patiala, India

2 Faculty of Computers and Information, Minia University, El-Minia Governorate, Egypt

stochastic search method, particle swarm optimization algorithm has characters of rapid convergence capability and simple computation. The approach was applied in separate industrial problems like modeling [41–43], prediction [41], parameters learning of neural network [44, 45], power systems [46–50], control [51], and parameters optimization of fuzzy system [52].

Over the years, researchers developed several hybrid techniques to enhanced the balanced between exploration and exploitation of the meta-heuristics and searching the superior quality of the solutions of the real life applications. For instance, Ali and Tawhid [53] developed a hybrid approach by integrating the PSO and GA to minimize a simplified model of the energy problem of the molecule. The advantages of existing approach have been presented by using the three different mechanisms. The accuracy of the approach has been verified on the potential energy function and thirteen unconstrained global optimization functions. Simulated results reveal that the newly existing method is an efficient and promising approach. Similarly, Yu et al. [37] presented a hybrid approach called HPSO. In their approach, they integrated space transformation search with a new modified velocity model. Simulation solutions on eight standard problems demonstrated that the hybrid particle swarm optimization holds superior accuracy in solving standard functions. Additionally, Mao et al. [54] proposed hybrid method integrated with a DE and PSO known as DEMPSO, to find the best and possible global optimal solution of the nonlinear model of the forward kinematics. Five configurations with distinct positions and orientations were used as an example to demonstrate the efficiency of the existing variant for finding the best global optimal solution of kinematic function of parallel manipulators. And the experimental solution of existing approach and the other meta-heuristics also revealed that newly hybrid approach can provide a superior performance regarding global searching properties and the convergence rate.

The several real life applications best optimal solutions has been finds by the researchers of the different fields such as environmental/economic dispatch [55], cost-based feature selection in classification [56], feature selection on high-dimensional data [57], dynamic economic dispatch with valve-point effect [58], robot path planning in uncertain environment [59] and many others.

In our previous research work, we presented several modified and hybrid nature-inspired approaches and these algorithms were applied on several real life applications and tried to improved and extend the working strength of the newly existing meta-heuristics [18, 31, 33, 35, 36, 23, 60]. In this research, we extend our previous work via proposing a new hybrid algorithm employing SSA and PSO to benefit from their exploitation and exploration. Furthermore, the proposed method is to aggregate advantages from SSA

and PSO with an integrated feedback mechanism to achieve a better optima goal (or solution). In fact, PSO is used to enhance the exploitation and exploration of SSA. As a result, the newly existing method has the balance ability amid local and global searching abilities to guarantee the better convergence. So, the major contributions of the research work can be explained as:

- The proposed methodology is enhancing the balanced between exploration and exploitation by feed-back mechanism.
- Search tendency is redefined to improve the capability of searching best optimal solution in the search space during the search process.
- Under this methodology, during the search process the proposed strategy is fastly trapping the global optima solution in the search space.

The experimental solutions reveal that the newly existing approach possess superior capability of search the global optimum than that of the other meta-heuristics. The performance of the existing approach is demonstrated by the obtained results of standard CEC 2005, CEC 2017 test suits and real life functions with comparison to others.

The rest of the article is structured as follows. Section 2, describes related research which describes the main aspects of the salp swarm algorithm and particle swarm optimization algorithm; Sect. 3, presents the hybrid SSA–PSO algorithms; the numerical and statistical experiments are described in Sect. 4; in Sect. 5, experiment and results are performed. The engineering applications are described in Sects. 6 and 7; Sect. 7 provides the conclusions.

2 Related work

In this section, we overview existing research showing different aspects of the two main algorithms have been employed in this research: salp swarm algorithm and particle swarm optimization algorithm. This is done via presenting existing studies from the literature and showing their relatedness to the current approach.

In 2017, Mirjalili et al. [61] developed a nature inspired approach, known as salp swarm algorithm that mimics the special swarming behavior of salps in oceans. Salps usually live in groups and often form a swarm called salp chain. The first salp is denoted as the leader, while the others are followers. The position of the leader should by using the following mathematical equations:

$$x_j^1 = \begin{cases} F_j + c_1\left(\left(ub_j - lb_j\right)c_2 + lb_j\right) & c_3 \geq 0.5 \\ F_j - c_1\left(\left(ub_j - lb_j\right)c_2 + lb_j\right) & c_3 < 0.5 \end{cases} \tag{1}$$

where x_j^1 is the position of the best and possible solution, ub_j, lb_j is the upper and lower bound of the jth dimension, F_j is the food source position of the dimension, c_2, c_3 are two random numbers in the range [0, 1] and c_1 is an most important variable in this algorithm that gradually decreases over the course of generations to allow high exploration at the early stages of the optimization process, then high exploitation in last steps.

$$c_1 = 2e^{-\left(\frac{4l}{L}\right)^2} \tag{2}$$

where L represents the maximum number of iteration and l indicates the current iterations. The follower's positions are updated by using the following mathematical equation:

$$x_j^i = \frac{1}{2}\left(x_j^i + x_j^{i-1}\right) \tag{3}$$

where x_j^i indicates the position of the ith follower at the jth dimension and $i \geq 2$.

Besides the SSP algorithm, the original version of PSO approach was proposed by Kennedy et al. [39]. Their algorithm emulates the flocking behavior of particles or birds to solve global optimization functions. In this approach, each possible result is considered as an agent (particle). All candidates have their own velocities and fitness values. These agents fly through the D-dimensional function area by learning from the historical information of the agents. A possible result is demonstrated by an agent, which adjusts its velocity and position according to mathematical Eqs. (4) and (5):

$$v_i^{k+1} = w \times v_i^k + c_4 r_1\left(p_i^k - x_i^k\right) + c_5 r_2\left(g_{best} - x_i^k\right) \tag{4}$$

$$x_i^{k+1} = x_i^k + v_i^{k+1} \tag{5}$$

where v_{ij}^k indicates the velocity of particle $i \in [1, 2, \ldots, n_x]$ in dimension j at time step k, w is the inertia constant,

---

Algorithm 1. Pseudocode of the Salp Swarm Algorithm

> Initialize population x_i i=1,2,...,n
>
> for generation from 1 to $\mathrm{max\_iter}$
>
> do
> Calculate all salp in the crowd
> Represent the best salp as F
>
> Update the value of c_1 by using the mathematical equation (2)
>
> for (all salp x_i)do
>
> if x_i is a leader
>
> then
> Update the position of the leader by using the mathematical equation (1)
> else
> Update the position of the follower's by using the mathematical equation (3)
> Return F

---

p_i^k is the personal best position of the agent, g_{best} is the best position of the neighbor agent, x_{ij}^k represents the position of the particle i in dimension j at time step k.

The constants w and c_4, c_5 are acceleration coefficients and rand values $r_{1j}^k, r_{2j}^k \in U(0, 1)$ in the range $[0, 1]$ at a time step k.

Algorithm 2. Pseudocode of the PSO (Particle Swarm Optimization) Algorithm

For each particle

Initialize particle x_i $i=1,2,...,n$

End
Do
For all particle
Calculate fitness value

If the fitness value is superior than the best fitness value (p_i^k) in history

Set current value as the new (p_i^k)

End

Choose the particle with the best fitness value of all the particles as the (g_{best})

For each particle
Calculate particle velocity and position according to equations (4) and (5)
End
While minimum error or maximum iterations criteria is not attained.

---

3 Hybrid SSA–PSO algorithm

Each algorithm is not able to solving each type of real-world problems. These algorithms may be failure to find the goal for any complex functions due to own weakness in the search area. Therefore, according to demand of the present situations of the real-world applications the researchers of the different fields are developing the newly modified and hybrid version for reducing the weakness of the present versions that these algorithms could be able to gives the best global optimal solution for the complex applications. After learning these strategies, we have tried to develop a new hybrid algorithm in this text for complex functions.

Although, the SSA method is more able to reveal a capable accuracy in contrast with other recent meta-heuristics, one of its limitations is that it might face the complexity of getting trapped in local/global optima. Additionally, it is not fitting for high difficult problems or functions and cannot handle their drawbacks such as slow convergence speed, slow diversity and premature convergence. To further boost the exploration and exploitation of SSA, a more focused particle swarm optimizer is incorporated into this algorithm to form a newly hybrid SSA–PSO algorithm. In order to reduce these types of weaknesses and to improve the SSA method

search capability, the merits of the two different methods have been merged and developed as an newly approach that we called the HSSAPSO algorithm. The present method has been applied to identify the best score of the complex optimization functions. The particle swarm optimization algorithm phase operates in the direction of exploring the vector of optimal solutions although the HSSAPSO approach is invoked as a local search method to enhance the optimal solution superiority. This methodology is also helpful in fastly trapping the global optimal solution and ignoring the local optima in the search area during the search process. Therefore, with this method we can strengthen the search ability and obtain accurate convergences by accelerating the search. Further, details of Hybrid SSAPSO algorithm is shown step by step as below:

3.1 Step 1: parameter setting

The newly existing hybrid approach starts by setting its constant values such as the number of search agents ($s = 30$), current iteration ($l = 2$), maximum iterations ($L = 50 - 500$), maximum weight ($w_{max} = 0.9$), minimum weight ($w_{min} = 0.2$) and maximum velocity ($v_{max} = 6$).

Fig. 1 Flowchart of Hybrid
SSAPSO algorithm
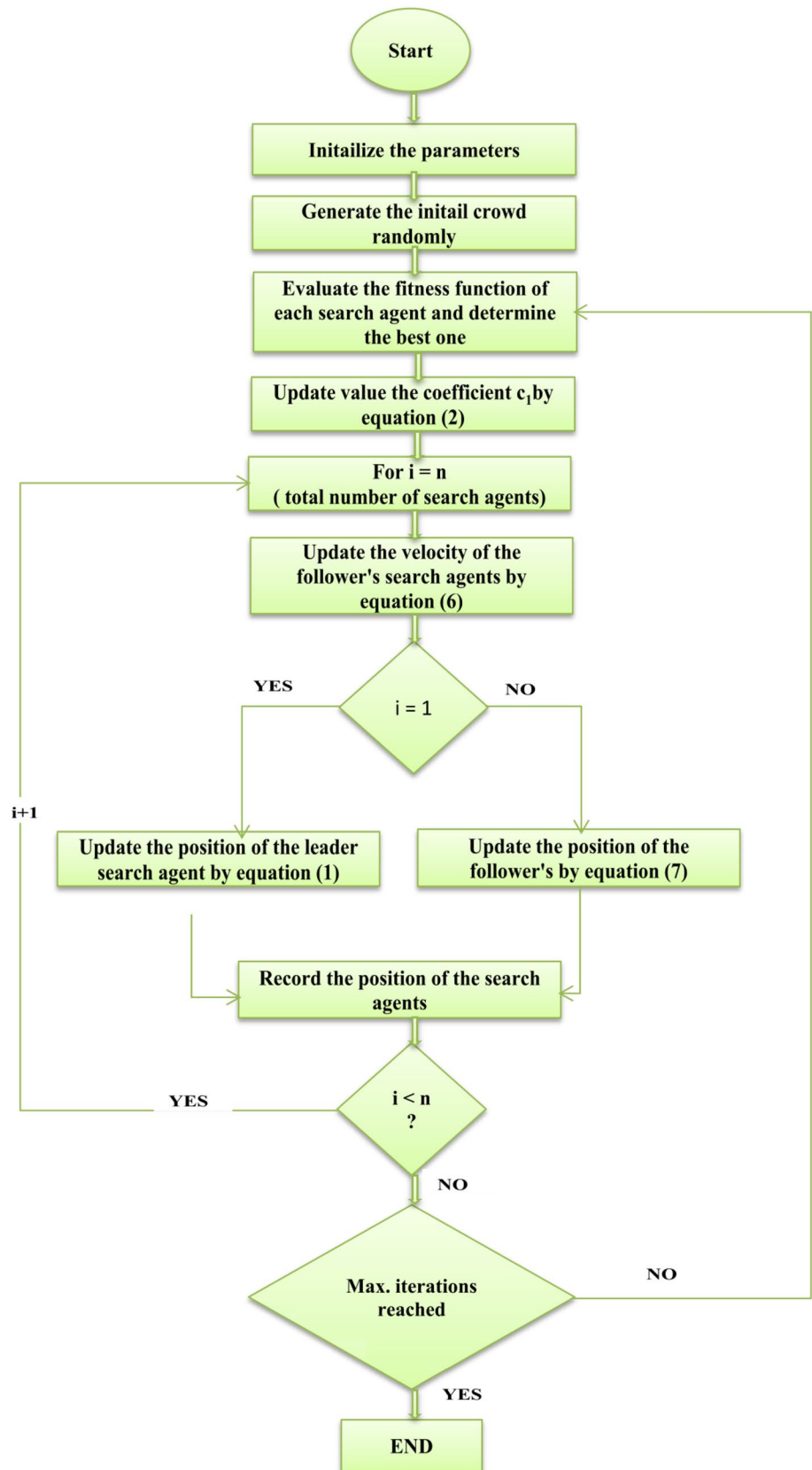
3.2 Step 1: initialization

During the search process of the algorithm in the search space firstly we are initializing the crowd and these are initialized randomly according of the given problems, where the algorithms assigns a random vector of n dimensional for the ith salp; $X = x_i \sim (i = 1, 2, 3, \ldots, n)$.

3.3 Step 2: evaluation

Under this step, during the search process the each search agent of the crowd is evaluated as per the superiority of the own location. The fitness value of each search agent is tested on the basis of objective function after that, the each search agents take the next new location according to its fitness value in the search space.

3.4 Step 3: leader position updating

The position of the main search agent (like; leader) has been updating by mathematical Eqs. (1)–(2) during the search process in the search space.

3.5 Step 4: velocity initialization

We easily see in the literature the velocity initialization is a most important role for the population-based algorithms. With the help of this concept we can reduced the search agent's efforts during the search process in the search space and ignored wrong positions. Many times, during the search process of the search agents for searching the global optima goal, these search agents may leave the intended boundaries of the search area, which will tend the wastage of finding effort and can a bad impact on the accuracy of the algorithm, hence this concept is play a important role for trapping the best global optima solution and ignoring the local optima in the search space. The velocity initialization can be done in

three different ways such as initialize to small random value, initialize to random values close to zero and lastly initialize to zero. Distinct strategies of this initialization have different impacts on the accuracy of the algorithm.

3.6 Step 5: follower's position updating

The position of the follower's in the search space during the search process has been replaced by the help of modified mathematical Eq. (6). Here the velocity concept plays a most important role for trapping the global optima fastly and ignored the wrong positions during the search process.

3.7 Step 5: stopping condition

At the end, the stopping condition has been applied for searching the global optima for all kind of problems. Here this criteria is used for evaluating each search agent assessment process and replacing the best search agent's position, it is repeated again and again until it satisfies the criteria of prevention for example it reaches the highest maximum iterations or the solution is earliest found. The rest of operations are the same as salp swarm algorithm. Furthermore, the working process of the proposed algorithm has been discussed in the flowchart Fig. 1.

The main structure of the newly existing approach HSSAPSO is presented in Algorithm 3.

Table 1 Parameter settings

Constants	Definitions	Values
s	Number of search agents	30
l	Current iteration	2
L	Maximum number of iterations	50–500
w_{max}	Max-Inertia value	0.9
w_{min}	Min-Inertia value	0.2
v_{max}	Maximum velocity	6

Table 2 Unimodal benchmark functions

Function	Dim	Range	f_{min}
$F_1(y) \sim \sum_{i=1}^{n} y_i^2$	30	$[-100, 100]$	0
$F_2(y) \sim \sum_{i=1}^{n} \lvert y_i \rvert + \prod_{i=1}^{n} \lvert y_i \rvert$	30	$[-10, 10]$	0
$F_3(y) \sim \sum_{i=1}^{n} (\sum_{j-1}^{i} y_j)^2$	30	$[-100, 100]$	0
$F_4(y) \sim \max_i \{ \lvert y_i \rvert, \ 1 \le i \le n \}$	30	$[-100, 100]$	0
$F_5(y) \sim \sum_{i=1}^{n-1} \left[100(y_{i+1} - y_i^2)^2 + (y_i - 1)^2 \right]$	30	$[-30, 30]$	0
$F_6(y) \sim \sum_{i=1}^{n} \left([y_i + 0.5] \right)^2$	30	$[-100, 100]$	0
$F_7(y) \sim \sum_{i=1}^{n} iy_i^4 + rand[0, 1)$	30	$[-1.28, 1.28]$	0

Algorithm 3. Pseudocode of the Hybrid Salp Swarm Algorithm with Particle Swarm Optimization Algorithm (HSSAPSO)

Set the initial constant;

Number of search agents $s = 30$, $l = 2$, $L = 50 - 500$, $w_{max} = 0.9$, $w_{min} = 0.2$ and

$v_{max} = 6$

Initialize population $x_i (i = 1, 2, ..., n)$

for generation from 1 to max_iter

do
Calculate all salp in the crowd
Represent the best salp as F

Update the value of c_1 by using the mathematical equation (2)

for (all salp (x_i)) do

Calculate the velocity of the follower's by using the equation (6):

$$v_i^{k+1} = w \times v_i^k + c_1 \times \left(salp_j^{fitness} - food_j^{position} \right)$$

$$\text{(6)}$$

if $v_i^k > v_{max}$

then $v_i^k = v_{max}$

end

if $v_i^k < -v_{max}$

then $v_i^k = -v_{max}$

end

if x_i is a leader

then
Update the position of the leader by using the mathematical equation (1)
Else
Update the position of the follower's by using the mathematical equation (7);

$$x_j^i = \frac{1}{3} \left(x_j^i + x_j^{i-1} + v_i^{k+1} \right)$$

$$\text{(7)}$$

end
Amend the search agents based on the upper and lower bounds of variables
end
Return F

Here, the proposed approach plays a important role for trapping the solution in the search space during the search process and its advantages has been described as followings;

- The proposed method hold the best optima solution after the each iteration and it can not at all be wiped out even if the entire crowd deteriorates.
- The proposed algorithm updates the location of each search agents in the search space according to the goal which is the best goal obtained so far; hence the best search agent explores and exploits in the search space it for a best one goal.
- The HSSAPSO update the position of follower agents with the help of velocity concept. This concept plays a important role for ignoring the wrong position and fastly trapping global optima in the search space. It's also enhanced the exploration and exploitation balance for trapping fastly optima during the search process.

Table 3 Multimodal benchmark functions

Function	Dim	Range	$f_{\min}$
$F_8(y) \sim \sum\limits_{i=1}^{n} -y_i \sin\left(\sqrt{|y_i|}\right)$	30	$[-500, 500]$	-418.9829×5
$F_9(y) \sim \sum\limits_{i=1}^{n} \left[y_i^2 - 10 \times \cos(2\pi y_i) + 10\right]$	30	$[-5.12, 5.12]$	0
$F_{10}(y) \sim -20e^{\left(-0.2\sqrt{\frac{1}{n}\sum\limits_{i=1}^{n} y_i^2}\right)}$ $\phantom{F_{10}} - e^{\left(\frac{1}{n}\sum\limits_{i=1}^{n} \cos(2\pi y_i)\right)} + 20 + e$	30	$[-32, 32]$	0
$F_{11}(y) \sim \frac{1}{4000} \sum\limits_{i=1}^{n} y_i^2 - \prod\limits_{i=1}^{n} \cos\left(\frac{y_i}{\sqrt{i}}\right) + 1$	30	$[-600, 600]$	0
$F_{12}(y) \sim \frac{\pi}{n}\left\{10\sin(\pi x_i) + \sum\limits_{i=1}^{n-1}(x_i-1)^2\left[1+10\sin^2(\pi x_{i+1})+(x_{n-1})^2\right]\right\}$ $\phantom{F_{12}} + \sum\limits_{i=1}^{n} u(y_i, 10, 100, 4)$ $x_i = 1 + \frac{y_i+1}{4}$ $u(y_i, a, k, m) = \begin{cases} k(y_i-a)^m & y_i > a \\ 0 & -a < y_i < a \\ k(-y_i-a)^m & y_i < -a \end{cases}$	30	$[-50, 50]$	0
$F_{13}(y) \sim 0.1\left\{\begin{array}{l}\sin^2(3\pi y_i) + \sum\limits_{i=1}^{n}(y_i-1)^2[1+\sin^2(3\pi y_i+1)] \\ +(y_n-1)^2[1+\sin^2(2\pi y_n)]\end{array}\right\}$ $\phantom{F_{13}} + \sum\limits_{i=1}^{n} u(y_i, 5, 100, 4)$	30	$[-50, 50]$	0

- Gradual movements of follower search agents during the search process in the search space prevent the HSSAPSO algorithm from effortlessly stagnating into local optima.
- The main parameter of SSA algorithm helps the HSSAPSO for reducing the complexity and makes it effortless to execute.

4 Numerical and statistical experiments

The numerical and statistical results of the proposed approach are compared to SSA, PSO, MFO, SCA and DA algorithms for the twenty-two standard benchmark functions. HSSAPSO, SSA, PSO, MFO, SCA and DA algorithms are programmed using Matlab 2015. Further details about the experiments are discussed in the following subsections.

4.1 Parameter settings

The literature review shows that researchers have mostly used constants for controlling the speed of the agents, exploration and exploitation tendency. Different settings of parameters have been employed in the search process. The following settings of parameters are used in this experiment

to enhance and improve the convergence ability of the meta-heuristics (Table 1).

4.2 Benchmark or standard functions

The initial ability of the HSSAPSO was verified on uni-modal, multi-modal and fixed dimension multi-modal functions. The brief information of these tested functions is described in the Tables 2, 3 and 4 respectively.

4.3 The convergence performance of HSSAPSO on 100–500 dimensions

In order to make sure the effectiveness of the partitioning process and integrating the standard SSA and PSO algorithm, the general convergence ability of the newly existing approach HSSAPSO and the others are presented for benchmark functions by plotting the function values versus the number of generations as described in Fig. 2. In Fig. 2, the red line indicates the standard SSA, while the black line indicates the newly existing variant HSSAPSO. Here, with the help of these conference graphs we can easily see that the velocity concept playing the important role for search agents in ignoring the wrong positions or directions and trapping

Table 4 Fixed-dimension multimodal benchmark functions

Function	Dim	Range	f_{min}
$F_{14}(y) \sim \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2}(y_i - a_{ij})^6} \right)^{-1}$	2	$[-65, 65]$	1
$F_{15}(y) \sim \sum_{i=1}^{11} \left[a_i - \frac{y_1(b_i^2 + b_i y_2)}{b_i^2 + b_i y_i + y_4} \right]^2$	4	$[-5, 5]$	0.00030
$F_{16}(y) \sim 4y_1^2 - 2.1y_1^4 + \frac{1}{3}y_1^6$ $+ y_1 y_2 - 4y_2^2 + 4y_2^4$	2	$[-5, 5]$	-1.0316
$F_{17}(y) \sim \left[1 + (y_1 + y_2 + 1)^2 \binom{19 - 14y_1 + 3y_1^2}{-14y_2 + 6y_1 y_2 + 3y_2^2} \right]$ $\times \begin{bmatrix} 30 + (2y_1 - 3y_2)^2 \\ \times \binom{18 - 32y_1 + 12y_1^2}{+48y_2 - 36y_1 y_2 + 27y_2^2} \end{bmatrix}$	2	$[-2, 2]$	3
$F_{18}(y) \sim -\sum_{i=1}^{4} c_i e^{\left(-\sum_{j=1}^{3} a_{ij}(y_j - p_{ij})^2 \right)}$	3	$[1, 3]$	-3.86
$F_{19}(y) \sim -\sum_{i=1}^{4} c_i e^{\left(-\sum_{j=1}^{6} a_{ij}(y_j - p_{ij})^2 \right)}$	6	$[0, 1]$	-3.32
$F_{20}(y) \sim -\sum_{i=1}^{5} \left[(Y - a_i)(Y - a_i)^T + c_i \right]^{-1}$	4	$[0, 10]$	-10.1532
$F_{21}(y) \sim -\sum_{i=1}^{7} \left[(Y - a_i)(Y - a_i)^T + c_i \right]^{-1}$	4	$[0, 10]$	-10.4028
$F_{22}(y) \sim -\sum_{i=1}^{10} \left[(Y - a_i)(Y - a_i)^T + c_i \right]^{-1}$	4	$[0, 10]$	-10.5363

the fastly global optima in the search space during the search process. It can search the solution for the given functions in least number of iterations as comparison to others and save the wasting time. The data in Fig. 2 is plotted after d (dim) generations. The convergence results of Fig. 2, reveals that the new approach is superior to the standard SSA and others which verifies that the applied partitioning mechanism and the integration amid the standard salp swarm strategy and standard particle swarm optimization algorithm can accelerate the convergence of the newly hybrid approach.

5 Experiment and results

In this part, twenty-two standard tested problems have been used [26, 61] demonstrate the quality, superiority and accuracy of the newly hybrid approach, where obtained optimal solutions by the newly approach have been compared with the standard SSA, PSO, MFO, SCA and DA algorithms. The standard benchmark functions like unimodal, multimodal and fixed dimension multi-modal functions or problems. The description and the convergence graph of all problems are represented in Tables 5, 6 and 7 and Figs. 3,

4 and 5 respectively. The statistical measures in terms of maximum and minimum objective function value, average, best score and standard deviation are recorded. Further, the welded beam design function has been solved by HSSAPSO algorithm and results are reported in the last section of this text. On the basis of obtained solutions, the importance and advantages of the newly hybrid approach has been discussed in the following subsections.

5.1 Results on unimodal benchmark test functions

In this subsection, has been utilized the unimodal function for verifying the working performance of algorithms. The numerical experimental solutions on these functions are discussed in Table 5. This table shows the comparison between the HSSAPSO, SSA, PSO, MFO, SCA and DA algorithms. Here, it can be easily seen that the hybrid algorithm gives superior or highly effective optimal solutions as compared to others. As previously discussed, the uni-modal benchmark tested functions are able for benchmarking exploitation of the approaches. Therefore, have been obtained optimal solutions proofs high rate of exploitation potential of the hybrid algorithm.

Fig. 2 Convergence performance graphs of meta-heuristics on 100–500 dimensions
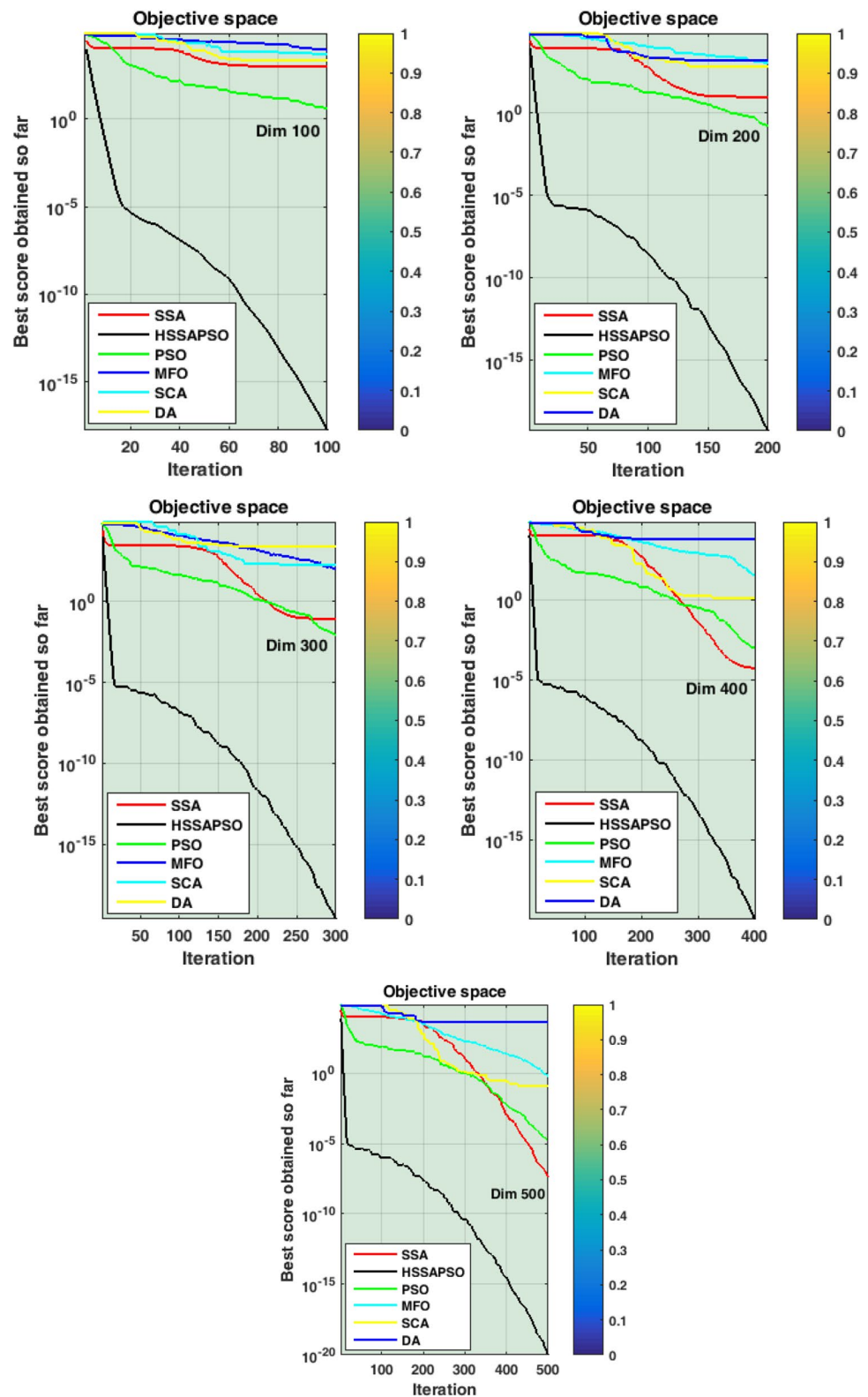
Table 5 Results of unimodal benchmark problems on 200 and 500 dimensions

Problem	Algorithm	Best score	Min	Max	SD	Mean	Dim	Search agents
1.	SSA	5.7926e−08	0	2.3852e+04	5.4544e+03	4.6203e+03	500	30
	HSSAPSO	1.1939e−20	0	8.6710e+03	1.4622e+04	1.0045e+04		
	PSO	3.7076e−04	3.7076e−04	7.1953e+04	6.8614e+03	1.0395e+03		
	MFO	3.2973	3.2973	6.9638e+04	1.9768e+04	1.2981e+04		
	SCA	7.0898	0	7.0634e+04	1.5236e+04	1.1256e+04		
	DA	1.3937e+03	1.3937e+03	6.3050e+04	2.2496e+04	1.4705e+04		
2.	SSA	2.2585e−05	0	32.0671	5.1827	4.4401	500	30
	HSSAPSO	1.2491e−11	0	157.8076	0.3768	0.0277		
	PSO	4.4620e−13	4.4620e−13	26.6797	3.5224	2.0653		
	MFO	4.3142e−09	4.3142e−09	57.2575	62.0730	9.4139		
	SCA	7.0800e−10	0	18.8925	4.3697	1.7218		
	DA	3.5979	3.5979	37.8736	12.8065	12.2109		
3.	SSA	1.9984e−08	0	6.3457e+03	1.3181e+03	933.6434	500	30
	HSSAPSO	1.8638e−20	0	7.8141e+03	1.1254e+03	994.0973		
	PSO	6.5057e−07	6.5057e−07	1.9732e+04	1.8661e+03	301.5329		
	MFO	6.6667e+03	6.6667e+03	1.5282e+04	1.2725e+03	7.1233e+03		
	SCA	3.7880e−05	0	1.6814e+04	3.3340e+03	1.9429e+03		
	DA	30.1597	30.1597	2.0518e+04	3.6640e+03	1.9544e+03		
4.	SSA	5.9776e−05	0	43.8478	12.1112	11.1070	500	30
	HSSAPSO	2.7882e−+11	0	96.2945	1.2939	0.0924		
	PSO	4.6958	1.2415e−05	70.4458	7.6754	1.7416		
	MFO	3.7007	3.7007	80.0806	14.9159	10.5832		
	SCA	7.9072e−05	0	66.8571	14.0143	7.8260		
	DA	4.2288	4.2288	70.9532	20.3763	15.4620		
5.	SSA	20.2446	0	3.4403e06	4.5862e+05	2.9579e+05	200	30
	HSSAPSO	8.7330	0	7.8138e+05	1.5259e+04	4.0083e03		
	PSO	23.2351	0.6246	2.0374e+07	1.6783e+06	1.9379e+05		
	MFO	33.6766	33.6766	5.6714e+07	8.6290e+06	2.9114e+06		
	SCA	8.9566	0	7.4528e+06	3.3673e+06	2.1655e+06		
	DA	8.9755e+03	8.4755e+03	7.0845e+07	1.9161e+07	5.0671e+06		
6.	SSA	3.2341e−09	0	5.2263e+03	568.8210	386.0136	200	30
	HSSAPSO	2.3760e−11	0	6.2355e+03	59.3087	14.0814		
	PSO	2.6748e−09	2.6748e−09	1.9446e+04	2.4597e+03	548.3564		
	MFO	9.2940e−04	9.2940e−04	1.2481e+04	2.7803e+03	1.1489e+03		
	SCA	3.5955	0	1.1383e+04	2.6547e+03	1.2365e+03		
	DA	8.2934	8.2934	2.2353e+04	7.0066e+03	3.5195e+03		
7.	SSA	0.0230	0	1.2617	0.0826	0.0712	500	30
	HSSAPSO	0.0059	0	9.2726	0.0138	0.0011		
	PSO	0.0069	0.0069	8.2005	0.9444	0.5463		
	MFO	0.0072	0.0072	3.9423	0.4952	0.1267		
	SCA	5.5911e−04	0	4.4394	0.2641	0.0718		
	DA	0.0260	0.0260	7.6920	2.2553	0.9315		

5.2 Results on multi-modal benchmark test functions

The optimal solutions of meta-heuristic approaches on the multimodal problems/functions have been discussed in this section and also verify the working ability of the algorithms.

These numerical experimental optimal results are represented in Table 6. The accuracy and ability of the presented approach has been verified in the terms of best scores, least and maximum cost of the objective function and statistical experiments on different dimensions. Table 6, results prove

Table 6 Results of multimodal benchmark problems on 200 and 500 dimensions

Problem	Algorithm	Best score	Min	Max	SD	Mean	Dim	Search agents
8.	SSA	−2.9432e+03	−2.9432e+03	0	163.2604	−2.4330e+03	200	30
	HSSAPSO	−3.0449e+03	−3.0449e+03	0	135.4370	−2.5385e+03		
	PSO	−2.0428e+03	−2.0428e+03	−1.1427e+03	171.4619	−1.9742e+03		
	MFO	−2.9994e+03	−2.9994e+03	−922.2770	494.5624	−2.7243e+03		
	SCA	−1.7365e+03	−1.7365e+03	0	140.1863	−1.6785e+03		
	DA	−2.4542e+03	−2.4542e+03	−1.1783e+03	452.3059	−2.0922e+03		
9.	SSA	23.8790	0	97.6607	14.4804	36.9362	500	30
	HSSAPSO	0	0	172.7369	4.2308	0.3255		
	PSO	12.9345	12.9345	111.1988	21.4631	33.0228		
	MFO	10.9445	10.9445	135.3036	23.0731	19.9183		
	SCA	1.2932e−12	0	95.7572	27.8833	18.2722		
	DA	37.1460	37.1460	108.7236	20.4295	52.9479		
10.	SSA	8.7643e−06	0	18.1594	5.3141	4.2305	500	30
	HSSAPSO	3.3038e−11	0	23.9880	0.7524	0.0627		
	PSO	3.8804e−11	3.8804e−11	19.0182	2.6527	1.6418		
	MFO	9.7300e−08	9.7300e−+08	19.6886	4.7199	2.1465		
	SCA	7.7344e−06	0	20.1437	8.6185	5.8990		
	DA	4.0672	0.0672	19.6234	6.4737	5.6422		
11.	SSA	0.1698	0	25.5869	8.4919	7.1692	500	30
	HSSAPSO	0	0	288.5100	0.8415	0.0519		
	PSO	2.2536	0.0615	88.7712	21.5538	7.9763		
	MFO	0.3470	0.3470	144.8050	20.7299	5.9697		
	SCA	0.0049	0	204.9183	29.2239	8.8299		
	DA	1.3078	1.3078	138.1165	23.3066	9.7577		
12.	SSA	0.4242	0	1.9563e+06	9.0642e+06	1.6256e+04	200	30
	HSSAPSO	0.0011	0	5.2996e+04	3.1473e+03	265.2714		
	PSO	4.1278e−10	4.1278e−10	1.0339e+08	9.0642e+06	1.0388e+06		
	MFO	2.5638e−05	2.5638e−05	4.0073e−07	4.4967e+07	4.1218e+06		
	SCA	0.0808	0	2.2523e+08	3.1206e+07	9.1233e+06		
	DA	7.8924	7.8924	9.7833e+07	3.2051e+07	1.4025e+07		
13.	SSA	3.8892e−04	0	6.5479e+06	1.4490e+06	7.8013e+05	200	30
	HSSAPSO	0.0110	0	8.5618e+05	1.0541e+04	4.2813e+03		
	PSO	1.3419e−07	1.3419e−07	4.6884e+07	4.8102e+06	5.8085e+05		
	MFO	0.0110	0.0110	4.6780e+07	6.8358e+06	1.5392e+06		
	SCA	0.4525	0	1.8438e+08	1.8409e+07	1.9738e+06		
	DA	0.3403	0.3403	7.7773e+07	3.7598e+07	2.1056e+07		

that the newly hybrid approach gives best score values, best minimum and maximum values on different dimensions.

Further, the results strongly demonstrate that high exploration of newly hybrid approach is capable to explore the search space extensively and provide capable regions of the search space.

5.3 Results on fixed dimension multi-modal benchmark test functions

Further, in this subsection, the results of HSSAPSO on fixed dimension multi-modal problems are discussed and

also verifying the convergence performance with other meta-heuristics in Table 7. Results are prove that the hybrid approach is capable to gives the best quality of solutions in the search area of these functions.

Hence, it can be concluded that the HSSAPSO algorithm has better characteristics in high quality of the optimal results and strength of the optimal solutions.

5.4 Statistical results of the algorithms

In this subsection, the working performance of the newly modified method has been verified by statistical solutions on

Table 7 Results of fixed dimension multi-modal benchmark problems on 50–80 dimensions

Problem	Algorithm	Best score	Min	Max	SD	Mean	Dim	Search agents
14.	SSA	3.9683	0	11.3839	1.1989	4.0488	50	30
	HSSAPSO	0.9980	0	65.0054	0.1415	9.9805		
	PSO	7.8740	7.8740	20.5206	3.4435	11.0557		
	MFO	5.9288	5.9288	49.6864	10.4699	8.7070		
	SCA	10.7632	0	11.1822	1.5397	10.6156		
	DA	2.9821	2.9821	215.0151	29.9163	12.0821		
15.	SSA	4.0204	0	0.1730	0.0166	0.0237	500	30
	HSSAPSO	3.07493e−04	0	0.9463	0.0071	0.0024		
	PSO	8.9060e−04	8.9060e−04	0.0518	0.0084	0.0029		
	MFO	7.82663e−04	7.8266e−04	0.8625	0.0546	0.0050		
	SCA	7.2909e−04	0	0.0327	0.0036	0.0016		
	DA	9.0452e−04	9.0452e−04	0.2349	0.0187	0.0094		
16.	SSA	−1.0316	−1.0316	0	0.1399	−0.9640	100	30
	HSSAPSO	−1.0316	−1.0316	0	0.1078	−0.9976		
	PSO	−1.0316	−1.0316	0	0.1401	−1.0128		
	MFO	−1.0316	−1.0316	−0.0532	0.1543	−0.9897		
	SCA	−1.0316	−1.0315	0	0.1865	−0.9916		
	DA	−1.0316	−1.0316	−0.3029	0.1508	−0.9742		
17.	SSA	3.0000	0	8.7831	1.5112	3.6260	80	30
	HSSAPSO	3.0000	0	85.4961	1.0241	5.1069		
	PSO	3.0000	3.0000	65.9740	7.0347	3.9725		
	MFO	3.0000	3.0000	80.6247	17.8383	7.9745		
	SCA	3.0000	0	10.5066	1.5860	3.6260		
	DA	3.0000	3.0000	48.8171	10.2422	6.7816		
18.	SSA	−3.8549	−3.8549	0	0.5630	−3.6345	50	30
	HSSAPSO	−3.8628	−3.8628	0	0.4459	−3.7825		
	PSO	−3.8628	−3.8628	−2.4674	0.2014	−3.8071		
	MFO	−3.8628	−3.8628	−3.1070	0.1496	−3.8221		
	SCA	−3.8472	−3.8472	0	0.5375	−3.6949		
	DA	−3.8292	−3.8628	−3.7051	0.4523	−3.8292		
19.	SSA	−3.1899	−3.1899	0	0.2841	−3.0542	500	30
	HSSAPSO	−3.3220	−3.3220	0	0.1099	−3.0233		
	PSO	−3.2031	−3.2031	−1.0762	0.1356	−3.1689		
	MFO	−3.3220	−3.3220	−1.5623	0.1518	−3.2952		
	SCA	−2.9132	−2.9132	0	0.3057	−2.7119		
	DA	−3.3217	−3.3217	−1.4780	0.5247	−3.1342		
20.	SSA	−2.6829	−2.6829	0	0.6835	−2.0970	50	30
	HSSAPSO	−5.0994	−5.0994	0	0.0231	−4.3374		
	PSO	−5.1005	−5.1005	−0.8952	1.4115	−3.6668		
	MFO	−5.0975	−5.0975	−0.5036	1.7439	−3.0601		
	SCA	−0.8812	−0.8812	0	0.1395	−0.8064		
	DA	−5.0994	−5.0994	−0.3078	2.0655	−2.8166		
21.	SSA	−2.7519	−2.7519	0	0.8814	−2.0175	100	30
	HSSAPSO	−5.0877	−5.0877	0	0.4000	−3.9669		
	PSO	−1.8376	−1.8376	−0.4561	0.3273	−1.6577		
	MFO	−3.7243	−3.7243	−0.4817	0.9090	−3.2069		
	SCA	−1.5233	−2.7519	0	0.8814	−2.0175		
	DA	−5.0288	−5.0288	−0.9433	0.9602	−4.4089		

Table 7 (continued)

Problem	Algorithm	Best score	Min	Max	SD	Mean	Dim	Search agents
22.	SSA	−2.8711	−2.8711	0	0.7695	−2.1808	50	30
	HSSAPSO	−5.1285	−5.1285	0	1.1737	−4.2235		
	PSO	−2.8711	−2.8711	−0.5628	0.6432	−2.3018		
	MFO	−5.1284	−5.1284	−1.0227	1.3751	−4.0762		
	SCA	−2.7511	−2.7511	0	0.7695	−2.1808		
	DA	−3.8898	−5.1285	−1.4964	1.6692	−3.8898		

the fixed dimension, multi-modal and uni-modal functions. These solutions are presented in Tables 5, 6 and 7. In order to do a fair comparison of a HSSAPSO algorithm with SSA, PSO, MFO, SCA and DA algorithms, average and standard deviation for multiple runs has been reported. In Tables 5, 6 and 7, the least statistic scores indicate that the HSSAPSO algorithm is the most robust and is competent to reproduce the best solution with the minimum discrepancy and has less dependence as comparison to others.

5.5 Convergence graphs

At the end, the convergence performance of the presented approach and others has been plotted on Figs. 3, 4 and 5. The accuracy of the meta-heuristics has been tested on different dimension in the search space of the functions and each algorithms code has been run several times for the test the fair comparison of the algorithms. Figures 3, 4 and 5, shows that the HSSAPSO algorithm finds the best optimal solution of the problems in least number of generations or iterations. But, on the other sides, recent algorithms are taking the maximum times for find the solution in the search areas of the functions.

Hence, the graphs result proves that the HSSAPSO approach more suitable and faster for benchmark function as comparison to others.

5.6 Comparison of the algorithms on high dimensional (CEC 2017) functions

Due to the high complexity, the CEC test functions [62] customized for global optimization has been used these function for evaluating the performance quality of the proposed algorithm as comparison to others. The accuracy of the algorithms has been compared on 100 dimension and 5000 iterations with each others. The CEC 2017 test function contains twenty-nine standard functions, but function (F2) has been excluded because it reveals unstable behavior as given in [62]. These functions have been described in four different categories such as uni-modal functions (F1 and F3), multimodal functions (F4–F10), hybrid functions (F11–F20) and composition functions (F21–F30). Here multimodal

standard function has been utilized to evaluate the competitive metaheuristics to avoid fall into local optimum because these functions have several local values. Further, the composite standard functions have been utilized to evaluate the balance amid exploration and exploitation phases.

Through the Table 8 has been reported the simulation global optimum solutions of the algorithms. The convergence performance of the algorithms has been compared through Fig. 6. These graphs have been plotted on 5000 iterations at 100 dimensions.

The experimental results of the SSA, PSO, MFO, SCA, EGWO, AGWO, GWOPSO and HSSAPSO algorithms have been compared through minimum value of objective function, maximum value of objective function, mean value and standard deviation value. The simulation results on all functions reveals that the proposed algorithms has very superior than others for evaluating the best global optima solution in the search space during the search process. Hence, it's capable for best solution for complex functions. According to the aforementioned findings, the following observations can be made.

- Unimodal functions: These functions have only one global optima, hence these functions have been utilized to evaluate the exploitation phase. The obtained global optimal solutions shows that proposed algorithm has exhibited a superior exploitation capability outperform than others.
- Multimodal functions; in which have many local optima and the number of design variables exponentially increases with the problem size compared to the unimodal functions. Generally these test suits utilized for representing the exploration ability of the metaheuristics. Here, the obtained solution shows that the proposed algorithm yields a superior exploration capability.
- Composition and Hybrid functions generally used for assessing the local global optima avoidance capability and the balance amid exploitation and exploration, which have a huge number of local global optima. The obtained simulation solutions prove that the proposed method is able to create the superior balance amid exploration/exploitation.

Fig. 3 HSSAPSO, SSA, PSO, MFO, SCA and DA algorithms convergence curves for uni-modal test functions
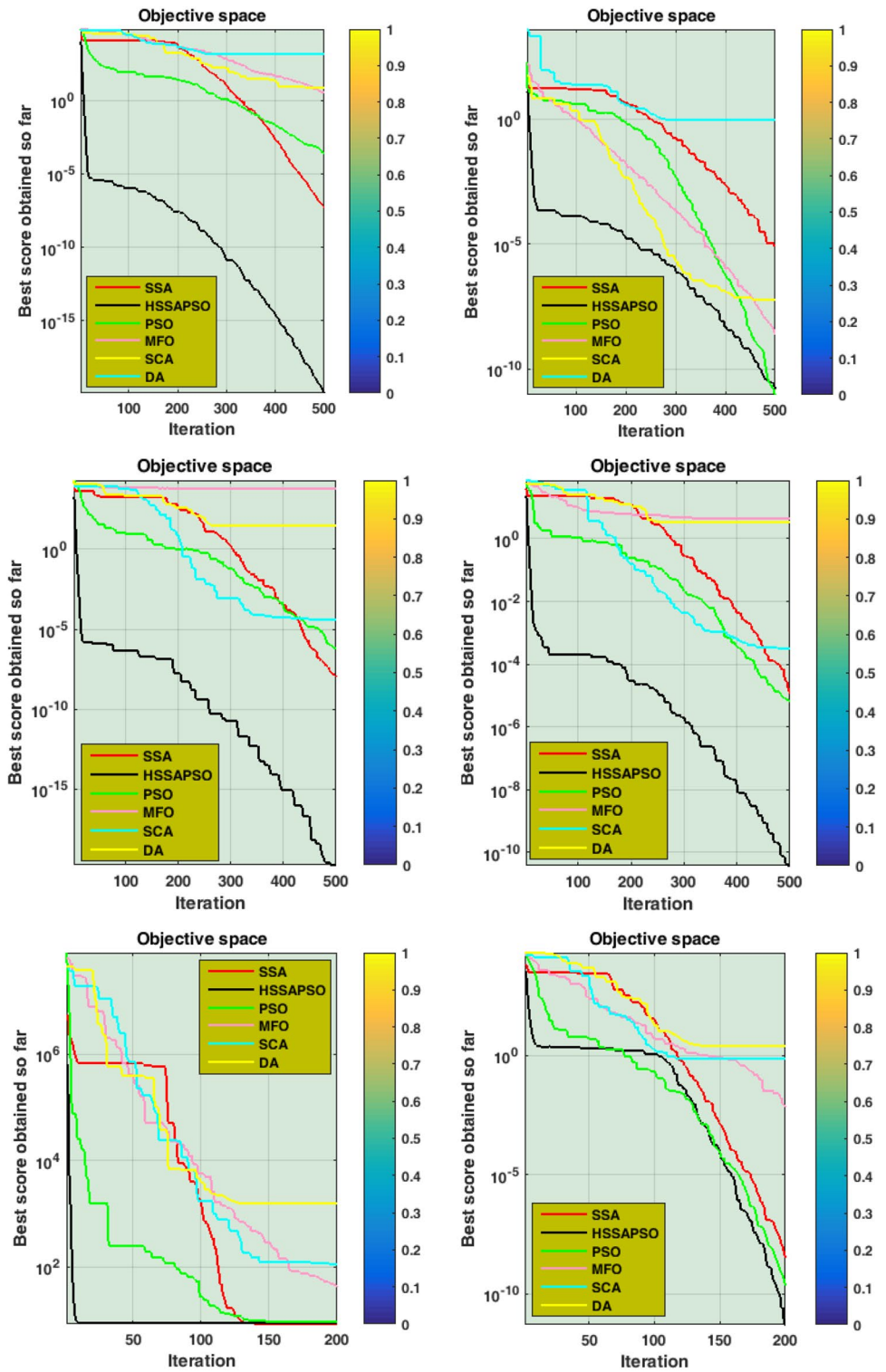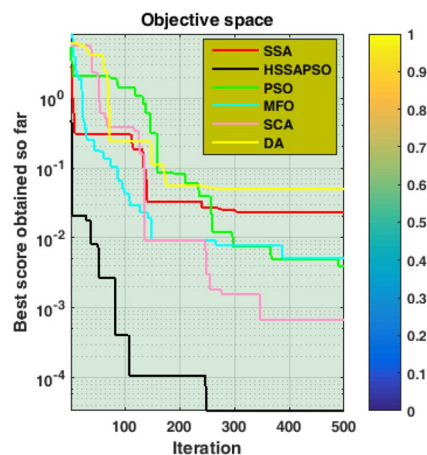
Fig. 3 (continued)

- Hence, the proposed methodology is most efficient approach for all CEC 2017 test suits outperform than others.

5.7 Testing performance of the HSSAPSO algorithm through Wilcoxon signed ranks method

In this sub-section, the working performance of the HSSAPSO algorithm has been confirmed by applying the Wilcoxon signed ranks (WSR) method on the median values (see in Tables 9, 10, 11) of the algorithms for the superior assessment [63]. The WSR test is a non-parametric test that is applied on two distinct samples or data columns, for searching the significance amid two samples/or two algorithms. With the help of this method/or test, we can be easily find the best one sample/or algorithms. In addition, the help of this method easily locates the significant difference of the behaviors of two meta-heuristics.

Here after using the WSR test we found the z-value and p value of the each algorithm. These values have been represented in the Tables 12, 13 and 14. Further we have used the condition of this test for verifying the significance of each algorithms, such as if $p < 0.05$, then it represents a rejection of the H_0 hypothesis, whereas $p > 0.05$ represents a failure to reject the null hypothesis. Hence p-values are less than 0.05, it can be determined that HSSAPSO algorithm is significantly superior to the other meta-heuristics. If not, the obtained improvements are not statistically significant.

To sum up, all numerical and statistical solutions has proven that the HSSAPSO algorithm is more helpful in enhancing and improving the convergence performance of the Salp Swarm Algorithm in the terms of optimal solution superiority as well as computational efforts.

Finally, the results of Tables 12, 13 and 14, illustrates that proposed method has better characteristics such that strength of the global optimal target and superiority of the optimal solution. Also, significant importance may be placed

in global exploration and local exploitation. Solutions illustrate based on the WSR method proved that the proposed algorithm presents the superiority performance as comparison to recent meta-heuristics. Hence, the obtained results by the proposed algorithm have been statistically superior and this has not happened by likelihood/or chance.

6 Welded beam design function

Welding can be presented as a procedure of combining metallic parts by heating to appropriate temperature without or with the application of pressure [64]. Here, the welding course that uses heat alone is known as mixture welding process. The components involved in this procedure are kept in position and given to attach the melted metal. Molten metal that can come from any part that it's called parent metal or external filler metal. The merging surface of different parts becomes plastic under the action of heat. When consolidated, two parts fuse into a unit.

A beam is a member subjected to loads applied transverse to the long dimension, causing the member to bend. Beam is often distributed on a support or reaction basis. Here the beam supported by smooth surface, rollers, pins at the split ends is known as simple beam. It will develop a general reaction to a beam assisted action but will not create a momentum at the reactions. If a beam project other than the supporters either ends either side then it is called the simple beam with the overhang.

A beam (A) is to be welded to a rigid member (B). The welded beam is to consist of 1010 steel and is to support a force P of 6000 lb. Beam dimensions should be selected so that the cost of the system is reduced. The structure of the beam has been illustrated by Fig. 7. In this image an rigid member (A) has been welded on one beam. A weight or load is applied at the last part of the member. Here, the beam is to be adapted for minimize cost by varying the weld

Fig. 4 HSSAPSO, SSA, PSO, MFO, SCA and DA algorithms convergence curves for multi-modal test functions
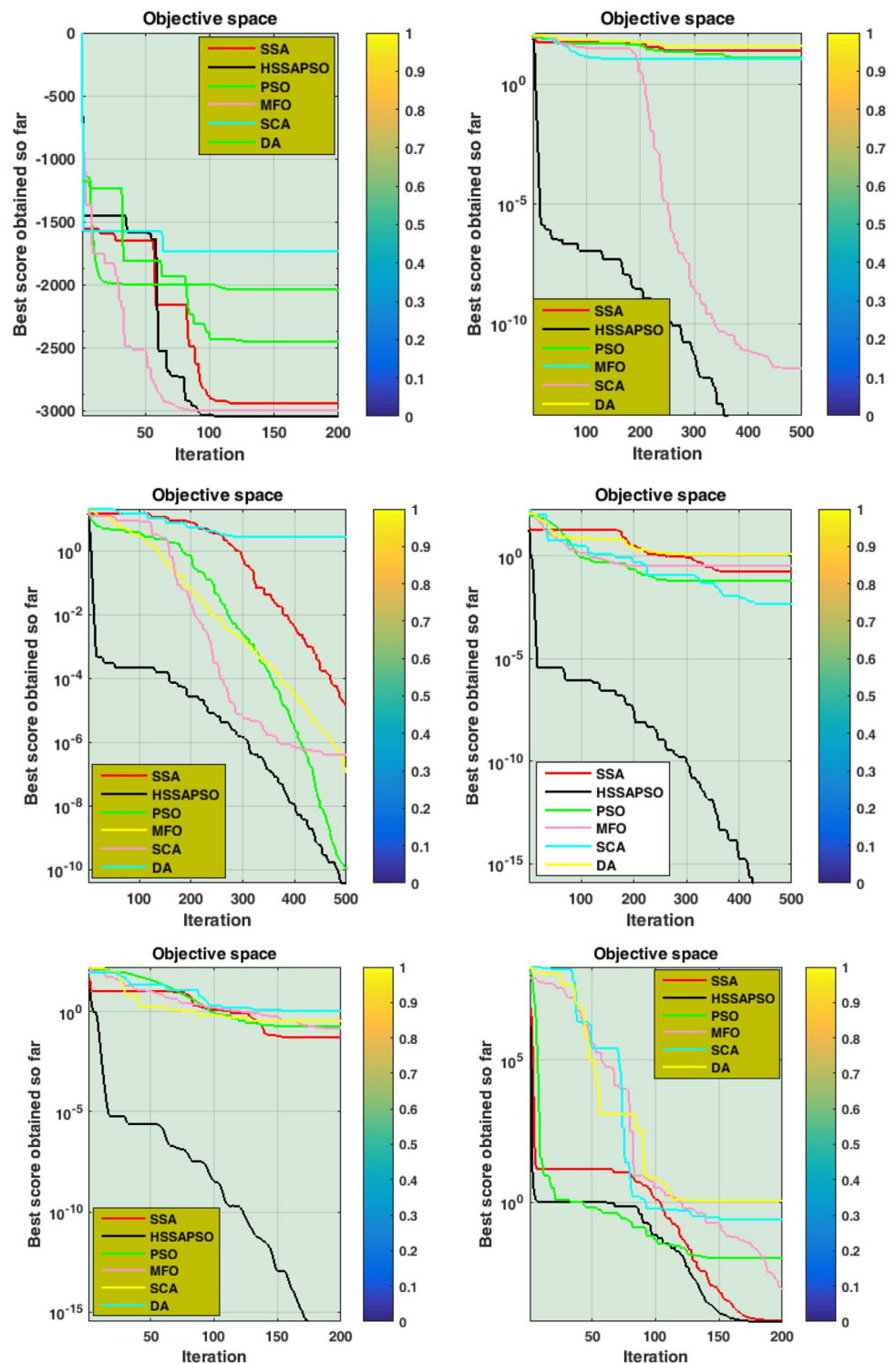
and member dimensions $\vec{y} = [y_1\, y_2\, y_3\, y_4] = [h\ l\ t\ b]$. This includes the limits of the end deflection, bending stress, buckling load and shear stress. The variables y_1 and y_2 are multiples of 0.0625 inch and but for this problem are assumed continuous [65, 66].

The main objective of this function is to minimize the total material and fabrication cost of a beam that is loaded in bending shown in Fig. 8 [67]. The beam dimensions are varied to decrease the total mass (thus reducing fabrication cost). However, the cost of welding is also measured,

Fig. 5 HSSAPSO, SSA, PSO, MFO, SCA and DA algorithms convergence curves for fixed dimension multi-modal test functions
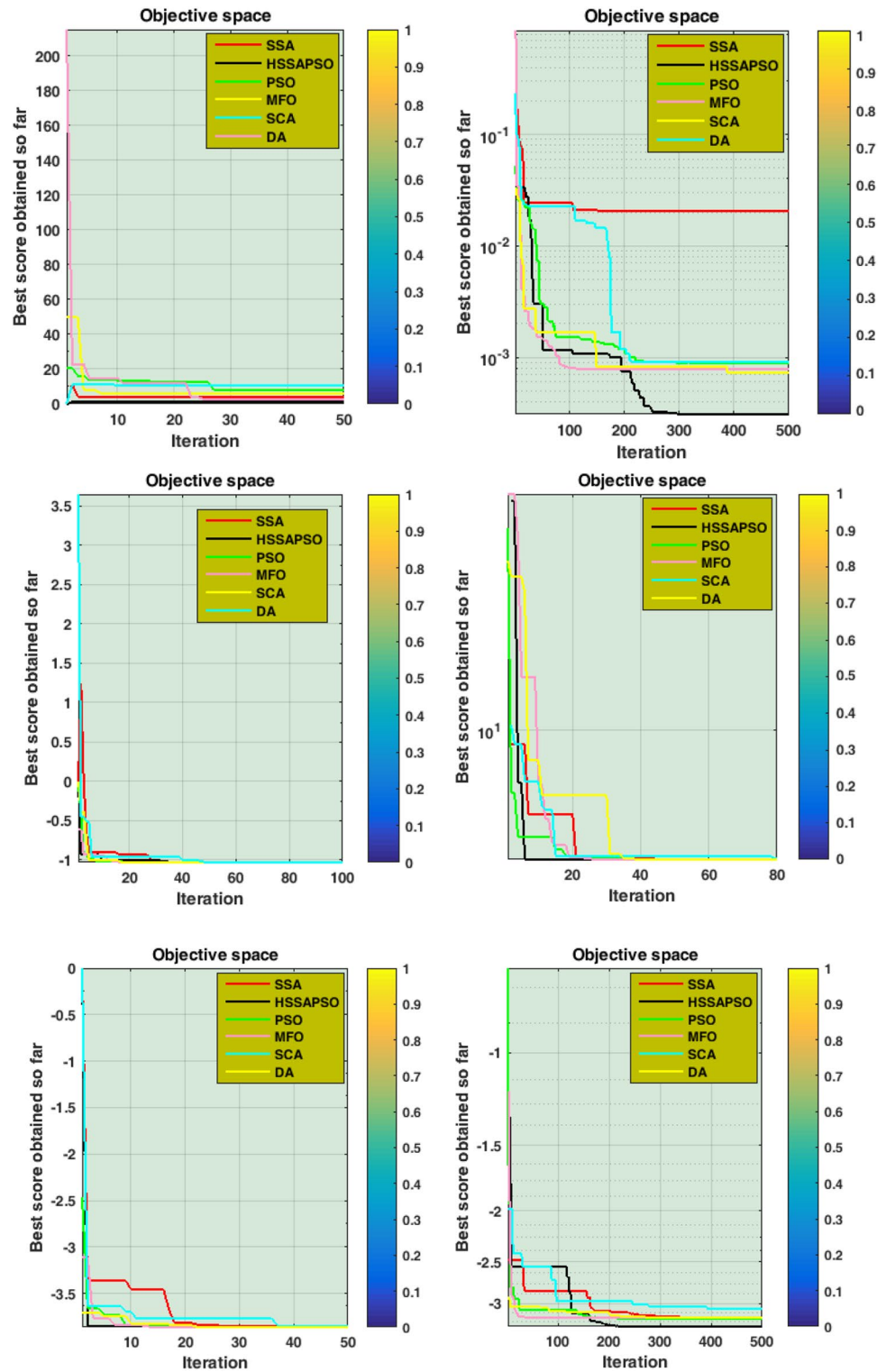

introducing more complexity to the problem. The subject to constraints of the problems are on l: shear stress, m: bending stress in the beam, n: buckling load on the bar, o: end deflection of the beam and side constraints. This function is depend on four variables such as $a(y_1)$: thickness of the weld

(h), $b(y_2)$: length of the clamped bar (l), $c(y_3)$: height of the bar (t) and $d(y_4)$: thickness of the bar (b) [67]. The function can be indicated as the following mathematical equations:

Consider $\vec{y} = \begin{bmatrix} y_1 \, y_2 \, y_3 \, y_4 \end{bmatrix} = [h \, l \, t \, b]$ (8)

Fig. 5 (continued)
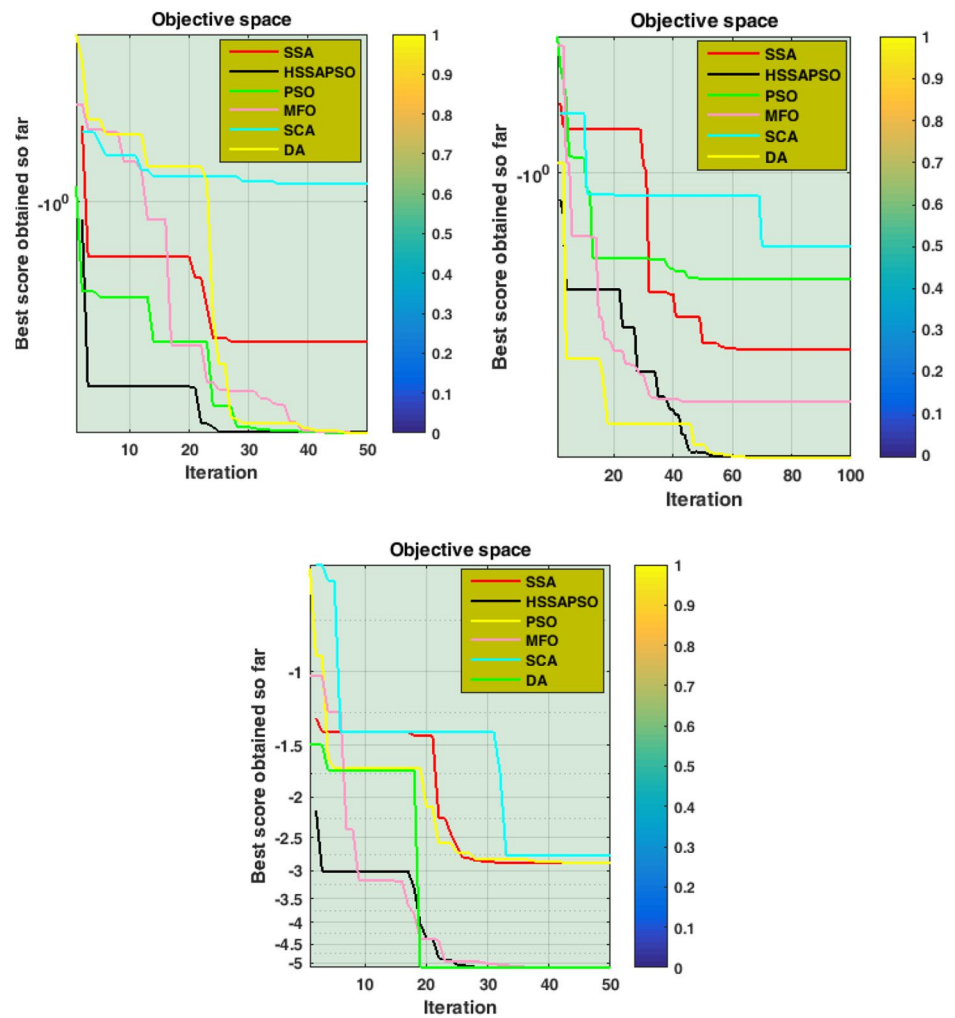
$$\text{Minimize}: f(Y) = 1.10471y_1^2 y_2 + 0.04811y_3 y_4 (14.0 + y_2) \tag{9}$$

$$r_1(Y) = l(Y) - l_{max} \le 0, \tag{10}$$

$$r_2(Y) = m(Y) - m_{max} \le 0, \tag{11}$$

$$r_3(Y) = y_1 - y_4 \le 0, \tag{12}$$

$$r_4(Y) = 0.010471y_1^2 + 0.04811y_3 y_4 (14.0 + y_2) \le 0, \tag{13}$$

$$r_5(Y) = 0.125 - y_1 \le 0, \tag{14}$$

$$r_6(Y) = o(Y) - o_{max} \le 0, \tag{15}$$

$$r_7(Y) = n - n(Y) \le 0, \tag{16}$$

where $l(Y) = \sqrt{(l')^2 + l'l''(y_2/2R) + (l'')^2}$, $l' = n/\sqrt{2}y_1 y_2$, $l'' = MR/J$, $M = n(L + y_2/2)$, $R = \sqrt{y_2^2/4 + ((y_1 + y_2)/2)^2}$,

$J = 2\left\{ \sqrt{2}y_1 y_2 \left[y_2^2/12 + ((y_1 + y_2))^2 \right] \right\}$, $m(Y) = 6nL^3/Ey_3^3 y_4$,

$n(Y) = \left(4.013E\sqrt{y_3^2 y_4^6/36/L^2} \right)\left(1 - (y_3/2L)\sqrt{E/4G} \right)$,

$n = 6000$ lb, $L = 14$ in, $E = 30 \times 10^6$ psi, $G = 12 \times 10^6$ psi, $l_{max} = 13,600$ psi, $m_{max} = 30,000$ psi, $o_{max} = 0.25$ in, $Y = (y_1, y_2, y_3, y_4)$ and $(0.1, 0.1, 0.1, 0.1) \le Y \le (2, 10, 10, 2)$.

The welded beam design function has been solved by newly HSSAPSO approach in this section. During this study the compatibility with extreme values of the parameter has been tested with different solvers. The results of the algorithms are of a stochastic type, hence these results have been tested on 20 trial. The results obtained by proposed algorithm has been compared with using the literature results of; ABC (Artificial Bee Colony) [68], CPSOS (Co-Evolutionary PSO) [69], CDE (Co-evolutionary) [70], HIS (Improved Harmony Search Algorithm) [71], MFO (Moth Flame Optimizer) [29], AFA (Adaptive FA) [72], CSS (Charged System Search) [73], LSA-SM (Hybrid Lightning Search Algorithm-Simplex Method) [74] and Water Cycle and Moth-Flame Optimization algorithms (WCMFO) [75] respectively. The results of Table 14, represents the obtained solution in the terms of

Table 8 Results of CEC 2017 on 100 dimensions of algorithms

Algorithms	F-1				F-2			
	Min.	Max	Mean	SD	Min	Max	Mean	SD
AGWO	8.7436e+10	5.8722e+11	1.0573e+11	2.6395e+10	2.5967e+05	1.7006e+12	4.1382e+08	2.4207e+10
EGWO	8.6188e+10	6.1715e+11	9.1301e+10	1.9248e+10	2.3662e+05	9.2129e+09	5.8695e+06	2.2562e+08
MFO	1.4959e+11	6.0861e+11	1.6594e+11	5.4908e+10	4.3100e+05	3.9002e+07	6.7332e+05	1.8826e+06
PSO	1.2885e+04	6.0038e+11	3.0637e+11	2.6633e+10	2.1680e+05	2.3924e+12	4.8263e+08	3.3834e+10
PSOGWO	8.3322e+09	6.7750e+11	6.2987e+11	6.7401e+10	2.2526e+05	2.0029e+12	4.0066e+09	2.8325e+11
SCA	0	6.2775e+11	3.0446e+11	1.5430e+11	0	9.5481e+10	1.0204e+08	2.7218e+09
SSA	0	3.8836e+11	1.1025e+11	1.4300e+11	0	8.6059e+10	2.7214e+07	1.3894e+09
HSSAPSO	0	6.9718e+11	6.7670e+10	1.0428e+11	0	6.7360e+13	1.3557e+04	9.5262e+05
Algorithms	F-3				F-4			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	6.6671e+03	3.1602e+05	1.0853e+04	1.1627e+04	1.6939e+03	2.9310e+03	1.8471e+03	94.3943
EGWO	1.5218e+04	4.3494e+05	1.7010e+04	1.2113e+04	1.7695e+03	2.9036e+03	1.8501e+03	97.9658
MFO	2.8858e+04	3.4552e+05	3.4744e+04	2.5453e+04	1.6916e+03	3.2675e+03	1.7587e+03	190.5930
PSO	775.6263	3.3804e+05	1.7261e+03	1.0734e+04	1.1985e+03	3.2442e+03	1.3301e+03	207.8733
PSOGWO	1.6850e+03	2.6136e+05	2.4243e+03	5.5267e+03	859.6864	3.2793e+03	1.3548e+03	249.7328
SCA	0	2.7651e+05	8.6822e+04	6.7722e+04	0	2.8853e+03	2.2167e+03	260.3634
SSA	0	2.2338e+05	3.0684e+04	4.8758e+04	0	2.5623e+03	1.8362e+03	385.3124
HSSAPSO	0	4.7555e+05	1.0874e+03	1.0051e+03	0	3.4025e+03	1.1015e+03	89.8728
Algorithms	F-5				F-6			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	668.2970	764.4739	675.0074	7.0826	2.4343e+03	1.3959e+04	2.7671e+03	573.4496
EGWO	704.6420	782.0255	710.6183	8.0391	3.5529e+03	1.2407e+04	3.6610e+03	383.8259
MFO	667.9487	772.4922	672.4760	11.9740	5.5769e+03	1.4527e+04	5.9351e+03	1.1863e+03
PSO	658.8228	772.1896	669.9477	16.0018	1.9266e+03	1.4316e+04	2.2490e+03	497.2001
PSOGWO	620.7786	769.8404	644.0656	15.5998	1.8606e+03	1.3474e+04	2.3313e+03	342.4363
SCA	0	767.9188	715.6532	24.8871	0	1.2014e+04	6.0600e+03	3.0979e+03
SSA	0	743.3641	696.1465	23.5754	0	6.7789e+03	3.2750e+03	1.2609e+03
HSSAPSO	0	789.9729	629.7751	5.6209	0	5.4154e+04	2.0305e+03	267.8433
Algorithms	F-7				F-8			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	1.9958e+03	3.5536e+03	2.1290e+03	126.9865	4.9631e+04	2.5903e+05	5.3733e+04	1.1695e+04
EGWO	1.9885e+03	3.5067e+30	2.0760e+03	108.7698	6.1380e+04	1.6582e+05	7.5310e+04	1.4865e+04
MFO	2.0279e+03	3.3703e+03	2.1212e+03	158.8603	4.1062e+04	2.5285e+05	4.8902e+04	2.2398e+04
PSO	1.7621e+03	3.9074e+03	1.8830e+03	198.5284	2.1764e+04	2.4414e+05	4.5453e+04	1.8786e+04
PSOGWO	.8070e+03	3.9462e+03	1.8833e+03	196.0034	5.0245e+04	2.4850e+04	5.5643e+04	1.1995e+04
SCA	0	3.4884e+03	2.6611e+03	354.1787	0	2.7414e+05	1.2326e+04	4.8056e+04
SSA	0	3.0286e+03	2.2353e+03	415.8781	0	1.5008e+05	7.2155e+04	3.0355e+04
HSSAPSO	0	3.9957e+03	1.0244e+03	069.3923	0	2.9948e+05	1.1389e+04	1.1740e+04
Algorithms	F-9				F-10			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	2.9358e+04	3.7325e+04	3.0877e+04	1.2484e+03	5.4516e+04	2.7167e+08	5.6084e+06	3.8421e+08
EGWO	2.1149e+04	3.6931e+04	2.2629e+04	1.9505e+03	5.1829e+04	2.5876e+06	6.5046e+04	6.8428e+04
MFO	1.9342e+04	3.6485e+04	2.0071e+04	2.2816e+03	1.9139e+05	1.7473e+08	7.5527e+06	3.4982e+08
PSO	1.5119e+04	3.6148e+04	1.7948e+04	4.1195e+03	2.0012e+03	1.9174e+09	6.2503e+05	3.0913e+07

Table 8 (continued)

Algorithms	F-9				F-10			
	Min	Max	Mean	SD	Min	Max	Mean	SD
PSOGWO	2.9253e+04	3.6790e+04	3.1817e+04	893.0032	6.4281e+04	8.9500e+07	1.1719e+05	1.7893e+05
SCA	0	3.6366e+04	3.2340e+04	889.6159	0	1.2684e+06	2.5031e+05	1.8776e+05
SSA	0	3.6366e+04	2.3757e+04	8.3438e+03	0	1.6123e+07	1.1452e+05	9.1196e+05
HSSAPSO	0	3.8905e+04	1.0729e+04	866.8765	0	1.9045e+10	4.9255e+04	5.1422e+04

Algorithms	F-11				F-12			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	4.8285e+06	1.7008e+08	9.3402e+06	1.4136e+07	8.2599e+03	6.0252e+08	1.8053e+05	8.5762e+06
EGWO	4.3598e+04	5.3178e+08	3.4218e+06	8.7304e+07	6.9252e+03	1.5263e+08	7.4891e+04	2.1854e+06
MFO	4.5947e+04	1.7995e+09	4.0425e+06	7.1322e+07	1.8449e+03	1.2945e+08	9.6372e+04	3.3853e+06
PSO	4.4669e+04	2.0752e+08	1.2964e+06	1.0994e+07	3.9955e+03	1.7837e+08	1.1076e+05	3.8417e+06
PSOGWO	2.3521e+04	3.3260e+09	3.7745e+06	5.7037e+07	2.5585e+03	4.7917e+08	1.2461e+05	6.8066e+06
SCA	0	1.2714e+09	2.5401e+07	4.7018e+07	0	5.3078e+08	3.3211e+05	8.8918e+06
SSA	0	1.3566e+09	2.1465e+07	1.0019e+08	0	4.8545e+08	5.4625e+05	7.0285e+06
HSSAPSO	0	4.4885e+09	1.1748e+06	1.0758e+07	0	6.1962e+08	6.8241e+04	1.2894e+06

Algorithms	F-13				F-14			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	7.3087e+06	8.1732e+08	1.2683e+07	3.6436e+07	8.4001e+05	1.0860e+10	1.0326e+07	2.5030e+08
EGWO	7.7040e+05	1.9395e+09	2.6307e+06	3.8845e+07	1.1980e+05	3.5952e+05	1.9094e+06	6.1603e+07
MFO	2.1269e+06	1.9635e+09	6.5316e+06	6.3673e+07	5.5655e+04	4.8642e+09	1.5943e+07	2.3239e+08
PSO	4.1131e+06	2.1774e+09	4.2955e+06	6.0298e+07	2.7978e+03	7.3571e+09	6.3917e+06	1.9124e+08
PSOGWO	6.3348e+06	2.8436e+09	1.3297e+07	4.6100e+07	4.6623e+06	3.1784e+09	1.8630e+08	3.1594e+08
SCA	0	8.6416e+08	8.2125e+07	5.4555e+07	0	7.9890e+09	4.9714e+07	2.1519e+08
SSA	0	1.3854e+09	4.0504e+07	7.3267e+07	0	1.1190e+09	4.8658e+07	1.2809e+08
HSSAPSO	0	9.2538e+09	1.3085e+06	3.3541e+07	0	2.3964e+10	1.0094e+06	1.9958e+07

Algorithms	F-15				F-16			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	1.0854e+04	4.4341e+04	1.1365e+04	910.3137	8.5516e+03	5.9602e+07	2.9602e+04	9.6888e+05
EGWO	1.3731e+04	5.5962e+04	1.4282e+04	1.2911e+03	7.6643e+05	1.1134e+08	8.7984e+05	2.0847e+06
MFO	6.4817e+03	5.1155e+04	6.9201e+03	2.2657e+03	8.5092e+03	4.4725e+08	4.8543e+05	1.1775e+07
PSO	6.1057e+03	4.5461e+04	7.1374e+03	2.2236e+03	5.2478e+03	1.6366e+08	2.0000e+05	5.0115e+06
PSOGWO	8.6066e+03	3.9198e+04	8.9150e+03	855.8692	1.8718e+04	3.5186e+08	4.0734e+05	6.1290e+06
SCA	0	3.3884e+04	1.5209e+04	2.8035e+03	0	8.6642e+07	7.3358e+05	4.3616e+06
SSA	0	3.5155e+04	1.0696e+04	4.7803e+03	0	5.6526e+07	1.2944e+05	1.2449e+06
HSSAPSO	0	5.9290e+04	5.7510e+03	1.2336e+03	0	6.2087e+08	2.1018e+04	2.1616e+05

Algorithms	F-17				F-18			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	9.6711e+06	1.7158e+09	2.5531e+07	5.3688e+07	6.5164e+08	5.7104e+10	1.2899e+09	1.3170e+09
EGWO	6.7700e+06	5.4483e+08	9.0476e+06	1.3261e+07	5.9956e+09	5.9566e+10	6.1580e+09	1.4984e+09
MFO	2.4659e+06	9.8822e+08	1.0253e+07	4.3492e+07	5.8139e+07	6.6649e+10	4.3559e+08	3.4207e+09
PSO	1.6884e+06	3.2664e+09	6.8220e+06	7.8703e+07	9.9707e+03	6.6649e+10	1.3035e+08	2.0850e+09
PSOGWO	3.7701e+06	1.1801e+09	1.2239e+07	3.4371e+07	2.6616e+07	5.6211e+10	1.0801e+09	1.9166e+09
SCA	0	1.6498e+09	1.5474e+08	1.5183e+08	0	6.1036e+10	8.5249e+09	6.4656e+09
SSA	0	1.0786e+09	5.1591e+07	9.6749e+07	0	4.2142e+10	3.0090e+09	5.7992e+09
HSSAPSO	0	8.8823e+09	4.3847e+06	1.2669e+07	0	6.9667e+10	1.2413e+08	1.1054e+09

Table 8 (continued)

Algorithms	F-19				F-20			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	6.5886e+03	9.8485e+03	6.9019e+03	635.2155	3.3616e+03	5.9252e+03	3.5074e+03	136.0660
EGWO	5.5678e+03	9.9133e+03	5.7517e+03	309.5110	3.7876e+03	5.1303e+03	3.8607e+03	192.9396
MFO	4.5636e+03	9.9426e+03	4.6647e+03	502.8372	3.7258e+03	5.7824e+03	3.7835e+03	173.9068
PSO	5.0743e+03	8.9084e+03	5.2775e+03	448.5211	3.4598e+03	5.5222e+03	3.6066e+03	248.1098
PSOGWO	4.9323e+03	1.0365e+04	7.0279e+03	789.2256	3.5693e+03	5.6888e+03	3.6749e+03	113.6552
SCA	0	9.9179e+03	8.1412e+03	503.6686	0	5.2939e+03	4.2506e+03	333.2527
SSA	0	9.9769e+03	6.2707e+03	1.2025e+03	0	5.1889e+03	3.6161e+03	484.9990
HSSAPSO	0	9.9808e+03	4.4857e+03	290.5478	0	5.9961e+03	3.1560e+03	104.2024

Algorithms	F-21				F-22			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	3.3244e+04	3.8619e+04	3.4299e+04	586.2696	4.2580e+03	7.4952e+03	4.3588e+03	182.9933
EGWO	2.2561e+04	3.7865e+04	2.4090e+04	2.3225e+03	5.0815e+03	7.9679e+03	5.1590e+03	194.7915
MFO	2.1369e+04	3.8943e+04	2.2906e+04	2.5291e+03	4.1106e+03	8.9157e+03	4.1463e+03	194.8181
PSO	1.8750e+04	3.8829e+04	2.1433e+04	4.1121e+03	5.0671e+03	8.5115e+03	5.3240e+03	428.5617
PSOGWO	2.0879e+04	3.8014e+04	2.1667e+04	1.3016e+03	3.9492e+03	8.8007e+03	4.1257e+03	172.2201
SCA	0	3.8976e+04	3.5586e+04	690.5235	0	6.3574e+03	4.9523e+03	212.3388
SSA	0	3.9216e+04	2.7337e+04	7.9593e+03	0	6.7323e+03	4.2415e+03	589.0682
HSSAPSO	0	3.9000e+04	2.0539e+04	1.2941e+03	0	8.9889e+03	4.0421e+03	160.1187

Algorithms	F-23				F-24			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	4.9896e+03	1.5714e+04	5.1378e+03	298.8665	7.5300e+03	1.1965e+05	9.5914e+03	4.7302e+03
EGWO	6.9222e+03	1.3659e+04	7.0263e+03	292.2962	6.7968e+03	1.1937e+05	8.7203e+03	8.2825e+03
MFO	4.2325e+03	1.3757e+04	4.3134e+03	464.3535	9.0983e+03	1.2916e+05	1.4181e+04	1.2958e+04
PSO	4.8551e+03	1.3030e+04	5.0519e+03	622.1611	3.2902e+03	1.4479e+05	3.6800e+03	4.2005e+03
PSOGWO	6.3798e+03	1.5310e+04	6.7003e+03	435.6941	4.8090e+03	1.5531e+05	5.8210e+03	4.0426e+03
SCA	0	1.2419e+04	6.7550e+03	448.5384	0	1.4587e+05	4.9684e+04	3.9826e+04
SSA	0	9.7776e+03	5.0219e+03	1.0635e+03	0	6.6087e+04	1.7156e+04	1.7948e+04
HSSAPSO	0	1.9747e+04	4.0117e+03	204.7735	0	1.8668e+05	3.1203e+03	2.2825e+03

Algorithms	F-25				F-26			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	2.1171e+04	1.1593e+05	2.2837e+04	3.0597e+03	4.2901e+03	1.4910e+04	4.7726e+03	615.7574
EGWO	3.7167e+04	8.9166e+04	3.7972e+04	2.1775e+03	6.2353e+03	1.7979e+04	6.3636e+03	360.3580
MFO	2.4304e+04	1.0381e+05	2.4786e+04	3.5126e+03	3.9943e+03	1.6516e+04	4.0724e+03	655.8559
PSO	2.0327e+04	9.7396e+04	2.2099e+04	4.6989e+03	3.2898e+03	1.8398e+04	3.4386e+03	1.0790e+03
PSOGWO	3.1131e+04	9.0001e+04	3.2974e+04	1.7793e+03	3.6069e+03	2.0822e+04	4.1059e+03	603.6115
SCA	0	8.5300e+04	4.0761e+04	5.4104e+03	0	2.0084e+04	8.1192e+03	862.1907
SSA	0	7.0612e+04	2.3569e+04	1.0331e+04	0	1.9752e+04	5.0059e+03	1.8725e+03
HSSAPSO	0	1.8712e+05	2.0292e+04	1.0769e+03	0	2.7276e+04	3.3799e+03	299.7865

Algorithms	F-27				F-28			
	Min	Max	Mean	SD	Min	Max	Mean	SD
AGWO	9.7315e+03	7.4850e+04	1.1871e+04	3.6681e+03	1.1015e+04	5.9090e+06	1.7728e+04	1.2724e+05
EGWO	1.4508e+04	7.0191e+04	1.5309e+04	2.1319e+03	1.0545e+04	8.6608e+05	1.1546e+04	1.3864e+05
MFO	2.1546e+04	8.2466e+04	2.2058e+04	3.7733e+03	8.6222e+03	1.9189e+07	1.1041e+05	1.8743e+06
PSO	3.3871e+03	8.1274e+04	3.9296e+04	3.7198e+03	7.0723e+03	5.0436e+06	1.6695e+04	1.6472e+05
PSOGWO	3.7770e+03	7.6115e+04	1.4825e+04	7.2186e+03	7.9841e+03	1.6743e+07	2.7565e+04	2.5260e+05

Table 8 (continued)

Algorithms	F-27				F-28			
	Min	Max	Mean	SD	Min	Max	Mean	SD
SCA	0	9.2411e+04	3.5672e+04	1.2854e+04	0	1.0481e+07	3.1805e+05	5.8952e+05
SSA	0	5.4133e+04	1.4597e+04	1.4150e+04	0	2.5735e+06	3.5253e+04	1.4772e+05
HSSAPSO	0	9.3249e+04	1.0671e+04	1.1434e+03	0	2.8604e+07	1.1143e+04	8.9224e+04

Algorithms	F-29			
	Min	Max	Mean	SD
AGWO	1.9308e+04	7.7933e+07	4.4752e+05	2.0558e+06
EGWO	3.5491e+04	1.9674e+08	2.3957e+05	9.9354e+06
MFO	9.0200e+05	7.4096e+07	1.0436e+06	2.6305e+06
PSO	6.9434e+03	2.2356e+08	4.2193e+05	7.7880e+06
PSOGWO	9.2347e+05	1.6009e+08	1.7660e+06	3.8386e+06
SCA	0	5.5052e+07	1.3731e+06	1.4919e+06
SSA	0	7.9730e+07	5.2916e+05	3.7957e+06
HSSAPSO	0	2.8875e+08	1.4591e+05	1.0464e+06

thickness of the weld, length of the clamped bar, height of the bar, thickness of the weld and minimum fabrication cost of a beam of this problem. Here, the thickness of weld (y_1) remains constant in the LSA-SM, CSS, AFA, MFO, HIS, ACO and it is greater of CPSO and CDE algorithms. The length of the clamped bar (y_2) remains greater in the LSA-SM, CSS, AFA, MFO, HIS, CDE, CPSO and ABC algorithms while it is least of HSSAPSO algorithm as comparison to others. The height of the bar (y_3) remains constant in HSSAPSO, LSA-SM, CSS, AFA, MFO, HIS and ABC algorithms while it is least of CDE algorithm outperforms than the others. Although, CPSO algorithm gives an high value of the height of the bar than others. The thickness of the weld (y_4) remains constant in the all algorithms. Finally, the minimum fabrication cost of a beam of this problem has been reported in the Table 15. Hence, these experimental solutions affirmed that the proposed algorithm is able to find the least cost of a beam of this problem outperform than others. Hence, the HSSAPSO approach is more suitable and capable for welded beam design function as compared to ABC, CPSO, CDE, HIS, MFO, AFA, CSS, LSA-SM and WCMFO algorithms.

7 Rolling element bearing

The main objective of this function is to increase the reliability and service life of the bearing [76, 77]. On the basis of operating requirements, several objectives for this function can be proposed such as longest fatigue life, mode of failure contact fatigue and many others. In which have been used three different critical constants or parameters such as pitch diameter (D_m), the number of balls (Z) and the ball diameter

(D_b). During this research, the basic dynamic capacity has been taken as objective for maximization and expression of basic dynamic capacity for ball bearing is given as:

$$\text{Max}\left[C_d(X)\right] = \begin{cases} \max\left(-f_c z^{2/3} D_b^{1.8}\right) & \text{if } D_b \leq 25.5 \text{ mm} \\ \max\left(-3.647 \times f_c z^{2/3} D_b^{1.4}\right) & \text{if } D_b \leq 25.5 \text{ mm} \end{cases}$$

Subject to:

$$g_1(x) = \frac{\phi_o}{2 \sin^{-1}\left(D_b/D_m\right)} - Z + 1 \geq 0, \tag{17}$$

$$g_2(x) = 2D_b - K_{Dmin}(D - d) \geq 0 \tag{18}$$

$$g_3(x) = K_{Dmax}(D - d) - 2D_b \geq 0 \tag{19}$$

$$g_4(x) = \xi B_w - D_b \geq 0 \tag{20}$$

$$g_5(x) = D_m - 0.5(D + d) \geq 0 \tag{21}$$

$$g_6(x) = (0.5 + e)(D + d) - D_m \geq 0 \tag{22}$$

$$g_7(x) = 0.5\left(D - D_m - D_b\right) - \xi D_b \geq 0 \tag{23}$$

$$g_8(x) = f_1 \geq 0.515 \tag{24}$$

$$g_9(x) = f_o \geq 0.515 \tag{25}$$

where

Fig. 6 Convergence graphs of algorithms on CEC 2017 test functions at 100 dimensions and 5000 iterations
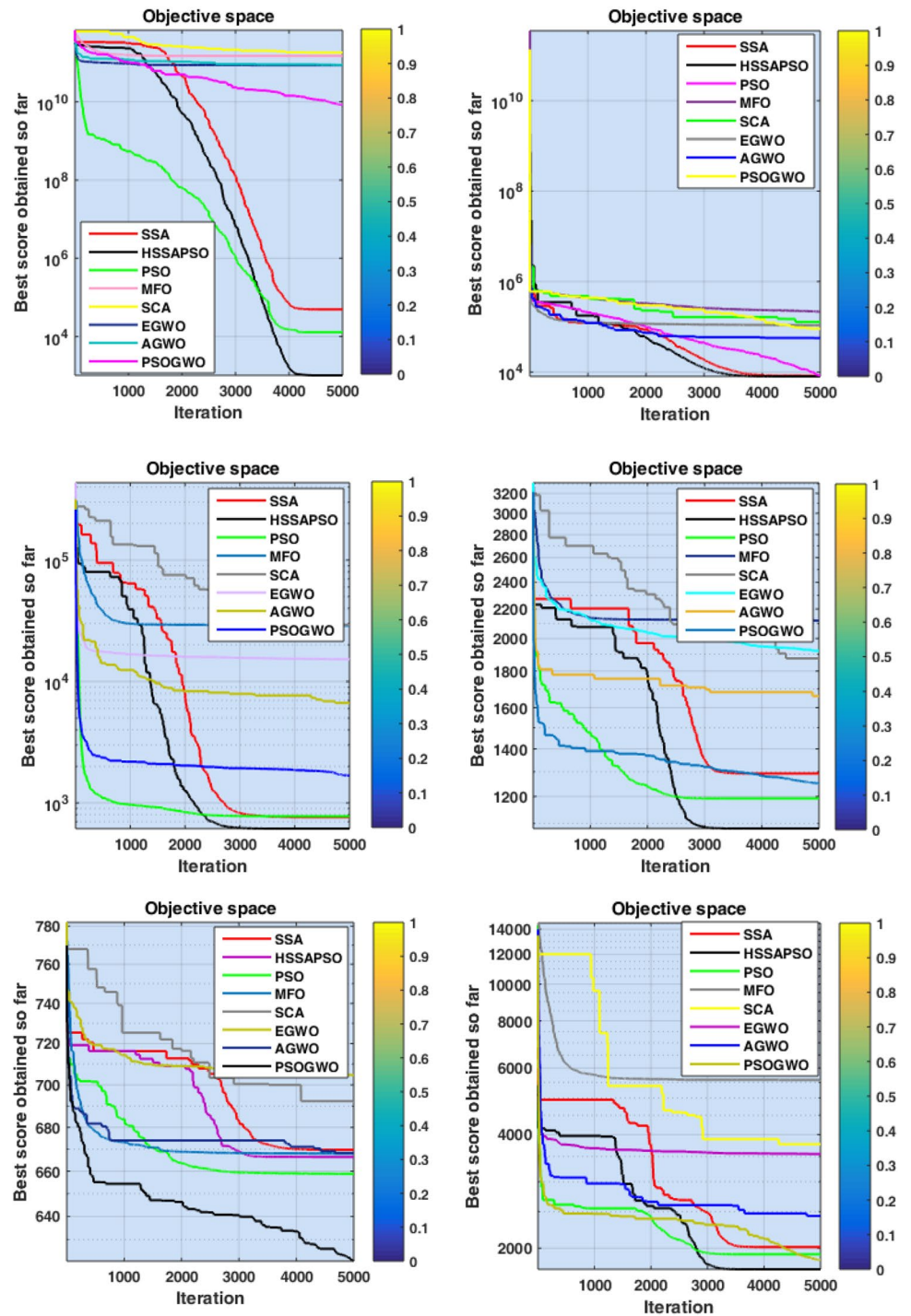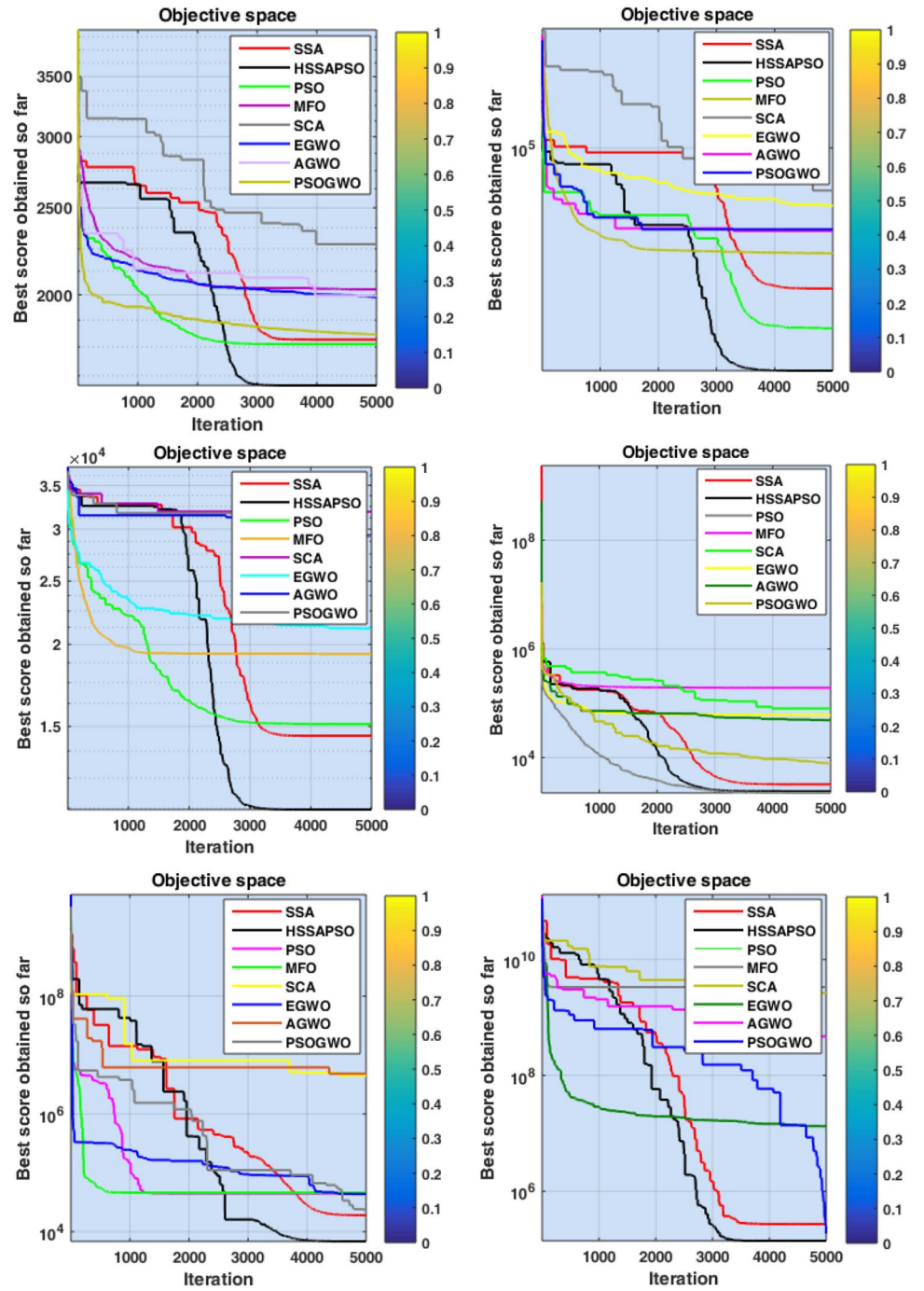
Fig. 6 (continued)

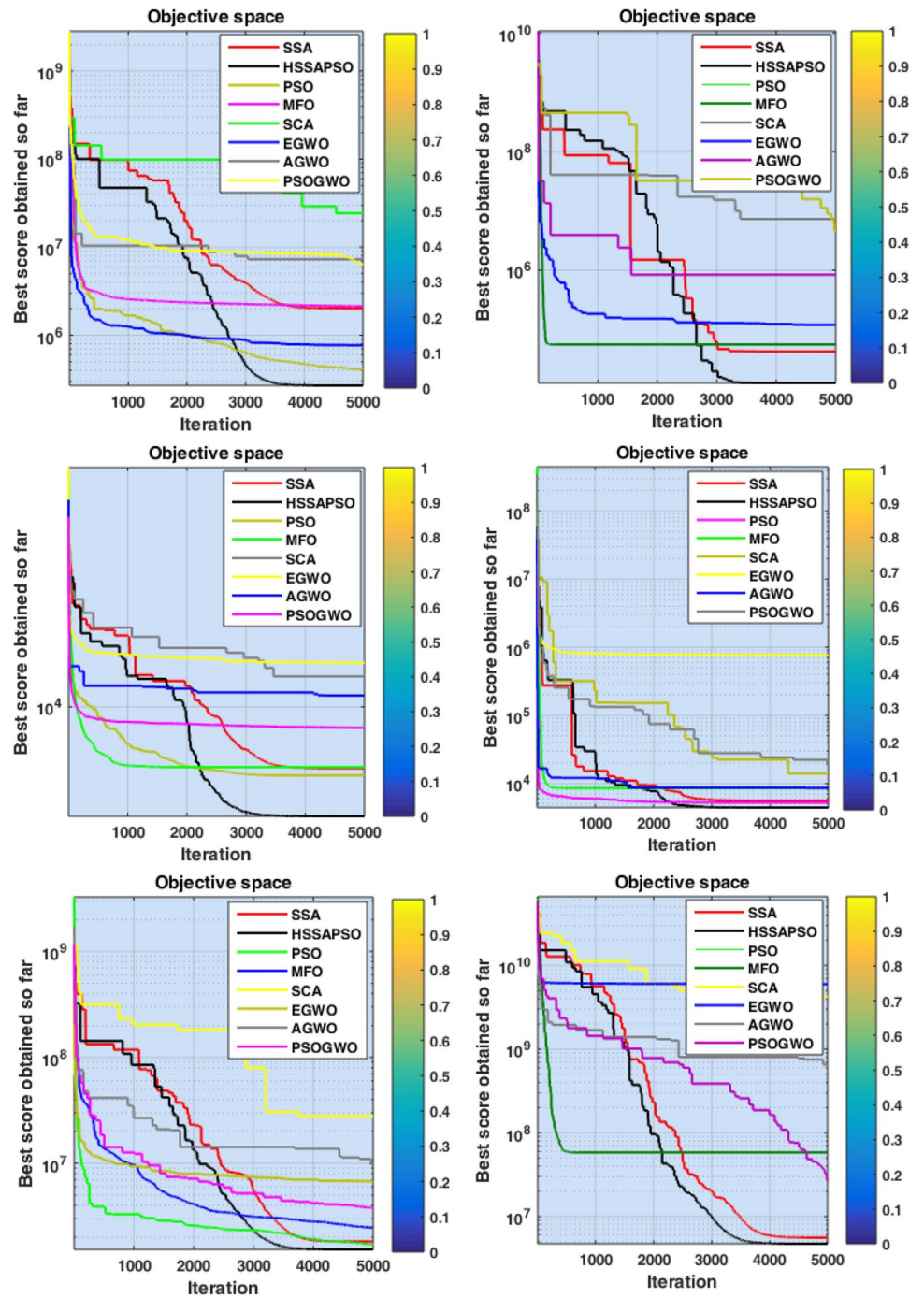**Fig. 6** (continued)

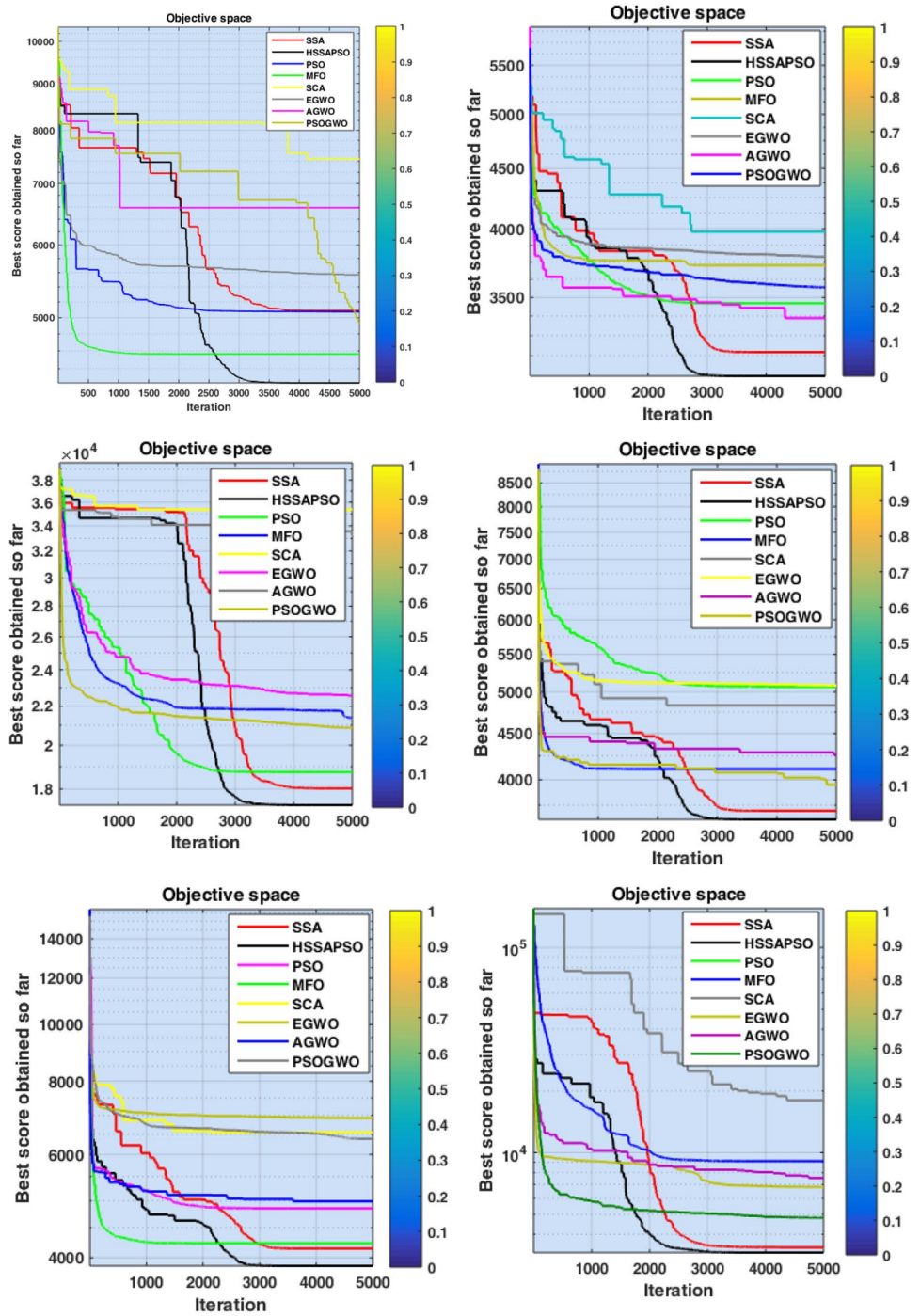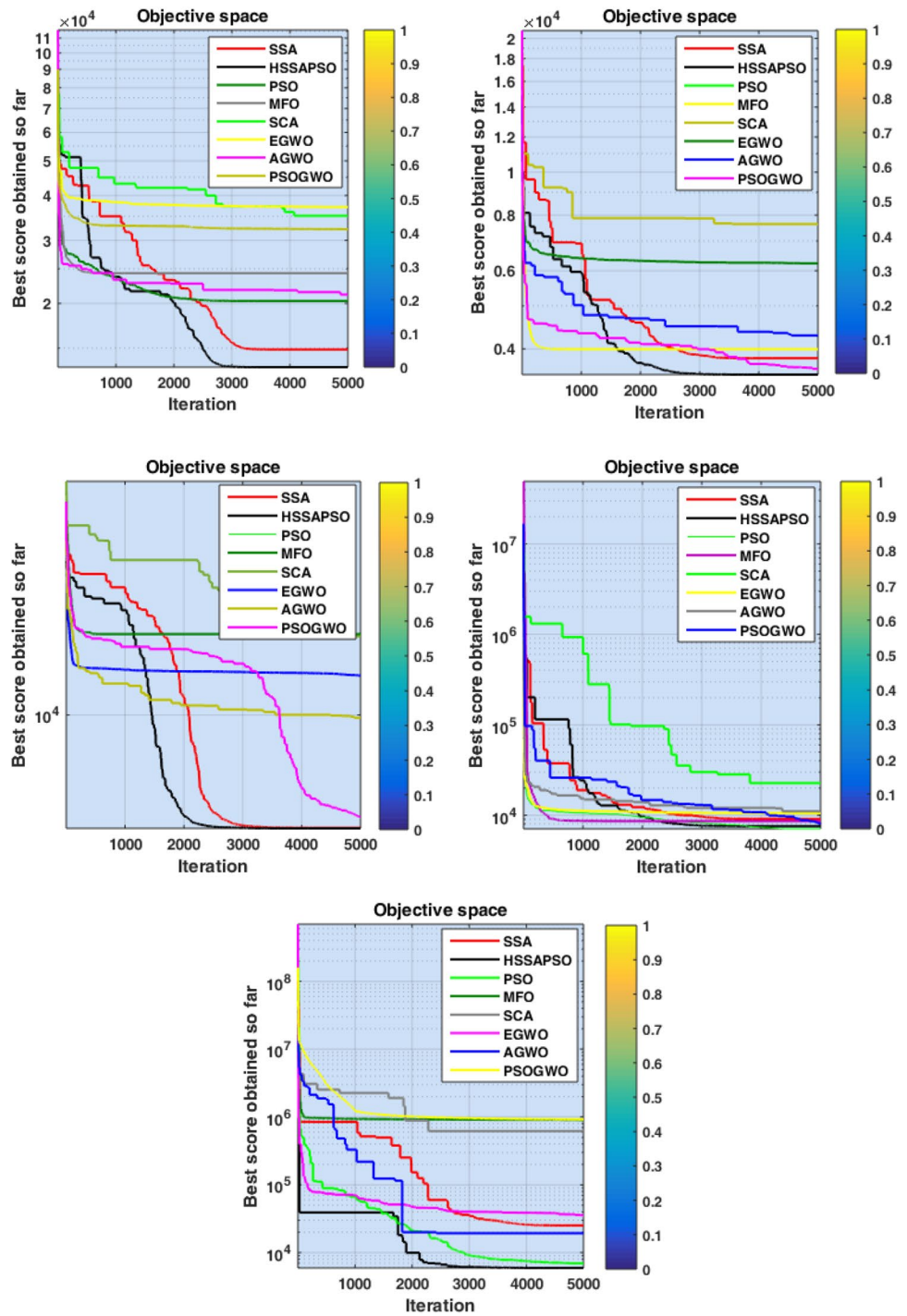**Fig. 6** (continued)

Fig. 6 (continued)

$$f_c = 37.91 \left\{ 1 + \left[1.04 \left(\frac{1-\gamma}{1+\gamma} \right)^{1.72} \left(\frac{f_i(2f_o - 1)}{f_0(2f_i - 1)} \right)^{0.41} \right]^{10/3} \right\}^{-0.3} \times \left[\frac{\gamma^{0.3}(1-\gamma)^{1.39}}{(1+\gamma)^{1/3}} \right] \left[\frac{2f_i}{2f_i - 1} \right]^{0.41} \quad (26)$$

Table 9 Median values of the algorithms on seven uni-modal problems

Problem no's	Algorithms					
	DA	MFO	PSO	SCA	SSA	HSSAPSO
	A_i	B_i	C_i	D_i	E_i	F_i
1.	2.4172e+03	1.3327e+03	5.0456	4.2684	270.3903	1.6154e−09
2.	0.6166	0.0025	0.0529	1.5677e−05	0.9594	6.5649e−06
3.	568.8904	67.4091	0.2664	1.0420	47.8683	5.3484e−10
4.	1.5306	5.841	0.1453	0.0251	2.7582	7.2903e−06
5.	1.6044e+03	903.1350	24.9223	55.5528	3.1845e+03	8.9597
6.	49.1651	65.3184	0.7598	0.9620	0.4365	0.7684
7.	0.0053	0.0110	0.0432	0.0054	0.0116	2.9255e−05

Table 10 Median values of the algorithms on sixmulti-modal problems

Problem no's	Algorithms					
	DA	MFO	PSO	SCA	SSA	HSSAPSO
	A_i	B_i	C_i	D_i	E_i	F_i
1.	−2.06e+03	−3.21e+01	−2.01e+03	−1.83e+03	−2.36e+03	−2.4866e+03
2.	36.809	12.9349	25.6489	6.4280e−08	21.6504	1.4554e−10
3.	3.9293	2.3170	0.0450	0.378	2.2106	9.4262e−06
4.	16.5049	0.1034	0.1527	0.8462	1.0706	1.6008e−09
5.	8.2092	2.5743	0.0105	5.4691	2.5248	0.2143
6.	10.4227	2.4161	0.0773	0.5830	0.4377	0.1404
7.						

Table 11 Median values of the algorithms on ninefixed dimension multi−modal problems

Problem no's	Algorithms					
	DA	MFO	PSO	SCA	SSA	HSSAPSO
	A_i	B_i	C_i	D_i	E_i	F_i
1.	8.8787	2.4808	1.9921	3.0386	6.9063	0.99805
2.	0.0023	7.8321e−04	0.0011	0.0013	0.0011	4.3092e−04
3.	−1.0252	−1.0316	−1.0316	−1.0312	−1.0309	−1.0315
4.	3.0155	3.0075	3.0282	3.0194	3.0185	3.0001
5.	−3.6349	−3.8624	−3.8595	−3.6657	−3.7523	−3.8606
6.	−3.3182	−3.2031	−3.3220	−2.9484	−3.1693	−3.3288
7.	−3.8130	−1.9801	−2.2929	−3.9838	−2.5998	−4.9648
8.	−4.4923	−5.1245	−3.5971	−3.0913	−7.8224	−10.0113
9.	−1.6132	−2.3891	−1.8648	−0.9710	−3.4059	−5.1091

$$\gamma = \frac{D_b \cos\alpha}{D_m} \tag{27}$$

$$T = D - d - 2D_b \tag{30}$$

$$f_1 = \frac{r_1}{D_b} \tag{28}$$

$$D = 160, \quad d = 90, \quad B_w = 30 \tag{31}$$

$$\phi_o = 2\pi - 20\cos^{-1}\frac{\left[\left\{(D-d)/2 - 3(T/4)^2 + \left\{D/2 - (T/4 - D_b)\right\}^2 - \left\{d/2 + (T/4)^2\right\}\right\}\right]}{2\{D - d/2 - 3(T/4)\}\{D/2 - (T/4) - D_b\}} \tag{29}$$

Table 12 Wilcoxon method for the comparison of solutions in Table 8

Proposed method	Compared algorithms	Sum of rank $(\sum_{i=1}^{7} R_i^-)$	Sum of rank $(\sum_{i=1}^{7} R_i^+)$	Difference (D)	z-value	p value	Accept $(p < 0.05)$ H_1	Reject $(p < 0.05)$ H_0
HSSAPSO algorithm	DA	28	0	$A_i - B_i$ to F_i	2.366432	.017962	Yes	Yes
	MFO	28	0		2.366432	.017962	Yes	Yes
	PSO	23	5		2.197401	.027992	Yes	Yes
	SCA	28	0		2.366432	.017962	Yes	Yes
	SSA	26	2		2.02837	.04253	Yes	Yes

Table 13 Wilcoxon method for the comparison of solutions in Table 9

Proposed method	Compared algorithms	Sum of rank $(\sum_{i=1}^{7} R_i^-)$	Sum of rank $(\sum_{i=1}^{7} R_i^+)$	Difference (D)	z-value	p value	Accept $(p < 0.05)$ H_1	Reject $(p < 0.05)$ H_0
HSSAPSO algorithm	DA	21	0	$A_i - B_i$ to F_i	2.201398	.027715	Yes	Yes
	MFO	21	0		2.201398	.027715	Yes	Yes
	PSO	15	6		0.943456	.345476	No	No
	SCA	21	0		2.201398	.027715	Yes	Yes
	SSA	21	0		2.201398	.027715	Yes	Yes

Table 14 Wilcoxon method for the comparison of solutions in Table 10

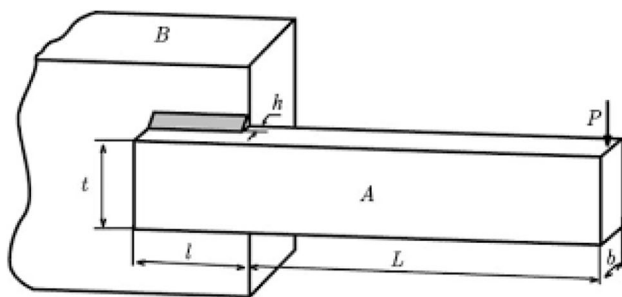| Proposed method	Compared Algorithms	Sum of rank $(\sum_{i=1}^{7} R_i^-)$	Sum of rank $(\sum_{i=1}^{7} R_i^+)$	Difference (D)	z-value	p value	Accept $(p < 0.05)$ H_1	Reject $(p < 0.05)$ H_0
HSSAPSO algorithm	DA	45	0	$A_i - B_i$ to F_i	-2.66557	.007699	Yes	Yes
	MFO	41	4		-2.19169	.028452	Yes	Yes
	PSO	44	1		-2.5471	.010865	Yes	Yes
	SCA	45	0		-2.66557	.007699	Yes	Yes
	SSA	45	0		-2.66557	.007699	Yes	Yes

Fig. 7 Weld Beam Structure

$$0.5(D + d) \leq D_m \leq 0.6(D + d) \tag{32}$$

$$0.15(D - d) \leq D_b \leq 0.45(D - d) \tag{33}$$

$4 \leq Z \leq 50; \quad 0.515 \leq f_1 \leq 0.6; \quad 0.515 \leq f_o \leq 0.6;$
$0.4 \leq K_{Dmin} \leq 0.5; \quad 0.6 \leq K_{Dmax} \leq 0.7$
$0.3 \leq \varepsilon \leq 0.4; \quad 0.02 \leq e \leq 0.1; \quad 0.6 \leq \xi \leq 0.85.$

where (d) is the bearing bore, (D) is the outside diameter (see Fig. 9, $\left(f_i = {}^{r_i}/D_b\right)$ is the curvature radius coefficient of the inner raceway groove, $\left(f_o = {}^{r_o}/D_b\right)$ is the curvature radius coefficient of the outer raceway groove, the outer (r_o) and

Fig. 8 Welded beam design function: **a** schematic of the weld; **b** stress distribution evaluated at the optimum design; **c** displacement distribution at the optimum design
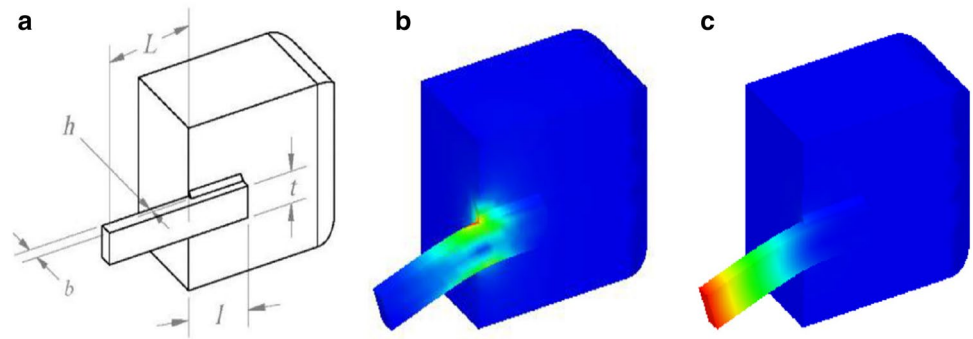

Table 15 Best optimal solutions of the welded beam design function by different algorithms

Algorithm	y_1	y_2	y_3	y_4	$f(Y)$
ABC [64]	2.06e−01	3.47e+00	9.04e+00	2.06e−01	1.72e+00
CPSO [65]	2.02e−01	3.54e+00	9.05e+00	2.06e−01	1.73e+00
CDE [66]	2.03e−01	3.54e+00	9.03e+00	2.06e−01	1.73e+00
HIS [68]	2.06e−01	3.47e+00	9.04e+00	2.06e−01	1.72e+00
MFO [29]	2.06e−01	3.47e+00	9.04e+00	2.06e−01	1.72e+00
AFA [69]	2.06e−01	3.47e+00	9.04e+00	2.06e−01	1.72e+00
CSS [70]	2.06e−01	3.47e+00	9.04e+00	2.06e−01	1.72e+00
LSA-SM [67]	2.06e−01	3.25e+00	9.04e+00	2.06e−01	1.70e+00
WCMFO [75]	0.206711	3.449553	9.03679	0.205731	1.723583
HSSAPSO	0.206711	3.440553	9.00179	0.202731	1.693651489
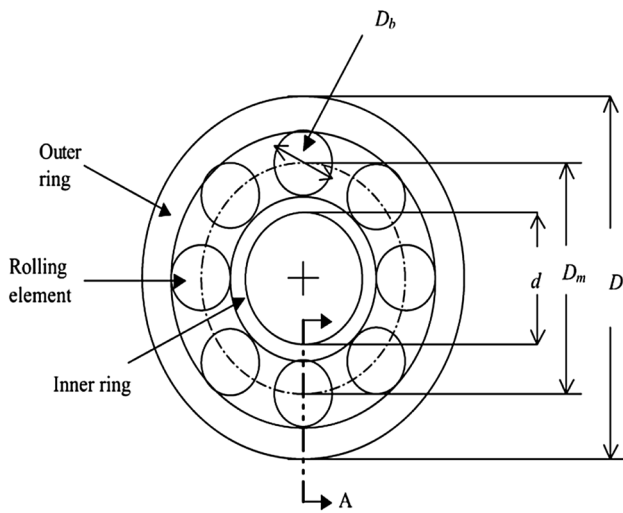
Fig. 9 Rolling Element Bearing macro-geometry
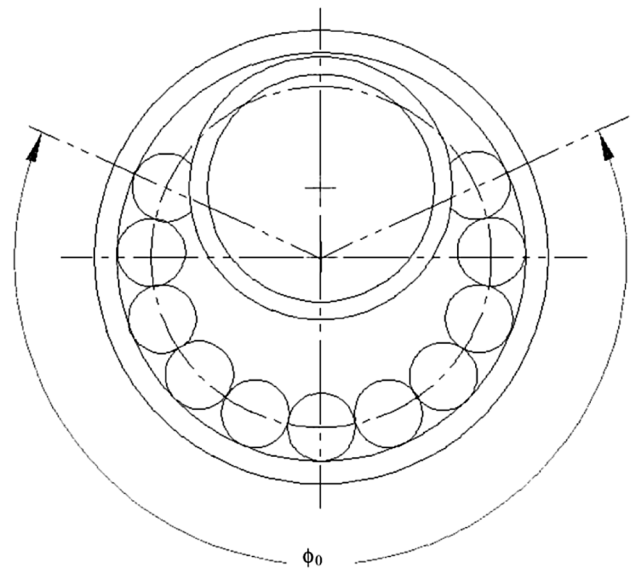
Fig. 10 Ball bearing showing the assembly angle

inner (r_i) (see Fig. 11) raceway groove curvature radius, (K_{Dmax}) is the maximum roller diameter limiter (0.8), (K_{Dmin}) is minimum roller diameter limiter (0.5), (e) is the parameter for mobility condition (0.1), (ε) is the parameter for outer ring strength (0.1) and (ϕ_o) is the maximum tolerable assembly angle (see Fig. 10) which depends upon the bearing geometry (Fig. 11).

For the analysis of the performance of the algorithms usually consider three parameters such as pitch diameter, ball diameter and the number of balls for reasons of complexity. But during this research have been applied five design parameters (D_m, D_b, Z, f_i, f_o). In the subject
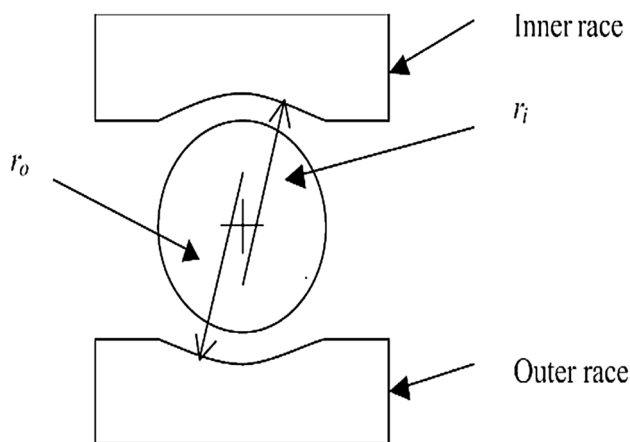
Fig. 11 Cut sections of bearing races

to the constraints, the constants values have been used $K_{Dmax} = 0.8$, $K_{Dmin} = 0.5$, $e = 0.1$, $\varepsilon = 0.1$ and $\phi_o = 4.7124$ rads which are obtained from considerations of the rings, strength of the ball and mobility of balls. In the Matlab code of the proposed algorithm have been used the parameters values such as; number of search agents (30) and maximum iterations (500). The performance of the proposed approach have been compared with the literature results of the SSA, SCA, GWO, GSA, MVO, PSO, EPO, SHO and ESA algorithms [78]. The simulation solutions of the all algorithms have presented through the Table 16.

The simulation results of Table 10 reveals that the proposed method able to search the best corresponding (max) value ($C_d(X)$) for this function outperforms than others. Here we can say that, on the basis of solutions of the proposed method, it can be improved and enhanced the life of the bearings.

8 Conclusion and future works

For the purpose of improving the exploration and exploitation of the algorithm, we have developing the newly hybrid approach called HSSAPSO in this article. This approach integrates the advantages of the salp swarm algorithm and particle swarm optimization algorithm to eliminate disadvantages, like the trapping in local optima and unbalanced exploitation. Welded beam design function, Rolling element bearing, twenty-two CEC 2005and twenty-nine CEC 2017 functions have been used for verifying the performance of the newly approach. The results and convergence graphs of these functions, proves that the HSSAPSO algorithm is a capable and faster algorithm for these functions outperform than recent metaheuristics.

Watchful evidences will determine the following advantages of the HSSAPSO algorithm.

(1) It can powerfully improve the examining capabilities of the exploration by introducing the PSO.
(2) It can recognize the accurate position HSSAPSO by tuning the constant around target.
(3) It can find the superior possible global optima solution in the search areas for the practical and real life functions.
(4) It can enhanced the convergence purity and extend the performance through reduction the computational time.

Future studies will investigate a new method with the help of statistical model to accelerate the speed of recent meta-heuristics as well as apply it for solving the other constrained nonlinear optimization functions, big data statistical and built train applications.

Table 16 Results of rolling element bearing function on different algorithms

Algorithms	Optimum variables										Opt. cost
	D_m	D_b	Z	f_i	f_o	K_{Dmin}	K_{Dmax}	ε	e	ζ	
SSA	125	20.77562	11.01247	0.515	0.515000	0.5	0.61397	0.300000	0.05004	0.610001	82,773.982
SCA	125	21.14834	10.96928	0.515	0.515	0.5	0.7	0.3	0.02778	0.62912	83,431.117
GWO	125.6199	21.35129	10.98781	0.515	0.515	0.5	0.68807	0.300151	0.03254	0.62701	84,807.111
GSA	125	20.85417	11.14989	0.515	0.517746	0.5	0.61827	0.304068	0.02000	0.624638	82,276.941
MVO	125.6002	21.32250	10.97338	0.515	0.515000	0.5	0.68782	0.301348	0.03617	0.61061	84,491.266
PSO	125	20.75388	11.17342	0.515	0.515000	0.5	0.61503	0.300000	0.05161	0.60000	81,691.202
EPO	125	21.41890	10.94113	0.515	0.515	0.4	0.7	0.3	0.02	0.6	85,067.983
SHO	125	21.40732	10.93268	0.515	0.515	0.4	0.7	0.3	0.02	0.6	85,054.532
ESA	125	21.41750	10.94109	0.510	0.515	0.4	0.7	0.3	0.02	0.6	85,070.085
HSSAPSO	125.7227	21.4233	11.0012	0.5150	0.5150	0.4515	0.6170	0.3000	0.0244	0.6259	85,539.189

References

1. Kalaiselvi K, Kumar VS, Chandrasekar K (2010) Enhanced genetic algorithm for optimal electric power flow using TCSC and TCPS. World congress on engineering, vol II, WCE 2010, June 30–July 2, 2010, London, UK

2. Bakirtzis AG, Biskas PN, Zoumas CE, Petridis V (2002) Optimal power flow by enhanced genetic algorithm. IEEE Power Eng Rev 22(2):60

3. Chung TS, Li YZA (2000) A hybrid GA approach for OPF with consideration of FACTS devices. IEEE Power Eng Rev 20(8):54–57

4. Cai LJ, Erlich I, Stamtsis G (2004) Optimal choice and allocation of FACTS devices in deregulated electricity market using genetic algorithms. In: IEEE PES power systems conference and exposition, 10–13 Oct 2004, New York, NY, USA

5. Slimani L, Optimal BT (2012) Optimal power flow solution of the algerian electrical network using differential evolution algorithm. TELKOMNIKA 10(2):199–210

6. Simon D (2008) Biogeography-based optimization. IEEE Trans Evol Comput 12(6):702–713

7. Duman S, Guvenc U, Sonmez Y, Yorukeren N (2012) Optimal power flow using gravitational search algorithm. Energy Convers Manag 59:86–95

8. Kennedy J (2011) Particle swarm optimization. In: Sammut C, Webb GI (eds) Encyclopedia of machine learning. Springer, Boston

9. Abido MA (2002) Optimal power flow using Tabu search algorithm. Electr Power Compon Syst 30:469–483

10. Sinsupan N, Leeton U, Kulworawanichping T (2010) Application of harmony search to optimal power flow problems. In: 2010 International conference on advances in energy engineering, 19–20 June 2010, Beijing, China

11. Alrashydah EI, Qudais SAA (2018) Modeling of creep compliance behavior in asphalt mixes using multiple regression and artificial neural networks. Constr Build Mater 159:635–641

12. Mirjalili S (2015) Dragonfly algorithm: a new meta-heuristics optimization technique for solving single-objetive, discrete, and multi-objective problems. Neural Comput Appl 27:1053–1073

13. Mirjalili S, Mirjalili SM, Lewis A (2014) Grey wolf optimizer. Adv Eng Softw 69:46–61

14. Mukherjee A, Mukherjee V (2015) Solution of optimal power flow using chaotic krill herd algorithm. Chaos Solitons Fractals 78:10–21

15. Bouchekara HREH (2014) Optimal power flow using black-hole-based optimization approach. Appl Soft Comput 24:879–888

16. Tal AB, Ghaoui LEI, Nemirovski A (2009) Robust optimization. Princeton series in applied mathematics. Princeton University Press, Princeton, pp 1–576

17. Mirjalili S (2015) The ant lion optimizer. Adv Eng Softw 83:80–98

18. Singh N, Singh SB (2011) One half global best position particle swarm optimization algorithm. Int J Sci Eng Res 2(8):1–9

19. Roger JM, Chauchard F, Maurel VB (2003) EPOPLS external parameter orthogonalisation of PLS application to temperature-independent measurement of sugar content of intact fruits. Chemometr Intell Lab 66:191–204

20. Fausto F, Cuevas E, Valdivia A, González A (2017) A global optimization algorithm inspired in the behavior of selfish herds. Biosystems 160:39–55

21. Joshi H, Arora S (2017) Enhanced grey wolf optimization algorithm for constrained optimization problems. Int J Swarm Intell 3(2/3):126–151

22. Qais MH, Hasanien HM, Alghuwainem S (2018) Augmented grey wolf optimizer for grid-connected PMSG-based wind energy conversion systems. Appl Soft Comput 69:505–515

23. Singh N, Singh SB (2017) Hybrid algorithm of particle swarm optimization and grey wolf optimizer for improving convergence performance. J Appl Math 2017:1–15

24. Soares J, Sousa T, Vale ZA, Morais H, Faria P (2011) Ant colony search algorithm for the optimal power flow problem. In: 2011 IEEE power and energy society general meeting, 24–28 July 2011, Detroit, MI, USA

25. Saremi S, Mirjalili S, Lewis A (2017) Grasshopper optimisation algorithm: theory and application. Adv Eng Softw 105:30–47

26. Mirjalili S (2016) SCA: a sine cosine algorithm for solving optimization problems. Knowl Based Syst 96:120–133

27. Mirjalili S, Lewis A (2016) The whale optimization algorithm. Adv Eng Softw 95:51–67

28. Singh S, Singh SB (2017) A novel hybrid GWO-SCA approach for optimization problems. Eng Sci Technol Int J 20(6):1586–1601

29. Mirjalili S (2015) Moth-flame optimization algorithm: a novel nature-inspire heuristic paradigm. Knowl Syst 89:228–249

30. Daryani N, Hagh MT, Teimourzadeh S (2016) Adaptive group search optimization algorithm for multi-objective optimal power flow problem. Appl Soft Comput 38:1012–1024

31. Singh N, Singh S, Singh SB (2017) A new hybrid MGBPSO-GSA variant for improving function optimization solution in search space. Evolut Bioinform 13:1–13

32. Mirjalili S, Mirjalili SM, Hatamlou A (2016) Multi-verse optimizer: a nature-inspired algorithm for global optimization. Neural Comput Appl 27:495–513

33. Singh N, Singh S, Singh SB, Arora S (2012) Half mean particle swarm optimization algorithm. Int J Sci Eng Res 3(80):1–10

34. Rao RM, Babu AVN (2013) Optimal power flow using cuckoo optimization algorithm. Int J Adv Res Electr Electron Instrum Eng 2(9):1–6

35. Singh N, Singh SB (2012) Personal best position particle swarm optimization. J Appl Comput Sci Math Suceava 12(6):69–76

36. Singh N, Hachimi H (2018) A new hybrid whale optimizer algorithm with mean strategy of grey wolf optimizer for global optimization. Math Comput Appl 23(1):1–32

37. Yu S, Wu Z, Wang H, Chen Z (2010) A hybrid particle swarm optimization algorithm based on space transformation search and a modified velocity model. In: Schaeffer J (ed) High performance computing and applications. Springer, Berlin, pp 522–527

38. Liang RH, Tsai SR, Chen YT, Tseng WT (2011) Optimal power flow by a fuzzy based hybrid particle swarm optimization approach. Electr Power Syst Res 81(7):1466–1474

39. Kennedy J, Eberhart R (1995) Particle swarm optimization. In: Proceedings of the ICNN'95—international conference on neural networks. IEEE, pp 1942–1948. https://doi.org/10.1109/icnn.1995.488968

40. Kennedy J, Eberhart RC, Shi Y (2001) Swarm intelligence. Morgan Kaufmann Publishers, Burlington

41. Liu W, Wang K, Sun B, Shao K (2006) A hybrid particle swarm optimization algorithm for predicting the chaotic time series. In: 2006 international conference on mechatronics and automation, 25–28 June 2006, Luoyang, Henan, China

42. Marinke R, Araujo E, Coelho LS, Matiko L (2005) Particle swarm optimization (PSO) applied to fuzzy modeling in a thermal-vacuum system. In: Fifth international conference on hybrid intelligent system (HIS'05), 6–9 Nov 2005, Rio de Janeiro, Brazil

43. Angeline PJ (1998) Evolutionary optimization versus particle swarm optimization: philosophy and performance differences. In: Porto VW, Saravanan N, Waagen D, Eiben AE (eds) Evolutionary

programming VII. EP 1998. Lecture notes in computer science, vol 1447. Springer, Berlin

44. Juang CF (2004) A hybrid of genetic algorithm and particle swarm optimization for recurrent network design. IEEE Trans Syst Man Cybern B (Cybern) 34(2):997–1006

45. Zhang C, Shao H, Li Yu (2000) Particle swarm optimisation for evolving artificial neural network. In: SMC 2000 conference proceedings. 2000 IEEE international conference on systems, man and cybernetics. 'cybernetics evolving to systems, humans, organizations, and their complex interactions', 8–11 Oct 2000, Nashville, TN, USA

46. Esmin AAA, Torres GL, Souza ACZD (2005) A hybrid particle swarm optimization applied to loss power minimization. IEEE Trans Power Syst 20(2):859–866

47. Esmin AAA, Torres GL, Alvarenga GB (2006) Hybrid evolutionary algorithm based on PSO and GA mutation. In: 2006 sixth international conference on hybrid intelligent systems (HIS'06'), 13–15 Dec 2006, Rio de Janeiro, Brazil

48. Zhao B, Guo CX, Cao YJ (2005) A multiagent-based particle swarm optimization approach for optimal reactive power dispatch. IEEE Trans Power Syst 20:1070–1078

49. Vlachogiannis JG, Leet KY (2006) A comparative study on particle swarm optimization for optimal steady-state performance of power systems. IEEE Trans Power Syst 21(4):1718–1728

50. Huang CM, Huang CJ, Wang ML (2005) A particle swarm optimization to identifying the ARMAX model for short-term load forecasting. IEEE Trans Power Syst 20(2):1126–1133

51. Esmin AAA, Torres GL (2012) Application of particle swarm optimization to optimal power systems. Int J Innov Comput Inf Control 8(3):1705–1716

52. Esmin AAA, Torres GL (2006) Fitting fuzzy membership functions using hybrid particle swarm optimization. In: 2006 IEEE international conference on fuzzy systems, 16–21 July 2006

53. Ali AA, Tawhid MA (2017) A hybrid particle swarm optimization and genetic algorithm with population partitioning for large scale optimization problems. Ain Shams Eng J 8(2):191–206

54. Mao B, Xie Z, Wang Y, Handroos H, Wu H, Shi S (2017) A hybrid differential evolution and particle swarm optimization algorithm for numerical kinematics solution of remote maintenance manipulators. Fusion Eng Des 124:587–590

55. Hadji B, Mahdad B, Srairi K, Mancer N (2015) Multi-objective PSO-TVAC for environmental/economic dispatch problem. Energy Procedia 74:102–111

56. Zhang Y, Gong DW, Cheng J (2017) Multi-objective particle swarm optimization approach for cost-based feature selection in classification. IEEE/ACM Trans Comput Biol Bioinf 14(1):64–75

57. Song XF, Zhang Y, Guo YN, Sun XY, Wang YL (2020) Variable-size cooperative co-evolutionary particle swarm optimization for feature selection on high-dimensional data. IEEE Trans Evolut Comput. https://doi.org/10.1109/TEVC.2020.2968743

58. Zhang Y, Gong DW, Geng N, Sun XY (2014) Hybrid bare-bones PSO for dynamic economic dispatch with value-point effects. Appl Soft Comput 18:248–260

59. Zhang Y, Gong DW, Zhang J (2013) Robot path planning in uncertain environment using multi-objective particle swarm optimization. Neurocomputing 103:172–185

60. Singh N, Singh SB (2017) A modified mean gray wolf optimization approach for benchmark and biomedical problems. Evol Bioinform 13:1–28

61. Mirjalil S, Gandomi AH, Mirjalili SZ, Saremi S, Faris H, Mirjalili SM (2017) Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Adv Eng Softw 114:163–191

62. Awad N, Ali M, Liang J, Qu B, Suganthan P (2017) Problem definitions and evaluation criteria for the CEC 2017 special session and competition on single objective real-parameter numerical optimization. Technical report

63. 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 Evol Comput 1(1):3–18

64. David DCN, Stephen CEA (2018) Cost Minimization of Welded Beam Design Problem using Non-traditional optimization through Matlab and validation through analyses simulation. Int J Mech Eng Technol IJMET 9(8):180–192

65. Mosavi A, Vaezipour A (2012) Reactive search optimization: application to multi-objective optimization problems. Appl Math 3:1572–1582

66. Li HS, Au SK (2010) Solving constrained optimization problems via subset simulation. In: 4th international workshop on reliable engineering computing, pp 439–453

67. Coello CAC (2000) Use of a self-adaptive penalty approach for engineering optimization problems. Comput Ind 41:113–127

68. Akay B, Karaboga D (2012) Artificial bee colony algorithm for large-scale problems and engineering design optimization. J Intell Manuf 23(4):1001–1014

69. He Q, Wang L (2007) An effective co-evolutionary particle swarm optimization for constrained engineering design problems. Eng Appl Artif Intell 20(1):89–99

70. Huang FZ, Wang L (2007) An effective co-evolutionary differential evolution for constrained optimization. Appl Math Comput 186(1):340–356

71. Mahdavi M, Fesangharg M, Damangir E (2007) An improved harmony search algorithm for solving optimization problems. Appl Math Comput 188(2):1567–1579

72. Baykasoğlu A, Ozsoydan FB (2015) Adaptive firefly algorithm with chaos for mechanical design optimization problems. Appl Soft Comput 36:152–164

73. Kaveh A, Talatahari S (2010) A novel heuristic optimization method: charged system search. Acta Mech 213:267–289

74. Lu Y, Zhou Y, Wu X (2017) A hybrid lightning search algorithm-simplex method for global optimization. Discrete Dyn Nat Soc 8342694:1–23

75. Khalilpourazari S, Khalilpourazary S (2019) An efficient hybrid algorithm based on water cycle and moth-flame optimization algorithms for solving numerical and constrained engineering optimization problems. Soft Comput 23:1699–1722. https://doi.org/10.1007/s00500-017-2894-y

76. Kumar A (2016) Ball bearing design through Jaya algorithm. Int J Adv Res Sci Eng 5(12):458–467

77. Chakraborty I, Kumar V, Nair SB (2003) Rolling element bearing design through genetic algoirthms. Eng Optim 35(6):1–26

78. Dhiman G (2019) ESA: a hybrid bio-inspired metaheuristic optimization approach for engineering problems. Eng Comput. https://doi.org/10.1007/s00366-019-00826-w

79. Yuan X, Dai X, Zhao J, He Q (2014) On a novel multi-swarm fruit fly optimization algorithm and its application. Appl Math Comput 233:260–271

80. Mosavi A, Vaezipour A (2012) Reactive search optimization; application to multiobjective optimization problems. Appl Math 3(10):1572–1582