A New Fusion of Salp Swarm with Sine Cosine for Optimization of Non-linear Functions

Narinder Singh[1], Le Hoang Son [2*], Francisco Chiclana [3], Jean-Pierre Magnot[4]

[1]Department of Mathematics, Punjabi University, Patiala-147002, Punjab, India
narindersinghgoria@ymail.com
[2] VNU Information Technology Institute, Vietnam National University, Hanoi, Vietnam
sonlh@vnu.edu.vn
[3] School of Computer Science and Informatics, De Montfort University, The Gateway, Leicester, LE1 9BH UK
chiclana@dmu.ac.uk
[4]LAREMA, universitéd' Angers, 2 boulevard Lavoisier49045 Angers cedex 01, France
jean-pierr.magnot@ac-clermont.fr

*: Corresponding author. Tel.: (+84) 904.171.284. Address: 144 Xuan Thuy, Cau Giay, Hanoi, Vietnam

Abstract: The foremost objective of this article is to develop a novel hybrid powerful meta-heuristic that integrates the Salp Swarm Algorithm with Sine Cosine Algorithm (called HSSASCA) for improving the convergence performance with the exploration and exploitation being superior to other comparative standard algorithms. In this method, the position of salp swarm in the search space is updated by using the position equations of sine cosine; hence the best and possible optimal solutions are obtained based on the sine or cosine function. During this process, each salp adopts the information sharing strategy of sine and cosine functions to improve their exploration and exploitation ability. The inspiration behind incorporating changes in Salp Swarm Optimizer Algorithm is to assist the basic approach to avoid premature convergence and to rapidly guide the search towards the probable search space. The algorithm is validated on twenty-two standard mathematical optimization functions and three applications namely the three-bar truss, tension/compression spring and cantilever beam design problems. The aim is to examine and confirm the valuable behaviors of HSSASCA in searching the best solutions for optimization functions. The experimental results reveal that HSSASCA algorithm achieves the highest accuracies with least runtime in comparison with the others.

Keywords: Standard global Optimization Functions; Heuristic Hybridization; Salp Swarm Algorithm; Sine Cosine Algorithm; Exploration and Exploitation.

1. INTRODUCTION

Nature inspired techniques are powerful and well-known for searching optimal solutions in optimization problems. Day by day, researchers have developed several newly meta-heuristics for improving and enhancing exploration and exploitation of the existing algorithms, for instance Gravitational Search Algorithm (GSA) [15], Grey Wolf Optimization (GWO) [30], Particle-Swarm-Optimization (PSO) [23], Ant-Colony-Optimization- (ACO) [3], Krill Herd Algorithm (KHA) [37], Ant Lion Optimizer (ALO) [36] and many others [1, 4, 5, 7, 8, 11-13, 17, 21, 22, 25, 29, 31-35, 38-39, 42, 44-52, 53-54, 58, 61, 84]. Each nature inspired algorithm has its own advantages and disadvantages so that there is no guarantee which algorithm is best suited for a specific problem [57]. It is possible that the single optimizer algorithm cannot find the best solution for each type of functions [57]. Therefore, implementing and proposing new and high-accuracy meta-heuristics for real applications have become a challenging task for scientists [9].

Hybridization of nature inspired algorithms is a popular approach for to merge merits and strength of standalone algorithms for handling those deficiencies [9]. Several typical studies can be seen in [21, 38, 42, 44, 47, 50,61, 82, 83] in which the hybrid algorithms merging advantages of single ones performed well in boosting the accuracy of functions and reducing classification time. As an example, Sarbazfard et al. [42] developed a hybrid variant called HFADE that integrates differential evolution (DE) with Firefly algorithm (FA) for improving exploration tendency of those algorithms. Firefly algorithm and differential evolution both are effective techniques but firefly approach depends on arbitrary instructions for hunt, which lead into retardation in searching the superior and possible global result in the search area. The existing variant was utilized on twenty-six standard functions for testing the convergence accuracy. Fouad [17] recently proposed a hybrid approach called Hybrid GWO-GA, amid the grey wolf optimizer (GWO) and genetic algorithm (GA) in order to minimize a simplified model of the energy function of the molecule. In this study, GWO was applied to create the equilibrium amid exploration and exploitation in the existing variant. The experiments revealed that the existing approach is more competent, capable and promising of searching nearest global optima minimum value of the standard problem than the others.

However, there are several meta-heuristics applied in real-world problems and no algorithm can solve all types of functions [2]. In this paper, we consider the extension of the salp swarm (SSA) [28], which is a robust algorithm in comparison with the other algorithms. It has good convergence rate, but there are still some shortcomings/demerits, like easy fall into low exploration, local optimum, poor solution accuracy, premature convergence and exploitation tendency [43]. Faris et al. [16] was presented an newly modified approach for enhance the performance of SSA algorithm. In this work two new wrapper FS algorithms that apply SSA as the search method. Firstly, eight transfer problems/or functions are employed to convert the continuous version of salp swarm algorithm to binary. And secondly, the crossover operator is applied in addition to the transfer problems/or functions enhanced the exploratory behavior of the approach. The working performance of this version have been tested on 22 standard problems and verified with several latest meta-heuristics in term of best and possible solution of functions. In order to handle these drawbacks, we propose the idea of combination between salp swarm and sine cosine algorithms (SCA) [34], which is competent for determining best solution with the exploitation and exploration being superior to other recent comparative standard algorithms [56]. Farnad and Jafarian [77] presented an efficient hybrid method for finding the solutions of engineering and constrained numerical functions. Three different algorithms such as genetic algorithm (GA), particle swarm optimization (PSO) and symbiotic organisms were integrated for finding solutions of function in a complex design space and to manage the feasibility of searching with penalty function strategy. The new algorithm was tested on the standard well-known functions and engineering applications with the recent meta-heuristics. Similarly, several recent hybrid, modified and newly evolutionary approaches have been presented by the researchers such as Hybrid Bacterial Flower Pollination Algorithm (HBFPA) [78], Flower Pollination Algorithm (FPA) [79], Hybrid whale optimization algorithm based on local search strategy [80], hybrid Q-learning sine-cosine- based strategy (QLSCA) [81], Adaptive Operator Quantum-Behaved Pigeon-Inspired Optimization Algorithm [85] and many others.

The main objective of this work is to present a hybrid salp swarm optimizer with sine cosine algorithm to solve engineering problems. This proposed method is called as hybrid SSASCA algorithm. Although the salp swarm algorithm is more capable to reveal a competent accuracy in comparison with other well-known meta-heuristics, it still may face the difficulty of getting trapped in local optima. It is also not fitting for high complex functions and cannot handle several their drawbacks such as premature convergence, slow diversity, slow convergence speed etc. Hence, in order to improve the slow convergence and other weakness of the salp swarm approach, SCA is invoked as a local search scheme. The proposed method transits from (exploration to exploitation) the search of solved with the use of optimal range in the trigonometry functions. Therefore, HSSASCA algorithm produces and refines a set of random optimal goals for the given functions and furthermore it intrinsically advantages from the local optima avoidance and high exploration compared to separate based meta-heuristics. Our methodology enhances search capabilities and global convergence rate by accelerating the search speed. The modified method has been tested on several well-known standard benchmark functions and engineering applications in the comparison with the related algorithms. All numerical and statistical optimal solutions of the functions reveal that the proposed method outperforms the others for searching the best value of the functions.

The remains are below: Section 2 describes the background of Salp Swarm and Sine Cosine. The motivation of the present work has been reported in the section 3. Section 4 shows details of the newly hybrid approach. Analysis and comparative experiments are described in Sections 5-6. In Section 7, three applications namely the three-bar truss, tension/compression spring and cantilever beam design problems are presented. Finally, Section 8 presents the concluding remarks and future studies.

2. BACKGROUND

2.1 Salp Swarm Algorithm (SSA)

Mirazalii et al. [28] introduced an extensive accessibility such as SSA, inspired from the navigation and foraging behavior of salp deep in the sea. These organisms attach roots and make a root or a slip chain. The salp chain tries to find the best place of food via process of searching with the help of a leader salp, as the rest of the followers. The crowd of salp swarm optimizer algorithm is initialized in two different groups like followers and leader. The first group is one salp taking the position at the front of the sequence. Let y denoted as position of a salp, and F represents food. Position of leader is updated as,

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

where x_j^1 is the position of the best solution, ub_j and lb_j are upper and lower bound of the j^{th} dimension, F_j is the food source position of the dimension and c_2, c_3 are random numbers,

Here, c_1 is an important constant that it maintains a balance amid exploitation and exploration, and it can be written/or given by:

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

where L is the maximum number of iteration and l is the current iteration.

The follower's positions are updated by using the following mathematical equation:

$$x_j^i = \frac{1}{2}at^2 + v_0 t \tag{3}$$

where $i \geq 2, x_i^j$ presents the position of i^{th} followers salp in j^{th} dimension, t is time, v_0 is the initial speed and $a = \frac{v_{final}}{v_0}$ where $v = \frac{x - x_0}{t}$.

Hence, the time in optimization in generation or iteration, the discrepancy between generations or iterations is equal to 1, and considering $v_0 = 0$, this equation can be expressed as follows:

$$x_j^i = \frac{1}{2}\left(x_j^i + x_j^{i-1}\right) \quad (4)$$

where x_j^i is the position of the i^{th} follower at the j^{th} dimension.

2.2 Sine Cosine Algorithm (SCA)

SCA [34] establishes various basic random agent solutions based on sine-cosine functions towards the best global optima. Main step of an optimizer is known to be the formulation of the position updating. Subsequent position of an agent is modified by:

$$\vec{x}_i^{t+1} = \vec{x}_i^t + r_1 \times \sin\left(r_2\right) \times \left|r_3 \times l_i^t - \vec{x}_i^t\right| \quad (5)$$

$$\vec{x}_i^{t+1} = \vec{x}_i^t + r_1 \times \cos\left(r_2\right) \times \left|r_3 \times l_i^t - \vec{x}_i^t\right| \quad (6)$$

The conditions in equations (4-5) for exploitation and exploration are:

$$\vec{x}_i^{t+1} = \begin{cases} \vec{x}_i^t + r_1 \times \sin\left(r_2\right) \times \left|r_3 \times l_i^t - \vec{x}_i^t\right| & , r_4 < 0.5 \\ \vec{x}_i^t + r_1 \times \cos\left(r_2\right) \times \left|r_3 \times l_i^t - \vec{x}_i^t\right| & , r_4 \geq 0.5 \end{cases} \quad (7)$$

where $\vec{x}_i^t$ is the current position at t^{th} iteration in i^{th} dimension, l_i is the targeted global optimal solution, r_1, r_2, r_3 are random numbers and $|\;|$ is the absolute value.

In order to get a balance amid exploitation and exploration, the first random value is chosen adaptively as follows.

$$r_1 = \text{constant} - present\_iter \times \frac{\text{constant}}{Max\_iter} \quad (8)$$

The first random value $\left(r_1\right)$ controls the new update position's region. The second random value $\left(r_2\right)$ decides the distance outwards or towards the destination. The third random value $\left(r_3\right)$ generates a random weight to stochastically deemphasize $\left(r_3 < 1\right)$ or emphasize $\left(r_3 > 1\right)$ effect of destination in defining the distance. The fourth random value is in [0,1] and uniformly switches amid the cosine and sine position updating.

3. Motivation of the present work

Although SSA is skilled to conceal well-organized accuracy in comparison with recent meta-heuristics, it is still may face the difficulty of getting trapped in local optima and is not fit for highly complex functions. To extent its search ability and overcome these limitations, a newly hybrid method called hybrid salp swarm optimizer and sine cosine algorithms (HSSASCA) algorithm is developed to solve engineering problems. During this work, SSA operates in the direction of exploring the vector of solutions while SCA is invoked as a local search scheme to improve the solution superiority. The natural characteristic of SCA algorithm to make compound mutation in the optimal solutions and to avoid to stuck in local optima. By this methodology, it is intended to improve the global convergence by accelerating the search seeking instead of letting the algorithm running several iterations without any improvement. The accuracy of the proposed method has been tested on various standard well-known benchmark and engineering functions. Experimental results reveal that the proposed approach is a robust search method for several optimization functions.

4. THE PROPOSED HSSASCA ALGORITHM

Researchers have been trying for developing new hybrid and modified version of the exiting algorithms for different specific complex functions of optimization problems. As per Talbi [86], two different algorithms can be hybridized in two ways such as low level versus high level and relay versus teamwork (sub-categories (i) low-level relay hybrid (LRH) and (ii) low-level teamwork hybrid (LTH) with co-evolutionary techniques as homogeneous. During this research, we hybridize the salp swarm algorithm with sine cosine algorithm using low-level teamwork hybrid (LTH) co-evolutionary mixed hybrid. Further, the main structure of the proposed hybrid method is explained. It is known as HSSASCA, which merges the Salp Swarm Algorithm (SSA) and the Sine Cosine Algorithm (SCA). The main part of the Salp Swarm Algorithm is modified by improving the updating phase of the population's position. In this modification the sine and cosine functions have been applied in the position update equation in SSA algorithm for enhancing the exploration and exploitation tendency of the algorithm. This integration adds more flexibility to the Salp Swarm Algorithm (SSA) in exploring the crowd/or population and ensures the diversity of it, as well as the appropriate value reaches quickly.

Further, during this study, a modified approach of hybrid SSA and SCA is incorporated in a parallel manner with the objective to replace bad optimal solutions via the one-to-one idea to find new crowd/or population. The main motive of this work is that the help of salp swarm can be improved exploitation tendency and exploration can be achieved with the help of sine and cosine. The proposed method uses trigonometry functions (i.e. sine and cosine) to search and exploit space between two solutions in the search area for finding a better optimal solution. The agent/or salp population and fitness value of the given function has been evaluated as per the newly hybrid method. Furthermore, the position of the each salp swarm in the entire group is improved by applying the position equations of sine and cosine functions. For this reason, better quality of global optimal results/or solutions have been tried to update based on these functions, which means that the exploration ability could be much stronger. The sine and cosine functions can more helps the Salp Swarm Optimization algorithm phase to attained the best solution/ or score more rapidly and improve the convergence rate.

Through that methodology, the natural properties of this improvement can be controlled by involving the SCA phase as a local research strategy which accelerates the behavior of the desire and prevents the system of metabolic modification without any modifications in the results. Indeed, the inefficiency of the SSA phase can be reduced efficiently. Here, the proposed algorithm proceeds to find the best and possible results in the search areas. Further, brief details of the newly hybrid approach is shown step by step as below:

Step 1: Initialization population

During this study, firstly we initialize the population in the search area. The crowds of salp are initialized randomly within the search area of the given functions, where the meta-heuristic assigns a random vector of n dimensional for the i^{th} salp; $X = x_i \sim (i = 1, 2, 3, ..., n)$.

Step 2: Evaluation

Every search member is evaluated according to the superiority of its position/or location which is allied to the preferred objective problem/or function, where the best solution (or goal) so far is obtained.

Step 3: Each agent locations/or positions updating

The following equation (9) affirmed that the leader only updates its position or location with respect to the food source. It is most important role of this parameter in SSA algorithm, since it creates balancing between exploration and exploitation. The sine and cosine functions have been applied to this parameter to enhance the convergence rate and balance between exploration and exploitation. The position equation of SSA algorithm has been modified as follows:

$$x_j^1 = \begin{cases} F_j + \sin(r_2) \times \left| c_1 \times F_j - S_j \right| & c_3 < 0.5 \\ F_j + \cos(r_2) \times \left| c_1 \times F_j - S_j \right| & c_3 \geq 0.5 \end{cases} \quad (9)$$

where x_j^1 is the position of the best solution at the j^{th} dimension, F_j is the food position at the j^{th} dimension, S_j is the salp position at the j^{th} dimension and random number is $r_2 = (2\times\pi)\times rand(\ )$. The parameter c_3 is random numbers uniformly generated in the interval of [0,1]. In fact, they dictate if the next position in j^{th} dimension should be towards positive infinity or negative infinity as well as the step size.

Step 4: Followers locations/or positions updating

The position update in equation (4) of the followers has been modified as equation (10). This methodology helps the best position value of the salp (S_j) in the entire swarm during the searching of the best goal for a given function have been directly providing best position scores in that equation for the enhancing the convergence rate and defeating the premature convergence. The update the position of the followers by following equations:

$$x_j^i = \frac{1}{2}\left(x_j^i + x_j^{i-1}\right) + S_j \tag{10}$$

where x_j^i is the position of the i^{th} follower at the j^{th} dimension, S_j is the salp position at the j^{th} dimension.

Step 5: Stopping condition

Finally, the stopping criteria have been applied for calculating the final optimal solution of the given functions. By the procedure of evaluating each agent/or salp assessment process and updating the best agent's place, it will be repeated again and again until it satisfies the criteria of prevention. i.e. it reaches to the highest number of generations/or iterations or the global optimal result/or goal is earliest found.

The first optimization process is to search optimal results using salp Swarm. Then, the position update equation of Sine and Cosine is used to refine the position of leader and followers during the search process. The rest of operations are the same as Salp Swarm. Algorithm **1** shows the HSSASCA flow.

Algorithm 1. The hybrid HSSASCA

```
Initialization the population X
Repeat
Compute the objective function for each solution x_i
Evaluate each salp in the population (best salp ~ F)
Determine the fitness (value) of each salp
    Update the repository optimal solutions considering the fitness (values) of best agents
Modify the constant c_1 value by equation (2)
    For (all salp (x_i)) do
        If x_i ~ leader Update the position of the leader of the group by applying the mathematical equation (9)
        Else
Update the position of each follower's by applying the mathematical equation (10)
While (t < max_iters)
Return F
```

The procedure of the proposed method is shown in algorithm 1. For the basic computational complexity of the Salp Swarm Algorithm (SSA) is of $O\left(t\left(d*n+\mathrm{Cof}*n\right)\right)$, where d is the number of variables (dimension), t shows the number of iterations, Cof presents the cost of objective function and n is the number of solutions. The time complexity of performing sine and cosine operations is of

$O(\max\_iters * n * d)$, where $\max\_iters$ is the maximum number of iterations or generations. Hence, the time complexity of the proposed algorithm (HSSASCA) is:

$$O(t(d*n+\text{Cof}*n))+O(\max\_iters*n*d).$$

Obviously, the time complexity of the proposed algorithm is higher than that of the standard salp swarm algorithm (SSA) while both of them are in the same order of magnitude.

With the above strategy, the newly hybrid approach hypothetically is competent to determine the global optimum of the optimization function due to the following reasons:

- In HSSASCA, the disparate regions of the search area are explored, when the cosine and sine trigonometry functions return an optimal solution less than negative one (-1) or greater than positive one (+1).
- In HSSASCA, the encouraging regions of the search area is exploited when the trigonometry function gives optimal solution amid negative one (-1) and positive one (+1).
- HSSASCA algorithm produces and refines a set of random optimal goals for the given function. Hence, it intrinsically advantages from the local optima avoidance and high exploration compared to separate based meta-heuristics.
- The HSSASCA approach transits from (exploration to exploitation) the search of solved with the use of optimal range in trigonometry functions.
- The finest estimate of the comprehensive optimum is stored in a variable as the target point and not at all gets mislaid throughout optimization.
- Because the optimal solutions always update their conditions around the best solution they have ever received, there is a trend toward the best areas of search during optimization.
- Because the newly proposed method considers the compatibility problem as a black box, the problems can be easily added in different areas, which are under the solution to the right problem.

5. ANALYSIS

The proposed algorithm has been applied on well-known standard and engineering optimization functions. Here these functions have been chosen for verifying our experimental solution with recent meta-heuristics. All the results are illustrated in Tables 2-10. Further, the experimental results/or solutions of the hybrid method are verified/or compared against the SSA (Salp Swarm Algorithm), PSO (Particle Swarm Optimization), MFO (Moth-Flame Optimization Algorithm), SCA (Sine Cosine Algorithm), DA (Dragonfly Algorithm), MVO (Multi-Verse Optimizer), ALO (Ant Lion Optimizer), MGWO (Mean Grey Wolf Optimizer), DCSGWO (Distributed Compressed Sensing + Grey Wolf Optimizer), FWAGWO (Fireworks Algorithm + Grey Wolf Optimizer) and HAGWO (Hybrid Algorithm of Grey Wolf Optimizer) algorithms.

The HSSASCA (Hybrid Salp Swarm Algorithm + Sine Cosine Algorithm), SSA, PSO, MFO, SCA, DA, MVO, ALO, MGWO, DCSGWO, FWAGWO and HAGWO algorithms are programmed by MATLAB 2015 and implemented on, 15.6” Intel HD Graphics, Pentium-Intel Core (TM), 16.9 HD LCD, 3GB Memory, 320 GB HDD and i5 Processor 430 M.

Massive experiments illustrate that the SCA and SSA techniques can be close to the best condition on these problems/functions when the number of generation/iteration and population size are set to 300 and 30, respectively. For a fair comparison, the swarm size and the number of generations must be the same for all variants used and should also 30 runs each algorithm for check the quality.

Hence, in this work, the same numbers of generation and population size were used for SSA, PSO, MFO, SCA, DA, MVO, ALO, MGWO, DCSGWO, FWAGWO, HAGWO and HSSASCA. Parameter settings for these techniques are listed in Table 1.

Table1. Parameter settings

Algorithm	Parameters
SSA	Set as in [28]
PSO	$s = 30, \max\_iter = 300,$ $v_{\max} = 6, c_1 = c_2 = 2, w_{\max} = 0.9$ and $w_{\min} = 0.2$
MFO	$s = 30, \max\_iter = 300,$ remaining by [33]
SCA	$s = 30, \max\_iter = 300,\ a = 2$
DA	$s = 30, \max\_iter = 300,$ remaining by [31]
MVO	$s = 30, \max\_iter = 300,$ remaining by [29]
ALO	$s = 30, \max\_iter = 300,$ remaining by [36]
MGWO	$s = 30, \max\_iter = 300,$ remaining by [53]
DCSGWO	$s = 30, \max\_iter = 300,$ remaining by [25]
FWAGWO	$s = 30, \max\_iter = 300,$ remaining by [46]
HAGWO	$s = 30, \max\_iter = 300,$ remaining by [5]
HSSASCA	$s = 30, \max\_iter = 300$, $c_1, c_2, c_3 \in [0,1]$, $r_1 = 2\pi \times rand(\ )$, $r_2 = 2 \times rand(\ )$, $v_0 = 0$ and $I_{runs} = 30$

Hence, the hybrid HSSASCA algorithm has been investigated on the tested functions. The uni-modal problems/or tasks are well-known to have only one global optimum and thus can be used to assess the exploitation capability of a meta-heuristic. Regularly having more than one local optimum, multi-modal and fixed-dimension multi-modal problems/or tasks are applied/or used to assess the exploration capability of a meta-heuristic. The proposed algorithm was run 20 times on each benchmark problem. The statistical and numerical solutions have been performed to illustrate that. By best parameter settings, it was found that the best solutions or results of the given functions lie within a reasonable number of generations/ or iterations. The different criteria in this work have been applied to assess the capability of proposed algorithm and others. The statistical values such as average and standard deviation have been used to assess the reliability of the algorithms. Further, the minimum and maximum value of the objective function represent the best possible cost of the given problem in the number of iterations. The average number of function evaluation of the successful runs and average computational time of the successful runs has been utilized to found the best cost of the problems.

For seven uni-modal problem, the quality of solution of the given functions obtained have been illustrated by the best score, max or min objective function value, average, standard deviation, self and total cpu time respectively. These obtained results are reported in table 2 and the convergence performance of the algorithms is shown in figure 2. Further, the accuracy of the algorithms has been verified on six multi-modal and nine fixed dimension functions. The experimental solutions of these functions and convergence performance of the algorithms are presented in table 3-4 and figure 3-4 respectively. At the end, we have solved the optimization engineering problems for verifying the performance of the algorithms and the brief details of these problems and solutions are reported in the section 7.

6. DISCUSSION ON EXPERIMENTAL RESULTS

6.1 Convergence performance of the Algorithms

The convergence performance of the HSSASCA algorithm and the others are presented for tested functions by plotting the standard function values against the-number of generations/or iterations as present in Figure 1. The red line represents the accuracy and performance of the standard SSA algorithm, whereas the black line represents the accuracy and performance of the HSSASCA algorithm. The data in Figure 1 are plotted after d-iterations. The convergence performance of the algorithm proves that the new hybrid method is superior than the standard SSA algorithm and other meta-heuristics. This confirms that the applied partitioning mechanism and the integration between the SSA and SCA algorithm can accelerate the convergence of the proposed algorithm. Here, we concluded that the modification reduces the running time of the algorithms and boosts the accuracy of classification problems.

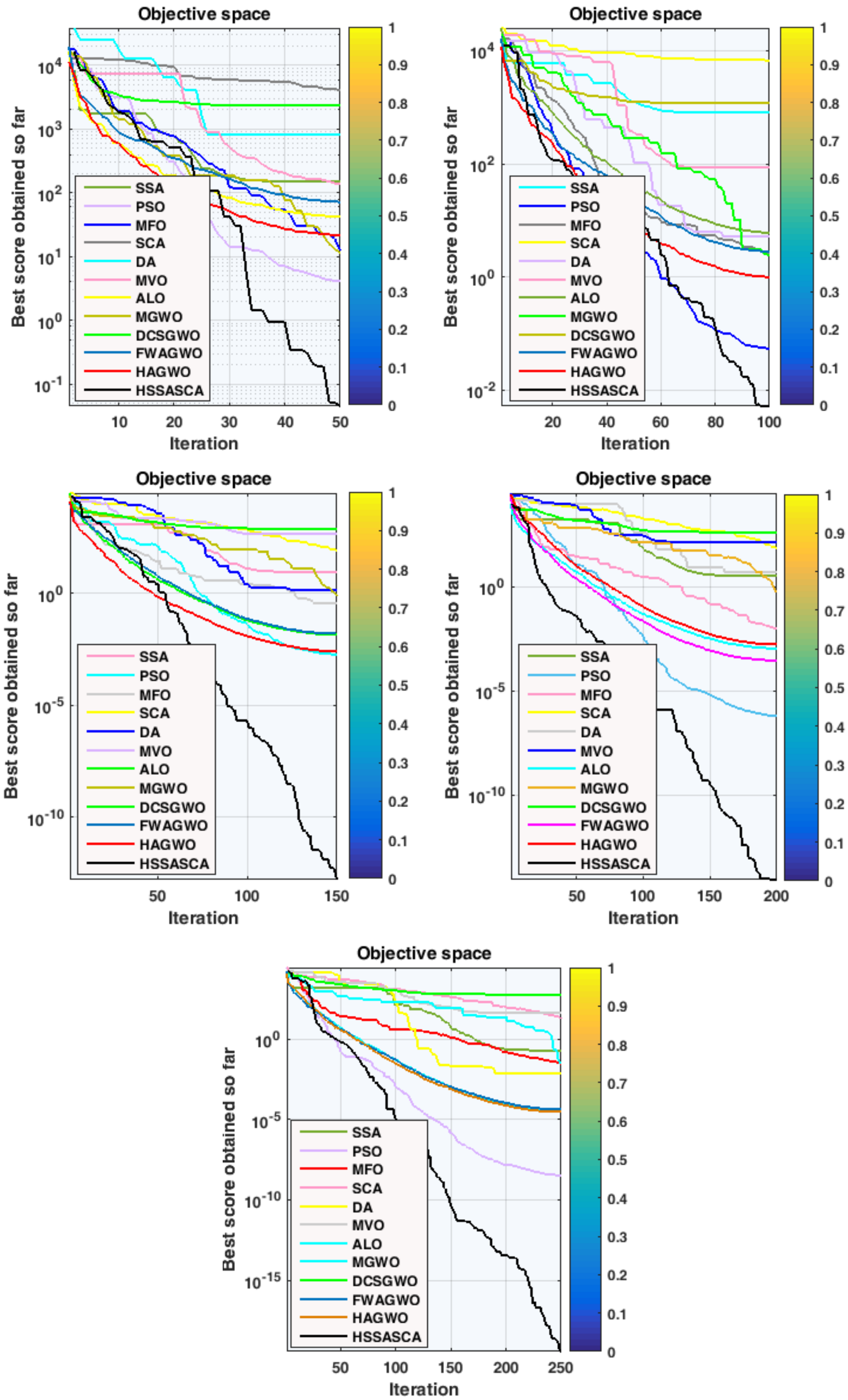


Figure 1. Convergence performance of algorithms on different dimensions

6.2 Standard Functions

The working accuracy of the hybrid HSSASCA algorithm has been confirmed on uni-modal, multi-modal and fixed dimension multi modal tested standard problems/or functions. The tested functions are presented in Appendix (Tables A - C).

6.3 Uni-modal Test Functions

In this subsection, the ability of the proposed method has-been-tested on uni-modal problems/or functions. The obtained solutions of these functions have been discussed in Table 2 and Figure 2. For verifying the performance and ability of the hybrid algorithms, we have used the standard PSO, SSA, MFO, SCA, DA MVO, ALO, MGWO, DCSGWO, FWAGWO and HAGWO algorithms. The best optimal solutions are written in bold font. Here, it-can-be easily seen-that-the newly proposed approach provides better or highly effective global optimal results as compared to other recent comparative algorithms. As previously discussed, these functions are more capable for benchmarking exploitation of the meta-heuristics. Therefore, it is evidence that the proposal achieves high rate of exploitation capability.

Table 2. Results of SSA, PSO, MFO, SCA, DA, MVO, ALO, MGWO, DCSGWO, FWAGWO, HAGWO and HSSASCA algorithms on seven Uni-modal functions at different iterations

Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
1.	SSA	0.2759	0	2.0834e+04	3.9077e+03	4.4991e+03	0.795	0.858
	PSO	0.0598	0.0598	6.7880e+04	1.4489e+03	7.0965e+03	0.950	1.029
	MFO	184.2208	184.2208	7.6447e+04	1.3452e+04	1.8670e+04	0.935	1.108
	SCA	130.2639	0	6.8458e+04	2.2581e+04	2.9145e+04	0.872	0.952
	DA	3.2003e+03	3.2003e+03	6.6178e+04	1.8076e+04	2.3940e+04	15.568	36.892
	MVO	3.9565	3.9565	4.9374e+04	4.3413e+03	8.5599e+03	3.712	5.163
	ALO	6.2254	0	3.9019e+04	2.6497e+03	5.8941e+03	1.254	40.886
	MGWO	2.8399e-04	2.8399e-4	7.6545e+04	1.3452e+04	1.8670e+04	1.669	1.763
	DCSGWO	2.3764e-04	2.3764e-04	6.3262e+04	767.7876	4.4856e+03	1.779	1.888
	FWAGWO	1.2829e-04	1.2829e-04	7.3761e+04	751.0181	5.0120e+03	1.811	1.857
	HAGWO	4.4683e-18	4.4683e-37	7.8074e+04	658.1758	5.5804e+03	3.199	3.276
	HSSASCA	1.2220e-20	0	7.8823e+04	1.273320	9.0073e+03	0.483	0.515
Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
2.	SSA	0.0106	0	17.3558	4.3063	4.7084	0.266	0.358
	PSO	1.2915e-06	1.2915e-06	92.4274	2.2606	5.9981	0.328	0.390
	MFO	1.1725e-05	1.1725e-05	472.5344	11.2343	57.1102	0.234	0.359
	SCA	2.3263e-05	0	20.3387	2.4146	4.6487	0.201	0.281
	DA	0.9367	0.9367	187.2231	11.4567	29.6896	7.884	12.793
	MVO	0.0736	0.0736	171.9180	4.8690	11.3572	0.685	1.014
	ALO	1.6103	0	22.9658	5.6325	6.8939	0.516	8.550
	MGWO	6.2826e-07	6.2826e-07	1.9385e+03	6.8720	111.9145	0.467	0.561
	DCSGWO	1.5601e-06	1.5601e-06	72.9144	0.9138	5.1513	0.390	0.453
	FWAGWO	1.1151e-06	1.1151e-06	33.4773	0.5179	2.7794	0.466	0.546
	HAGWO	6.5042e-06	6.5042e-34	2.2434e+03	7.8006	129.5365	0.858	0.983
	HSSASCA	1.5649e-25	0	522.6365	0.4596	2.2949	0.125	0.187
Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
3.	SSA	0.0069	0	5.3657e+03	443.0421	578.0686	0.517	1.404
	PSO	5.3411e-04	5.3411e-04	1.8194e+04	459.8058	2.2769e+03	0.126	0.904
	MFO	1.0982	1.0982	1.0448e+04	1.7434e+03	2.2465e+03	0.485	1.404
	SCA	0.0035	0	5.3657e+03	443.0421	578.0686	0.343	1.248

Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
	DA	147.3776	147.3776	9.1964e+03	2.3952e+03	3.0610e+03	12.699	23.759
	MVO	0.3424	0.3424	2.2581e+04	806.4928	2.2571e+03	0.984	1.825
	ALO	143.0023	0	7.8399e+03	1.0820e+03	1.7367e+03	0.419	8.487
	MGWO	4.9523e-07	4.9523e-07	2.0697e+04	182.8018	1.2946e+03	0.765	1.622
	DCSGWO	2.3524e-06	2.3524e-06	9.4976e+03	107.9226	683.9416	0.668	1.170
	FWAGWO	3.4906e-07	3.4906e-07	1.4675e+04	101.5267	912.9078	0.673	1.388
	HAGWO	5.8588e-14	5.8588e-14	1.3787e+04	271.7098	1.3267e+03	1.638	2.216
	HSSASCA	8.1256e-20	0	9.7932e+04	1.0234e+3	951.0627	0.299	0.717

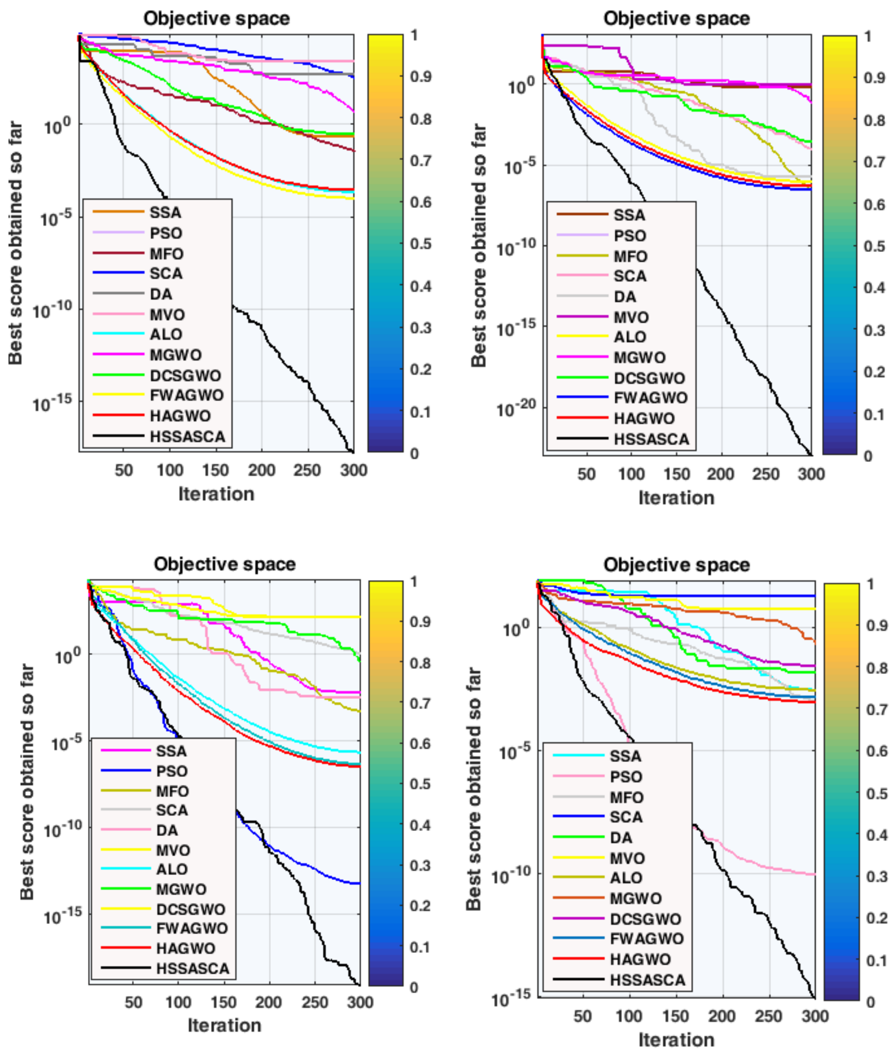
Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
4.	SSA	2.3394e-04	0	38.6947	8.3834	9.4275	0.562	0.780
	PSO	0.0018	0.0018	73.6156	2.9436	10.2213	0.781	0.874
	MFO	1.7649	1.7649	59.2316	14.1327	12.0027	0.455	0.577
	SCA	0.0144	0	68.9919	20.7699	29.0354	0.280	0.468
	DA	1.8577	1.8577	56.9883	14.7945	18.7588	11.600	21.060
	MVO	0.1512	0.1512	57.9955	6.5487	8.0449	1.531	2.106
	ALO	1.2354	0	42.7337	7.5733	9.1118	0.848	10.296
	MGWO	0.0012	0.0012	55.9351	1.3939	5.3739	0.778	1.030
	DCSGWO	0.0019	0.0019	69.8150	1.6043	6.1068	0.704	0.858
	FWAGWO	7.7089e-04	7.7089e-04	69.6581	0.9924	5.0094	0.715	0.920
	HAGWO	7.6393e-08	7.6393e-12	62.8402	0.9806	5.2959	1.343	1.451
	HSSASCA	2.5975e-17	0	76.4551	0.5363	6.3498	0.185	0.296

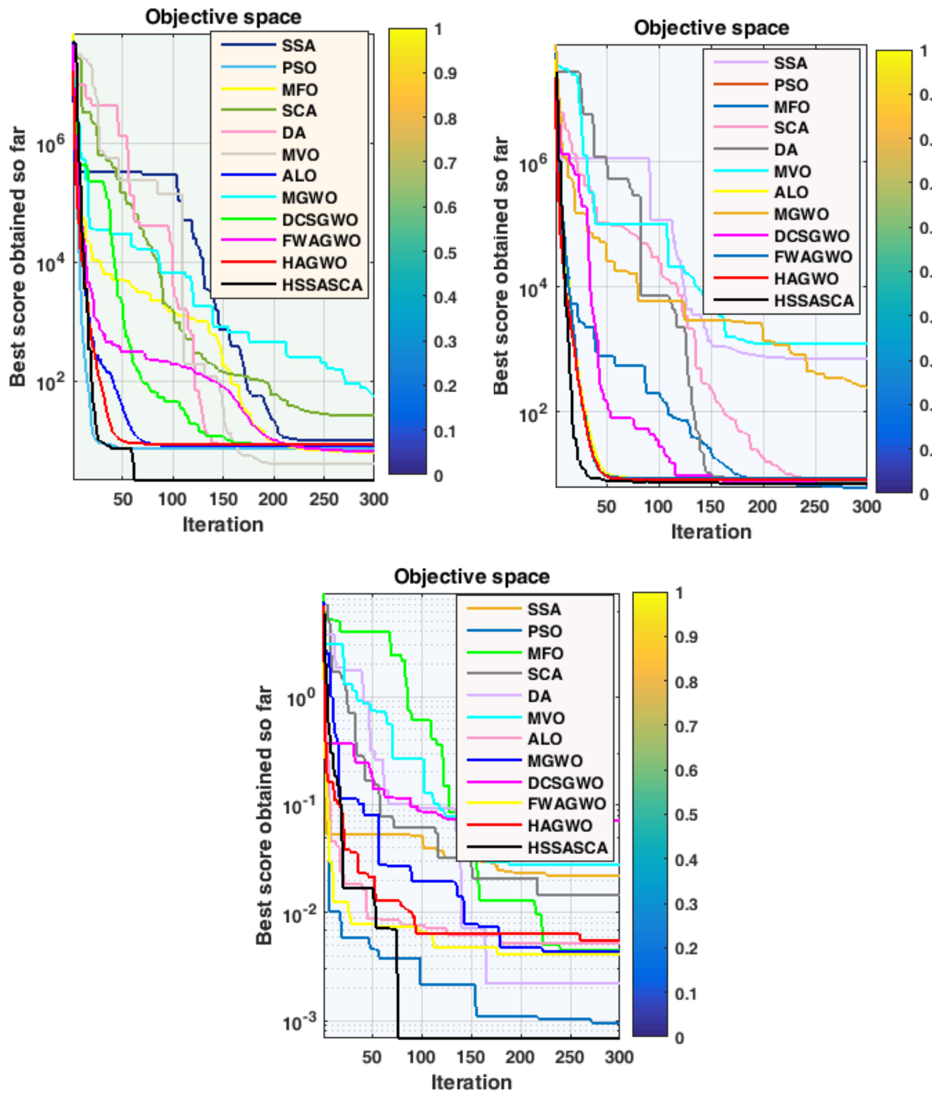
Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
5.	SSA	8.3157	0	1.3494e+06	1.9852e+05	2.8568e+05	0.655	0.889
	PSO	5.6829	5.6829	3.5715e+07	1.9152e+05	2.2477e+06	0.545	0.952
	MFO	21.2357	21.2357	1.7631e+07	8.2866e+05	3.0527e+06	0.639	0.795
	SCA	7.3599	0	1.3494e+06	1.9852e+05	2.8568e+05	0.531	0.686
	DA	141.9876	141.9876	5.2798e+07	3.0983e+06	8.4395e+06	13.117	23.648
	MVO	2.1040e+03	2.1040e+03	1.3201e+07	2.2837e+05	1.3008e+06	1.484	2.106
	ALO	8.7070	2.1040e+03	1.3201e+07	2.2837e+05	1.3008e+06	1.140	15.614
	MGWO	8.9504	8.9504	2.5173e+07	9.6965e+04	1.4602e+06	0.842	0.967
	DCSGWO	8.9602	8.9602	1.7942e+07	7.5526e+04	1.0570e+06	0.953	1.123
	FWAGWO	9.6635	9.6635	2.8436e+07	1.1315e+05	1.6614e+06	0.845	1.014
	HAGWO	7.2040	7.2040	3.6735e+07	1.8210e+05	2.2771e+06	1.452	1.669
	HSSASCA	6.8053	0	5.4183e+07	1.3471e+04	5.1118e+06	0.237	0.390

Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
6.	SSA	7.9154e-10	0	4.4506e+03	746.7452	975.7130	0.608	0.671
	PSO	4.5756e-15	4.5756e-15	1.7447e+04	335.3805	1.9012e+03	0.672	0.733
	MFO	2.0967e-06	2.0967e-06	2.0146e+04	1.1386e+03	3.1252e+03	0.563	0.671
	SCA	0.1799	0	1.9716e+04	994.3458	2.0959e+03	0.499	0.577
	DA	32.3095	32.3095	5.5216e+03	988.6377	1.6564e+03	14.385	24.442
	MVO	0.0417	0.0417	1.4102e+04	391.3245	1.3024e+03	1.451	2.012
	ALO	3.8248e-07	0	4.93083+03	201.1074	682.6841	0.670	0.811
	MGWO	1.0053	1.0053	1.1956e+04	74.7038	717.8152	1.173	14.849
	DCSGWO	2.0017	2.0017	1.6848e+04	106.0464	1.0448e+03	0.780	0.858
	FWAGWO	0.2533	0.2533	1.5178e+04	96.9959	950.+6137	0.859	0.889
	HAGWO	3.4225e-05	3.4225e-05	1.9603e+04	141.1776	1.3994e+03	1.423	1.544
	HSSASCA	1.7573e-15	0	5.6263e+04	112.1714	1.3203e+03	0.141	0.281

Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
7.	SSA	0.0219	0	0.8737	0.0394	0.0559	0.562	0.780
	PSO	0.0045	0.0045	9.0468	1.1778	1.8024	0.669	0.842

	MFO	0.0146	0.0146	7.0675	0.3132	1.0587	0.485	0.609
	SCA	0.0023	0	3.7177	0.3634	0.8358	0.470	0.624
	DA	0.0280	0.0280	3.0533	0.4134	0.7881	12.300	22.905
	MVO	0.0043	0.0043	7.6655	0.1185	0.5758	1.454	1.966
	ALO	0.0719	0	0.3726	0.1204	0.0929	0.798	15.118
	MGWO	0.0041	0.0041	6.6000	0.0323	0.3832	0.750	0.858
	DCSGWO	0.0053	0.0053	7.4071	0.0440	.4361	0.731	0.936
	FWAGWO	0.0055	0.0055	7.0363	0.0454	0.4123	0.841	0.936
	HAGWO	9.5054e-04	9.5054e-04	3.3142	0.0244	0.2209	1.500	1.654
	HSSASCA	6.9032e-04	0	9.7721	0.0197	0.3771	0.218	0.327





Figures 2. Convergence graphs of algorithms on Uni-modal functions

6.4 Multi-Modal Test Functions

The accuracy/or performance of new hybrid method has-been discussed on the multi-modal functions in this subsection and also verifying the ability of the algorithm with others. The experimental results of these functions have been prescribed in Table 3 and Figure 3. The superiority and ability of the proposed variant has been verified in the terms of best scores, average, minimum-and-maximum-objective-function-value, standard-deviation, self and total time on different dimensions. Here we see that, the proposed approach achieves superior quality of numerical solutions on these functions outperforms than others. Moreover, the testing results reveal that the high exploration of the new hybrid method is competent to explore the search space extensively and provide potential areas of the search field.

Table 3. Results of SSA, PSO, MFO, SCA, DA, MVO, ALO, MGWO, DCSGWO, FWAGWO, HAGWO and HSSASCA algorithms on six Multi-modal functions at different iterations

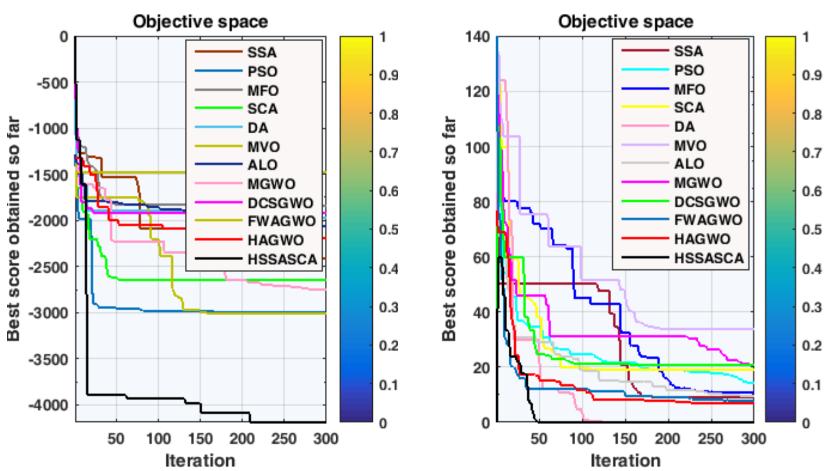
Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
1.	SSA	-2.4116e+03	-2.4116e03	0	-2.0829e+03	425.4179	0.670	0.889
	PSO	-1.8404e+03	-1.8404e+03	-997.8602	-1.7778e+03	167.7536	0.732	0.811
	MFO	-2.6410e+03	-2.6410e+03	-897.3669	-2.5613e+03	243.4334	0.609	0.733
	SCA	-1.9765e+03	-1.9765e+03	0	-1.8812e+03	133.6997	0.515	0.639
	DA	-3.0145e+03	-3.145e+03	-1.3349e+03	-2.5636e+03	566.1670	10.870	21.586
	MVO	-2.7450e+03	-2.7450e+03	-1.4355e+03	-2.3753e+03	361.4196	1.327	1701
	ALO	-1.9258e+03	-1.9258e+03	0	-1.9002e+03	153.8352	0.858	15.238
	MGWO	-1.4753e+03	-1.4753e+03	-1.4753e+03	-1.4753e+03	2.2775e-13	0.875	0.952
	DCSGWO	-2.0578e+03	-2.0578e+03	-1.2804e+03	-1.9186e+03	137.3616	0.810	0.951
	FWAGWO	-2.1944e+03	-2.1944e+03	-1.3215e+03	-2.0473e+03	216.2622	0.841	0.952
	HAGWO	-3.0038e+03	-3.0038e+03	-1.3125e+03	-2.9084e+03	275.5093	1.513	1.622
	HSSASCA	-4.1896e+03	-4.1896e+03	0	-3.9115e+03	606.3818	0.267	0.328

Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
2.	SSA	8.9546	0	81.7858	28.5519	19.9196	0.654	0.889
	PSO	10.9448	10.9448	127.4636	38.4682	26.3051	0.593	0.749
	MFO	18.9042	18.9042	122.8827	27.4775	20.6718	0.574	0.718
	SCA	8.7333e-08	0	124.1208	11.2527	26.3977	0.532	0.702
	DA	33.8302	33.8302	134.4354	52.2949	22.9888	11.188	20.123
	MVO	20.9123	20.9123	127.2166	34.9222	14.8151	1.436	1.935
	ALO	20.8941	0	107.3150	26.6767	14.8754	1.036	14.976
	MGWO	7.8697	7.8697	140.1540	12.8307	12.1917	0.922	1.077
	DCSGWO	9.6395	9.6395	118.3669	19.9249	13.5677	0.872	0.998
	FWAGWO	6.7601	6.7601	76.7524	13.4773	13.6135	0.967	1.061
	HAGWO	14.2713	14.2713	91.5892	25.4487	12.6885	1.544	1.716
	HSSASCA	0	0	159.8672	3.8438	11.3658	0.187	0.344

Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
3.	SSA	1.1551	0	16.8131	6.1739	5.2526	0.592	0.951
	PSO	5.6659e-07	5.6659e-07	19.8968	2.0283	3.0227	0.811	1.014
	MFO	1.6335e-04	1.6335e-04	20.1039	3.1792	5.3948	0.671	0.921
	SCA	7.2574e-05	0	18.3448	4.5073	6.9009	0.470	0.827
	DA	2.8165	2.8165	20.2293	7.6156	6.1782	13.477	24.025
	MVO	0.1112	0.1112	20.2943	5.5607	3.3157	1.569	2.309
	ALO	4.7989e-04	0	17.1075	2.5250	4.0269	1.051	15.304
	MGWO	3.4251	3.4251	19.8991	3.9145	1.7838	0.890	1.123
	DCSGWO	3.4093	3.4093	19.6077	3.9133	1.8934	0.856	1.061
	FWAGWO	3.5745	3.5745	20.6039	3.9387	1.6163	0.809	1.123
	HAGWO	7.9936e-15	7.9936e-15	16.7955	0.3859	2.0401	1.437	1.685
	HSSASCA	4.4409e-15	0	20.8976	0.3421	1.6766	0.234	0.453

Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
4.	SSA	0.2140	0	44.8338	8.6201	10.8585	0.656	0.827
	PSO	0.0910	0.0910	118.7388	14.6086	30.9800	0.595	0.780
	MFO	0.5658	0.5658	159.3326	12.6717	31.8605	0.577	0.811
	SCA	0.3985	0	127.9587	6.6873	21.4123	0.546	0.702
	DA	0.5188	0.5188	172.6721	18.2751	45.0597	12.817	22.678
	MVO	0.4759	0.4759	99.8207	4.4026	11.9767	1.280	1.810
	ALO	0.3320	0	45.6347	3.4729	9.0076	0.738	13.913

	MGWO	0.3183	0.3183	95.7587	5.6985	1.1209	0.717	0.967
	DCSGWO	0.0199	0.0199	137.9988	8.1391	0.8978	0.792	1.014
	FWAGWO	0.2699	0.2699	142.6935	1.2146	8.5175	0.801	1.046
	HAGWO	0.0760	0.0760	137.0655	1.1758	9.3190	1.525	1.684
	HSSASCA	0	0	199.5720	0.9100	7.2836	0.186	0.359
Fun. No.	**Algorithm**	**Best Score**	**Min Value**	**Max Value**	**Mean**	**S.D.**	**Self Time (s)**	**Total Time (s)**
5.	SSA	3.7703	0	1.3683e+06	4.5741e+03	7.8999e+04	0.483	1.123
	PSO	5.6729e-16	5.6729e-16	3.3044e+07	1.6631e+05	2.0397e+06	0.764	1.342
	MFO	0.3110	0.3110	5.3700e+07	1.4316e+06	7.3381e+06	0.465	1.264
	SCA	0.1381	0	7.1577e+07	6.7998e+05	2.0068e+07	0.527	1.170
	DA	0.7992	0.7992	6.8721e+07	8.2586e+06	2.0728e+07	14.620	25.620
	MVO	0.4393	0.4393	8.5204e+07	8.0930e+05	6.2949e+06	1.713	2.527
	ALO	0.3864	0	8.0049e+05	1.5012e+04	4.2542e+04	1.203	16.023
	MGWO	0.9801	0.9801	8.2126e+07	2.7614e+05	4.7416e+06	0.734	1.419
	DCSGWO	1.2909	1.2909	5.7562e+06	2.1552e+04	3.3471e+05	0.858	1.467
	FWAGWO	2.5874	2.5874	4.5782e+07	1.5274e+05	2.6432e+06	0.878	1.467
	HAGWO	0.0396	0.0396	4.4526e+07	3.4605e+05	3.6925e+06	1.264	1.747
	HSSASCA	0.0294	0	8.8939e+07	1.0155e+05	4.3034e+06	0.251	0.702
Fun. No.	**Algorithm**	**Best Score**	**Min Value**	**Max Value**	**Mean**	**S.D.**	**Self Time (s)**	**Total Time (s)**
6.	SSA	0.0110	0	2.2956e+07	9.3937e+04	1.3308e+06	0.687	1.263
	PSO	5.4204e-14	5.4204e-14	1.1091e+08	7.9905e+05	8.1415e+06	0.608	1.248
	MFO	1.0427e-05	1.0427e-05	2.2890e+08	4.2415e+06	2.3843e+07	0.407	0.827
	SCA	0.3529	0	8.5715e+07	1.0342e+07	2.2772e+07	0.407	0.827
	DA	0.2926	0.2926	1.6085e+08	1.4400e+07	4.1299e+07	13.120	23.872
	MVO	0.0050	0.0050	1.3696e+08	1.1160e+06	1.0416e+07	1.343	2.231
	ALO	1.3600e-06	0	3.1454e+07	8.4060e+05	3.3873e+06	1.308	14.947
	MGWO	0.2892	0.2892	1.2595e+08	4.3471e+05	7.2744e+06	0.858	1.420
	DCSGWO	1.0270	1.0270	7.0615e+07	2.5140e+05	4.0809e+06	0.779	1.342
	FWAGWO	1.2499	1.2499	2.5434e+08	1.0215e+06	1.4858e+07	0.856	1.404
	HAGWO	0.5034	0.5034	1.1278e+08	4.7706e+05	6.6348e+06	1.343	2.231
	HSSASCA	1.0215e-07	0	9.4968e+08	1.0799e+05	6.3980e+06	0.248	0.608



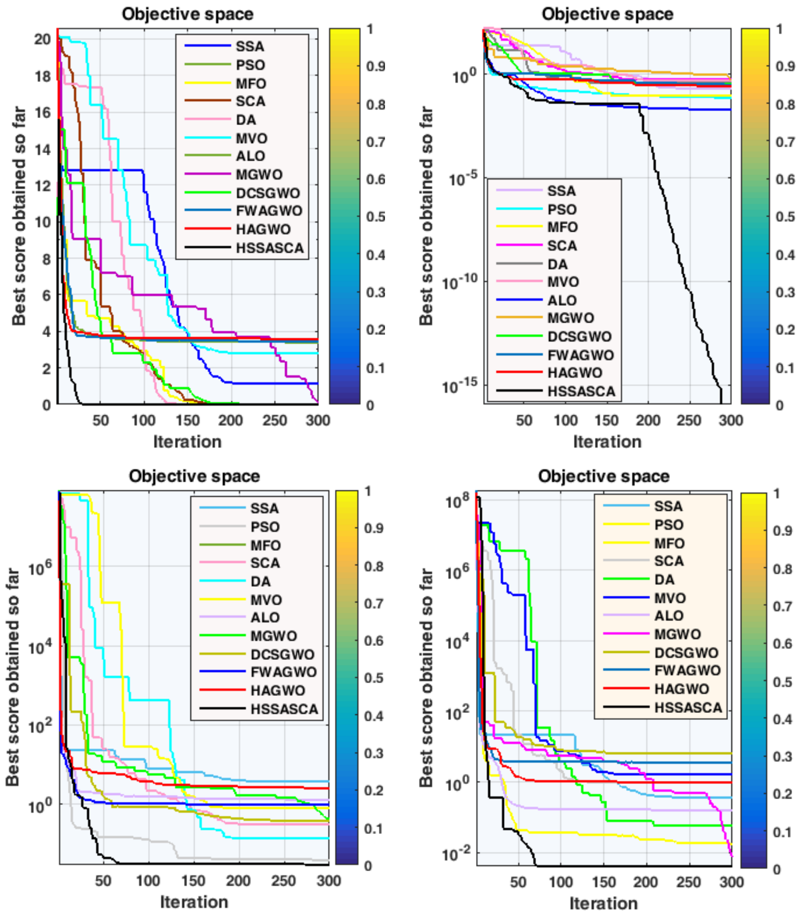


Figure 3. Convergence graphs of algorithms on Multi-modal functions

6.5 Fixed-Dimension-Multi-Modal-Test-Functions

In this subsection, we discuss solution of the fixed dimension multimodal functions. All the solutions of meta-heuristics have been illustrated in Table 9 and Figure 4. Table 7 reveals that the proposed variant is more competent and reliable to search the best and superior quality of the optimal results in the search area/or space of the functions. These solutions depict that in the modified method has been better characteristics in superior quality of the optimal results and also robustness of the optimal solutions.

6.6 Exploitation Tendency

As per experimental solutions of Table 7, Hybrid method is able to found best, possible and very completive solutions of the uni-modal tested problems. This approach outperforms than others in all problems/or functions. It could be noted that the uni-modal problems are more appropriate for-benchmarking exploitation. Hence, these optimal solutions show the better performance of newly method in-terms of exploiting the optimum. This is owing to the planned exploitation operators discussed earlier.

6.7 Exploration Tendency

In distinction to the multi-modal and uni-modal tested problems, there are several local optima, whose number is increasing with the dimension. This makes them suitable for testing functions the exploration capability of an meta-heuristic. As per solutions of the Table 3 and 4, newly proposed algorithm is competent to find the best, possible and highly competitive solutions, on these test functions/or problems as well. The HSSASCA outperforms SSA, PSO, MFO, SCA, DA, MVO, ALO, MGWO, DCSGWO, FWAGWO and HAGWO on the majority of the tested problems. All statistical solutions reveal that the newly hybrid approach has highly merit in the terms of exploration.

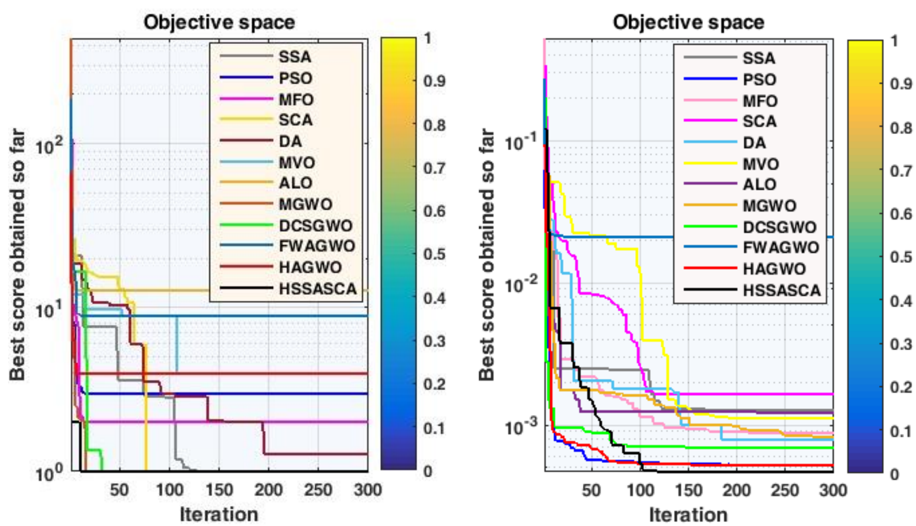
Table 4. Results of SSA, PSO, MFO, SCA, DA, MVO, ALO, MGWO, DCSGWO, FWAGWO, HAGWO and HSSASCA algorithms on nine fixed dimension multi-modal functions at different iterations

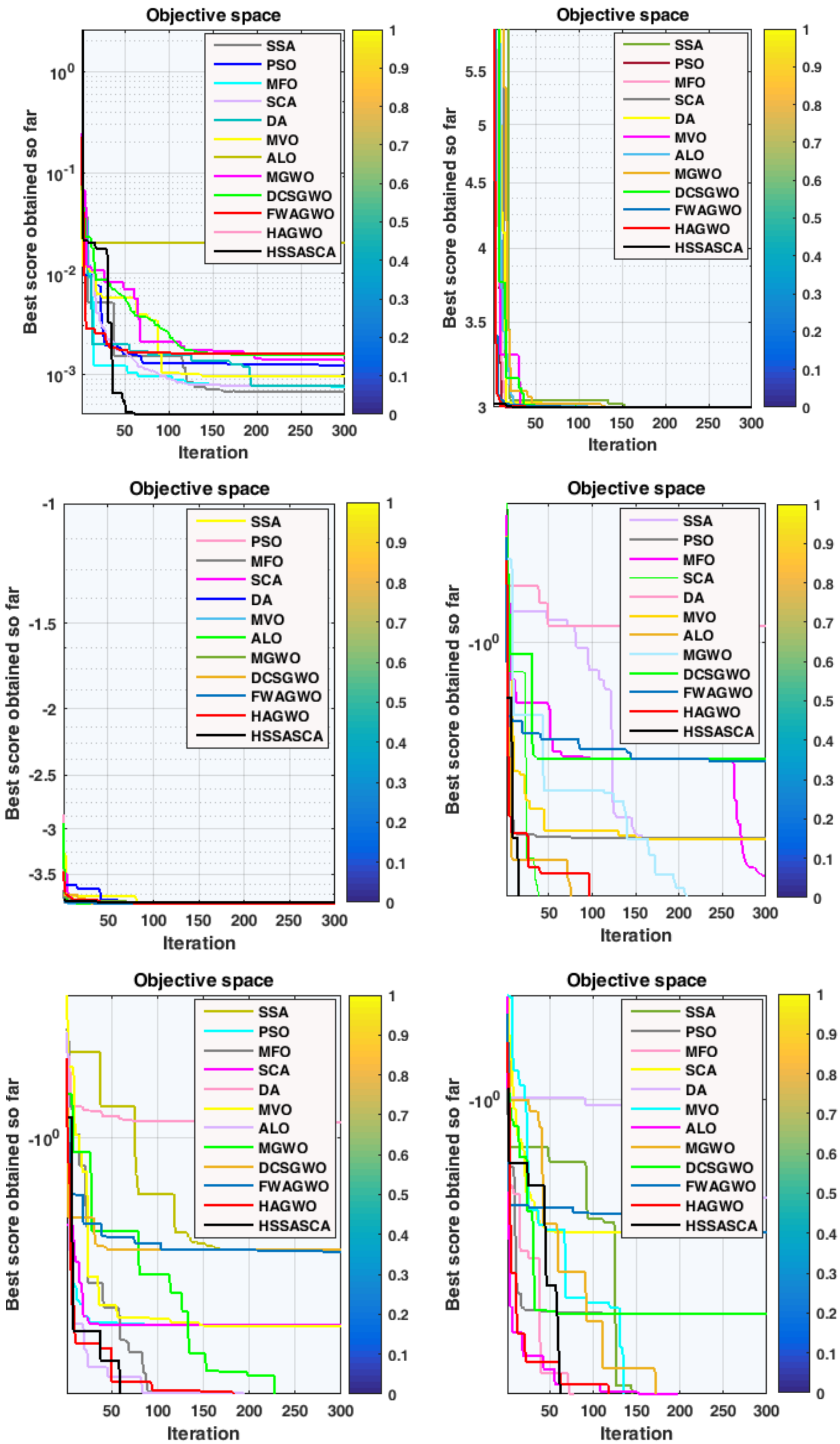
Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
1.	SSA	0.9980	0	20.5960	2.9227	4.0735	0.581	2.980
	PSO	1.9920	1.9920	167.1864	2.9917	10.7484	0.640	3.168
	MFO	0.9980	0.9980	26.6434	4.4308	6.4538	0.363	3.137
	SCA	1.2776	0	21.0748	4.2044	4.4692	0.327	3.090
	DA	3.9683	3.9683	12.0260	5.9982	2.7965	10.021	17.210
	MVO	0.9980	0.9980	440.7209	2.6465	25.4380	0.468	3.043
	ALO	0.9980	0	16.4741	1.7857	3.3767	0.651	6.303
	MGWO	8.8408	8.8408	188.1171	9.5783	10.3986	0.329	2.871
	DCSGWO	12.6705	12.6705	172.1800	13.2671	9.26705	0.442	3.027
	FWAGWO	3.9683	3.9683	66.2492	4.2082	3.6046	0.157	2.825
	HAGWO	2.9821	2.9821	22.2761	3.1396	1.3548	0.716	3.527
	HSSASCA	0.9980	0	557.9979	1.0247	0.1809	0.249	2.309
Fun. No.	**Algorithm**	**Best Score**	**Min Value**	**Max Value**	**Mean**	**S.D.**	**Self Time (s)**	**Total Time (s)**
2.	SSA	0.0032	0	0.3935	0.0160	0.0340	0.420	0.608
	PSO	8.7454e-04	8.7454e-04	0.0869	0.0043	0.0098	0.420	0.640
	MFO	0.0015	0.0015	0.1614	0.0040	0.0168	0.406	0.546
	SCA	0.0016	0	0.1533	0.0050	0.0162	0.635	1.083
	DA	0.0017	0.0017	0.0563	0.0032	0.0074	10.233	16.353
	MVO	7.8148e-04	7.8148e-04	0.1850	0.0031	0.0170	0.637	1.092
	ALO	0.0012	0	0.0258	0.0016	0.0020	0.406	5.383
	MGWO	6.2455e-04	3.2455e-04	0.0273	6.3083e-04	0.0016	0.311	0.468
	DCSGWO	0.0204	0.0204	0.4823	0.0223	0.0268	0.484	0.546
	FWAGWO	0.0072	0.0072	0.0960	0.0061	0.0079	0.328	0.468
	HAGWO	6.0925e-04	6.0925e-04	0.01289	0.0013	0.0075	0.810	0.936
	HSSASCA	5.0974e-04	0	0.7326	0.0004	0.0091	0.077	0.218
Fun. No.	**Algorithm**	**Best Score**	**Min Value**	**Max Value**	**Mean**	**S.D.**	**Self Time (s)**	**Total Time (s)**
3.	SSA	-1.0316	-1.0316	0	-1.0114	0.0625	0.451	0.483
	PSO	-1.0316	-1.0316	0.5105	-1.0144	0.0958	0.375	0.421
	MFO	-1.0316	-1.0316	-0.5932	-1.0261	0.0460	0.311	0.343
	SCA	-1.0316	-1.0316	0	-1.0219	0.0625	0.265	0.265
	DA	-1.0316	-1.0316	-0.1815	-1.0206	0.0667	9.776	14.988
	MVO	-1.0316	-1.0316	-0.8003	-1.0131	0.0507	0.483	0.796
	ALO	-1.0316	-1.0316	0	-1.0088	0.0846	0.513	3.696
	MGWO	-1.0316	-1.0316	13.2415	-0.9795	0.8272	0.250	0.265
	DCSGWO	-1.0316	-1.0316	-0.9555	-1.0310	0.0067	0.313	0.374
	FWAGWO	-1.0316	-1.0316	1.7977	-1.0215	0.1637	0.265	0.296
	HAGWO	-1.0316	-1.0316	8.0805	-0.9837	0.5404	0.623	0.686
	HSSASCA	-1.0316	-1.0316	0	-1.0364	0.9637	0.093	0.125
Fun. No.	**Algorithm**	**Best Score**	**Min Value**	**Max Value**	**Mean**	**S.D.**	**Self Time (s)**	**Total Time (s)**
4.	SSA	3.0000	0	7.9274	3.1482	0.7718	0.405	0.452
	PSO	3.0000	3.0000	19.5874	3.3578	1.6602	0.359	0.390
	MFO	3.0000	3.0000	124.9686	5.7929	15.0834	0.266	0.390

	SCA	3.0002	0	35.2907	3.8395	3.9662	0.280	0.328
	DA	3.0000	3.0000	180.3669	3.8784	10.5057	10.157	14.996
	MVO	3.0000	3.0000	13.5599	3.3639	1.8021	0.438	0.718
	ALO	3.0000	0	220.9587	4.1085	12.8038	0.425	3.246
	MGWO	3.0004	3.0006	40.2218	3.2219	2.6586	0.326	0.374
	DCSGWO	3.0007	3.0009	42.3056	3.2389	2.6805	0.313	0.328
	FWAGWO	3.0003	3.0000	59.3899	3.2371	3.3222	0.328	0.359
	HAGWO	3.0000	3.0006	80.3246	3.2815	4.4711	0.624	0.686
	HSSASCA	3.0000	0	199.1416	3.0233	6.4278	0.048	0.125
Fun. No.	**Algorithm**	**Best Score**	**Min Value**	**Max Value**	**Mean**	**S.D.**	**Self Time (s)**	**Total Time (s)**
5.	SSA	-3.8628	-3.8628	0	-3.8103	0.2302	0.484	0.702
	PSO	-3.8628	-3.8628	-3.6959	-3.8611	0.0102	0.389	0.670
	MFO	-3.8628	-3.8628	-3.4972	-3.8548	0.0478	0.391	0.702
	SCA	-3.8512	-3.8512	0	-3.8103	0.2302	0.372	0.561
	DA	-3.8628	-3.8628	-3.7891	-3.8611	0.0085	9.418	14.601
	MVO	-3.8628	-3.8628	-3.7774	-3.8495	0.0190	0.596	1.061
	ALO	-3.8628	-3.8628	0	-3.8412	0.2243	0.499	4.337
	MGWO	-3.8627	-3.8627	-3.7996	-3.8596	0.0059	0.404	0.671
	DCSGWO	-3.8627	-3.8627	-2.9422	-3.8573	0.0546	0.373	0.639
	FWAGWO	-3.8626	-3.8626	-3.4605	-3.8503	0.0338	0.406	0.655
	HAGWO	-3.8619	-3.8619	-2.8640	-3.8562	0.0583	0.749	1.108
	HSSASCA	-3.8747	-3.8747	0	-3.8990	0.2625	0.078	0.405
Fun. No.	**Algorithm**	**Best Score**	**Min Value**	**Max Value**	**Mean**	**S.D.**	**Self Time (s)**	**Total Time (s)**
6.	SSA	-3.1891	-3.1891	0	-3.0012	0.3369	0.498	0.796
	PSO	-3.3220	-3.3220	-1.9358	-3.2112	0.1981	0.529	0.812
	MFO	-3.3220	-3.3220	-1.0571	-3.2213	0.3406	0.296	0.624
	SCA	-2.9947	-2.9947	0	-2.7919	0.2863	0.278	0.577
	DA	-3.1834	-3.1834	-0.8540	-2.8268	0.7628	10.001	16.927
	MVO	-3.3220	-3.3320	-1.1084	-3.1077	0.4177	0.780	1.295
	ALO	-3.3220	-3.3220	0	-3.2221	0.3272	0.392	7.566
	MGWO	-3.2003	-3.2003	-0.8011	-3.1328	0.1728	0.498	0.733
	DCSGWO	-3.220	-3.3220	-2.0937	-3.3008	0.0829	0.467	0.764
	FWAGWO	-3.2005	-3.2005	-1.1022	-3.1428	0.1385	0.514	0.718
	HAGWO	-3.3220	-3.3220	-1.2463	-3.2341	0.1915	0.875	1.217
	HSSASCA	-3.3334	-3.3334	0	-3.3035	0.2012	0.188	0.375
Fun. No.	**Algorithm**	**Best Score**	**Min Value**	**Max Value**	**Mean**	**S.D.**	**Self Time (s)**	**Total Time (s)**
7.	SSA	-2.6305	-2.6305	0	-1.9516	0.0865	0.406	1.061
	PSO	-10.1532	-10.1532	-0.3889	-8.2818	2.9933	0.484	0.998
	MFO	-5.0552	-5.0552	-2.1183	-4.9041	0.5773	0.295	0.749
	SCA	-0.8788	-0.8788	0	-0.8527	0.0697	0.392	0.874
	DA	-5.1008	-5.1008	-0.2933	-4.5620	1.1286	10.366	16.692
	MVO	-10.1517	-10.1517	-0.6883	-5.8217	3.1665	0.608	1.217
	ALO	-2.6305	-2.6305	0	-2.5495	0.2579	0.502	5.132
	MGWO	-2.6828	-2.6828	-1.0478	-2.4915	0.3078	0.265	0.936
	DCSGWO	-10.1529	-10.1529	-0.4052	-8.7260	1.5970	0.265	0.936
	FWAGWO	-10.1523	-10.1523	-0.5025	-8.5997	1.6691	0.594	0.936
	HAGWO	-5.0551	-5.0551	-0.4099	-4.9000	0.5906	0.797	1.186
	HSSASCA	-10.7670	-10.7670	0	-8.9083	1.9256	0.109	0.515

Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
8.	SSA	-10.4029	-10.4029	0	-6.6534	4.1686	0.502	1.155
	PSO	-10.4029	-10.4029	-0.5261	-9.0420	2.5434	0.577	1.139
	MFO	-2.7519	-2.7519	-0.4646	-2.6001	0.4474	0.425	1.061
	SCA	-2.1206	-10.4029	0	-6.6534	4.1686	0.281	0.874
	DA	-10.4029	-10.4029	-0.4527	-7.2501	3.6717	10.286	16.376
	MVO	-10.4016	-10.4016	-0.7142	-6.7928	3.2783	0.590	1.419
	ALO	-5.1288	-5.1288	0	-4.7426	1.1003	0.557	6.316
	MGWO	-2.7518	-2.7518	-0.5219	-2.4939	0.2473	0.406	1.077
	DCSGWO	-10.4007	-10.4007	-0.4564	-9.0084	1.4387	0.453	1.061
	FWAGWO	-10.4018	-10.4018	-0.6443	-9.1005	1.7360	0.436	0.998
	HAGWO	-10.4029	-5.0876	-0.9081	-4.9373	0.6288	0.639	1.358
	HSSASCA	-10.4029	-10.4029	0	-9.8214	3.8687	0.124	0.640

Fun. No.	Algorithm	Best Score	Min Value	Max Value	Mean	S.D.	Self Time (s)	Total Time (s)
9.	SSA	-2.8711	-2.8711	0	-2.4130	0.7001	0.410	1.357
	PSO	-10.5364	-10.5364	-0.6748	-5.7052	3.2818	0.465	1.389
	MFO	-5.1756	-10.5352	-0.6100	-9.0959	1.8907	0.358	1.279
	SCA	-4.5889	-4.5889	0	-3.4313	1.5506	0.298	1.123
	DA	-10.5364	-10.5364	-0.5823	-9.0404	2.3700	10.318	17.037
	MVO	-10.5350	-10.5350	-0.6869	-6.1065	3.0318	0.388	2.558
	ALO	-2.8711	-2.8711	0	-2.7203	0.4816	0.646	6.786
	MGWO	-10.5352	-10.5352	-0.6100	-9.0959	1.8907	0.372	1.310
	DCSGWO	-10.5330	-10.5330	-0.9110	-9.2509	1.5787	0.310	1.310
	FWAGWO	-2.8710	-2.8710	-0.8279	-2.5696	0.3489	0.360	1.279
	HAGWO	-2.8711	-2.4211	-0.8496	-2.3135	0.2769	0.751	1.498
	HSSASCA	-10.8569	-10.8569	0	-10.1080	1.5102	0.125	0.733




Objective space
Best score obtained so far
Iteration
SSA
PSO
MFO
SCA
DA
MVO
ALO
MGWO
DCSGWO
FWAGWO
HAGWO
HSSASCA
Objective space
Best score obtained so far
Iteration
SSA
PSO
MFO
SCA
DA
MVO
ALO
MGWO
DCSGWO
FWAGWO
HAGWO
HSSASCA
Objective space
Best score obtained so far
Iteration
SSA
PSO
MFO
SCA
DA
MVO
ALO
MGWO
DCSGWO
FWAGWO
HAGWO
HSSASCA
Objective space
Best score obtained so far
Iteration
SSA
PSO
MFO
SCA
DA
MVO
ALO
MGWO
DCSGWO
FWAGWO
HAGWO
HSSASCA
Objective space
Best score obtained so far
Iteration
SSA
PSO
MFO
SCA
DA
MVO
ALO
MGWO
DCSGWO
FWAGWO
HAGWO
HSSASCA
Objective space
Best score obtained so far
Iteration
SSA
PSO
MFO
SCA
DA
MVO
ALO
MGWO
DCSGWO
FWAGWO
HAGWO
HSSASCA

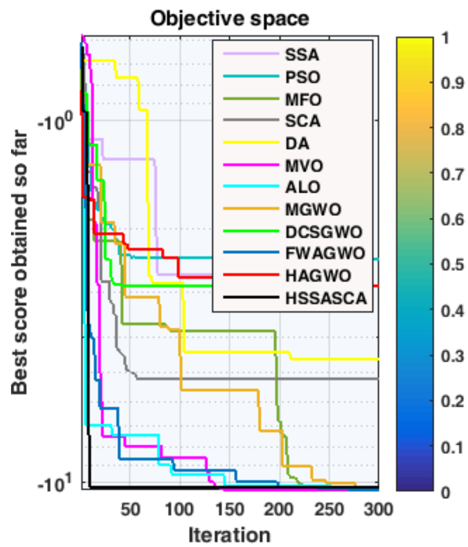


Figure 4. Convergence graphs of algorithms on fixed dimension multi-modal functions

6.8 Performance Assessment of Hybrid SSASCA

Concerning the consideration, the performance/or accuracy of the newly method has verified through applying the Wilcoxon signed ranks method for a superior assessment [14]. It is a non-parametric method that is utilized on two different samples, for finding the significance between them. On behalf of significance, we can easily choose the best one sample between them. In addition, the help of this method easily locates the significant difference of the behaviors of two meta-heuristics. The steps are shown as follows:

i. Select the data of two samples x_i and y_i.

ii. Calculate each and every paired difference: $d_i = x_i - y_i$.

iii. Take $|d_i| = |x_i - y_i|$.

iv. Rank the $|d_i|$, ignoring the negative sign's (i.e. allot rank-1 to the highest least valve of $|d_i|$ and rank-2 to the next, etc.)

v. Calculate the positive values $(|d_i| > 0, 1, -1)$

vi. Find the signed rank of using $|d_i| \times (|d_i| > 0, 1, -1)$

vii. Calculate $\sum_{i=1}^{n} R^{+}$ and $\sum_{i=1}^{n} R^{-}$ the sum of the ranks of the positive and negative, further check the total, $\sum_{i=1}^{n} R^{+} + \sum_{i=1}^{n} R^{-} = \frac{n(n+1)}{2}$, where n is the strength of the sample.

viii. Calculate $W = \max(R^{+}, R^{-})$, if two or more differences/observations may be equal, In that case we handle the tied problem by using $\frac{t^3 - t}{48}$, where t ~ is a total number of tied.

ix. Use normal approximation and calculate $\mu_W = \frac{n(n+1)}{4}$, $\sigma_W = \sqrt{\frac{n(n+1)(2n+1)}{24}}$ and we get

$$z = \frac{\max(R^{+}, R^{-}) - \mu_W}{\sqrt{\frac{n(n+1)(2n+1)}{24} - \frac{t^3 - t}{48}}}$$

x. Finally find out the p-value by using the value of z.

Here, these steps give the p-value after using the z-value. Here 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 HSSASCA is significantly superior to the other optimizer. If not, the obtained improvements are not statistically significant. The obtained p-values are presented in Tables 8-10.

Table 5. Results of the median values of the meta-heuristics on uni-modal functions

Benchmark functions	HSSASCA	SSA	PSO	MFO	SCA	DA	MVO	ALO	MGWO	DCSGWO	FWAGWO	HAGWO
	A_i	B_i	C_i	D_i	E_i	F_i	G_i	H_i	I_i	J_i	K_i	L_i
1.	25.0237	2.5703e+03	114.2949	1.7984e+04	2.3657e+04	2.9852e+03	1.5149e+03	4.1311e+03	0.0164	0.0144	5.1721e-21	3.6892e-21
2.	0.0021	3.5665	2.9704	8.4573	0.7450	3.3260	2.7573	2.0038	2.51149e-05	6.0572e-05	4.8253e-21	5.3105e-21
3.	0.2966	163.1911	15.3970	428.8130	1.8709e+03	2.3374e+03	67.6860	370.9822	2.1497e-04	6.9505e-04	3.8811e-09	3.6718e-09
4.	0.0051	3.8484	0.7070	23.5791	40.7947	30.4155	5.7345	3.6395	0.0099	0.0169	1.8890e-10	2.3853e-10
5.	8.9537	485.4100	150.5395	2.0845e+04	2.4126e05	3.2456e+04	3.3311e+03	273.0672	188.9512	8.9608	8.5523	7.2447
6.	0.8525	38.9567	0.6935	490.7669	103.2815	1.9025e+03	115.8474	266.6742	1.0130	2.0044	0.0310	0.0249
7.	0.0013	0.0122	1.3338	0.2139	0.0292	0.0768	0.0137	0.1274	0.0048	0.0062	0.0045	0.0022

Table 6. Results of the median values of the meta-heuristics on multi-modal functions

Benchmark functions	HSSASCA	SSA	PSO	MFO	SCA	DA	MVO	ALO	MGWO	DCSGWO	FWAGWO	HAGWO
	A_i	B_i	C_i	D_i	E_i	F_i	G_i	H_i				
1.	-3.4018e+03	-2.6326e+03	-2.0588e+03	-2.9877e+03	-2.0790e+03	-2.6500e+03	-2.5711e+03	-2.4516e+03				
2.	35.3485	35.0121	36.8090	59.8340	35.0813	78.9501	34.2582	11.5555				
3.	0.0034	3.8557	3.9293	10.7276	0.3835	9.8998	5.8717	3.1587				
4.	0.3222	1.3386	16.5049	3.4179	1.2529	2.5839	2.4355	0.7592				
5.	0.0906	6.2432	8.2092	9.1456	2.5472	17.2437	3.5896	7.4915				
6.	0.2277	3.9453	10.4227	2.5091e+04	4.1417	4.4351	5.6141	17.0240				

I_i	J_i	K_i	L_i
-1.4753e+03	-1.9437e+03	-2.0941e+03	-2.9867e+03
99.6272	104.9259	68.4197	71.7602
3.5123	3.4430	3.6518	7.9936e-15
95.7587	137.9988	142.6935	137.0655
1.0071	1.4176	2.8629	0.0434
1.2595	7.0615e+07	2.5434e+08	1.1278e+08

Table 7. Results of the median values of the meta-heuristics on fixed dimension multi-modal functions

Benchmark functions	**HSSASCA**	**SSA**	**PSO**	**MFO**	**SCA**	**DA**	**MVO**	**ALO**
	A_i	B_i	C_i	D_i	E_i	F_i	G_i	H_i
1.	12.6705	0.9821	1.9920	0.9980	2.0276	0.9983	0.9980	12.6705
2.	7.8992e-04	0.0257	0.0013	0.0022	9.1590e-04	4.0039	7.9026e-04	0.0088
3.	-1.0300	-1.0250	-1.0316	-1.0316	-1.0284	-1.0297	-1.0294	-1.0316
4.	3.0529	3.0201	3.0022	3.0000	3.0174	3.0128	3.0018	3.0001
5.	-3.7823	-3.8540	-3.8626	-3.8628	-3.8301	-7.8427	-8.8557	-3.8628
6.	-3.0889	-3.1547	-3.2250	-3.2026	-2.8229	-8.9627	-8.2420	-3.3217
7.	-2.3183	-5.4771	-9.1597	-5.0251	-0.8898	-11.1008	-9.1150	-5.0502
8.	-6.5662	-10.6754	-6.0874	-3.7210	-2.0809	-9.7860	-6.7621	-10.3705
9.	-2.2071	-6.7476	-3.7724	-2.8710	-0.9363	-8.3825	-2.8710	-10.5186

MGWO	**DCSGWO**	**FWAGWO**	**HAGWO**
I_i	J_i	K_i	L_i
8.8408	12.6705	3.9683	2.9821
4.6689e-04	0.0204	0.0073	3.2167e-04
-1.0316	-1.0316	-1.0316	-1.0316
3.0013	3.0019	3.0000	3.0018
-3.8604	-3.8624	-3.8624	-3.8619
-3.1494	-3.3174	-3.1740	-3.3006
-7.6340	-9.0251	-8.9404	-8.0504
-9.5044	-9.2157	-9.8197	-7.0860

Table 8. Wilcoxon test for comparison results in Table 5

Compared Techniques		Solution Evaluations						
Proposed variant	**Compared variant**	**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
HSSASCA	SSA	28	0	$A_i - B_i$ **to** L_i	2.366432	0.0180	yes	Yes
	PSO	27	1		2.197401	0.0280	yes	Yes
	MFO	28	0		2.366432	0.0180	yes	Yes
	SCA	28	0		2.366432	0.0180	yes	Yes
	DA	28	0		2.366432	0.0180	yes	Yes
	MVO	28	0		2.366432	0.0180	yes	Yes
	ALO	28	0		2.366432	0.0180	yes	Yes
	MGWO	16	12		0.338062	0.735363	no	no
	DCSGWO	15	13		0.169031	0.865797	no	no
	FWAGWO	0	26		-2.02837	0.02802	yes	yes
	HAGWO	1	27		-2.1974	0.02802	yes	yes

Table 9. Wilcoxon test for comparison results in Table 6

Compared Techniques		Solution Evaluations						
Proposed variant	Compared variant	Sum of rank ($\sum_{i=1}^{6} R_i^-$)	Sum of rank ($\sum_{i=1}^{6} R_i^+$)	Difference (D)	z-value	p-value	Accept ($p < 0.05$) H_1	Reject ($p < 0.05$) H_0
HSSASCA	SSA	20	1	$A_i - B_i$ to L_i	1.991741	0.0464	yes	yes
	PSO	21	0		2.201398	0.0277	yes	yes
	MFO	21	0		2.201398	0.0277	yes	yes
	SCA	20	1		1.991741	0.0464	yes	yes
	DA	21	0		2.201398	0.0277	yes	yes
	MVO	20	1		1.991741	0.0464	yes	Yes
	ALO	16	5		1.153113	0.2489	no	no
	MGWO	21	0		2.201398	0.027715	yes	yes
	DCSGWO	20	1		1.991741	0.046404	yes	yes
	FWAGWO	21	0		2.201398	0.027715	yes	yes
	HAGWO	18	3		1.572427	0.115858	no	no

Table 10. Wilcoxon test for comparison results in Table 7

Compared Techniques		Solution Evaluations						
Proposed variant	Compared variant	Sum of rank ($\sum_{i=1}^{9} R_i^-$)	Sum of rank ($\sum_{i=1}^{9} R_i^+$)	Difference (D)	z-value	p-value	Accept ($p < 0.05$) H_1	Reject ($p < 0.05$) H_0
HSSASCA	SSA	3	42	$A_i - B_i$ to L_i	2.310161	0.0209	yes	yes
	PSO	7	38		1.836282	0.0663	no	no
	MFO	9	36		1.599342	0.1097	no	no
	SCA	29	16		-0.77005	0.4413	no	no
	DA	5	40		2.073221	0.0382	yes	yes
	MVO	3	42		2.310161	0.0209	yes	yes
	ALO	4	41		2.191691	0.0284	yes	yes
	MGWO	0	45		2.66557	0.007687	yes	yes
	DCSGWO	4	41		2.191691	0.028408	yes	yes
	FWAGWO	2	43		2.42863	0.015157	yes	yes
	HAGWO	0	45		2.66557	0.007687	yes	yes

Hence, we implement the Wilcoxon test for the newly hybrid method against the several meta-heuristics that appears in Tables 5-7 and the obtained statistical solutions for Wilcoxon test is represented in Table 8-10. On this basis of these statistical results, it is realized that the HSSASCA method has better characteristics such that superiority of the optimal solution and strength of the global optima goal. Also, significant importance may be placed in local exploitation and global exploration. Results illustrate based on Wilcoxon test proved the better performance/or accuracy of the newly method among-others-in-comparison. Hence, the-obtained solutions by the HSSASCA method are-statistically superior and this has not happened by likelihood/or chance.

The convergence performance of Hybrid HSSASCA and comparative algorithms has been verified on basis of statistical and numerical results of uni-modal, multi-modal, fixed dimension multi-modal and constrained engineering functions in this section. The major motivation of the superior accuracy and working performance of the newly hybrid method lies after the optimal result create strategy induced by integrating the important SSA phase with SCA phase. In the presented method, the position of the each salp swarm in the entire group is improved by applying the position equations of sine and cosine; hence the superior optimal solutions have been tried to update based on the sine or cosine function, which means that the exploration ability/or capability can be even more powerful. Indeed, in HSSASCA algorithm, the sine and cosine phase of the SCA algorithm can helps SSA algorithm to find the superior optima value in the search space more rapidly and also refine the working accuracy/or performance and enhance convergence rate. During this research methodology, it certifies that the internal quality of this development is regulated to include the SCA as a local research strategy that accelerates the behavior of the transformation and without any amendments in the results of meta-heuristic avoids the systematic. The HSSASCA approach has proven its own significant accuracy during the search procedure, and also reduced the immature convergence inaccuracies of the SSA algorithm by using SCA phase. About the offered analytics, it certifies that the internal quality of this development is regulated to include the SCA as a local research strategy that accelerates convergence behavior, and ignores the meta-heuristics routine tour without any modifications in the results. Thus, there is an important precision and performance in the proposed HSSASCA algorithm and the inefficiency of the inaccessible transition of SSA approach is reduced powerfully.

Further, in order to do a fair comparison of a proposed algorithm with standard PSO, SSA, MFO, SCA, DA MVO, ALO, MGWO, DCSGWO, FWAGWO and HAGWO algorithms, mean and standard deviation for multiple runs have been reported. Here, a least statistic value indicates that the proposed algorithm is more robust, is capable to reproduce the solution with minimum discrepancy and has less dependency on initial population as comparison to other comparative approaches. In the assessment with other meta-heuristics, it seems that the new method performed more significantly. Further, the ability and capability of the proposed algorithm has been verified on the basis of taking a least time during searching of the optimal values in the search space of the functions. Results indicate that the proposed algorithm take a least time during searching the best and possible optimal solution of the problems outperforms than others. Hence, it can be concluded that the proposed algorithm is competent for searching the best optimal solution in the least time.

For testing the convergence performance of the algorithms such as PSO, SSA, MFO, SCA, DA MVO, ALO, MGWO, DCSGWO, FWAGWO, HAGWO and HSSASCA have been plotted the graphs with respect to number of iterations. In the graphs, x-axis represents the number of iterations and y-axis represents the best score obtained so far. Black line represents the performance of the proposed algorithm and on the basis of others colors we identifies the performance of rest of the algorithms. In these graphs, it can be easily seen that the proposed algorithm takes least time for convergence and search the best optimal solution in the least number of iterations as comparison to others. All the convergence graphs, numerical and statistical solutions of the proposed version assert that it is competent to improving the strength, accuracy, exploration, exploitation in dimensionality reduction tasks and reducing the complexity time of the standard versions. We trust that sine cosine algorithm helps to overcome the drawback of salp swarm.

To summarize, all experimental solutions reveal that the new hybrid method is more supportive in improving the competence of the standalone algorithms in the terms of global optimal result/or solution worth as well as computational efforts. Lastly, we expect that this method will motivate scientists and researchers in meta-heuristics and global optimization areas.

7. APPLICATIONS

7.1 Three-bar Truss Design

In this section, we apply the proposed method to-solve the three-bar-truss-design-function/or problem [26]. The main aim of this function is to achieve the minimum weight subjected to buck constraints, stress and deflection (see figure 5). The pseudo code of newly method has been run for searching the completive or best solution of this function on the setting of parameters mentioned in section 4. Three-bur truss design function has been solved by HSSASCA and compared with recent comparative algorithms. The mathematical explanation of this function has been described as following:

$$\text{Minimize } f(x) = \left(2\sqrt{2}x_1 + x_2\right) * r , \tag{10}$$

$$\text{Subject to} \quad l_1(x) = \left(\frac{\sqrt{2}x_1 + x_2}{\sqrt{2}x_1^2 + 2x_1x_2}\right) \times P - d \le 0 \tag{11}$$

$$l_2(x) = \left(\frac{x_2}{\sqrt{2}x_1^2 + 2x_1x_2} \right) \times P - d \le 0 \tag{12}$$

$$l_3(x) = \left(\frac{1}{\sqrt{2}x_2 + 2x_1x_2} \right) \times P - d \le 0 \tag{13}$$

where

$$0 \le x_1, x_2 \le 1,\ r = 100cm,\ M = 2KN/cm^2 \text{ and } d = 2KN/cm^2. \tag{14}$$

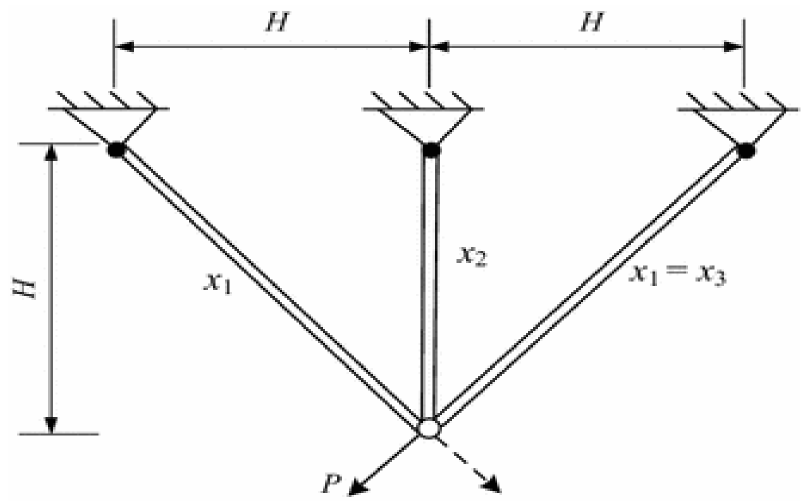


Figure 5. The three-bar truss design problem

Table 11. Best solutions of the three-bar truss design by different algorithms

Algorithm	x_1	x_2	$f(x)$
CS	0.78867	0.40902	263.9716
DEDS	0.78867513	0..40824828	263.8958434
SC	0.7886210370	0.4084013340	263.8958466
MBA	0.7885650	0.4085597	263.8958522
MFO	0.788244770931922	0.409466905784741	263.895979682
PSO-DE	0.7886751	0.4082482	263.8958433
LSA-SM	0.7886136	0.4084224	263.8958
HSSASCA	**0.7885923**	**0.4083256**	**263.8801451585985**

The obtained results of the algorithms have been described in Table 11 including the comparison between the proposed algorithm and others namely Cuckoo search algorithm (CS) [19], Differential-evolution-with-dynamic-stochastic-selection-(DEDS) [60], Society and civilization (CS) [40], Mine blast algorithm (MBA) [41], Moth Flame Optimizer (MFO) [33], Hybridizingparticleswarmoptimizationwithdifferentialevolution(PSO-DE) [24] and lightning search algorithm-simplex method (LSA-SM) [26]. Here, it-can-be easily seen that the newly method provides the minimum value (263.8801451585985) of the objective function of this problem in comparison to those of the others. Hence, it can be concluded that the HSSASCA is highly competent for this function than others.

7.2 Tension/Compression-Spring-Design

The Tension-Spring-Design-function/or Problem [26] has been solved in this section. The origin motive of this function is to reduce the weight of tension spring design. The pseudo code of newly method has been run for searching the completive or best solution of this function on the setting of parameters mentioned in section 4. A graphic view of Tension/Compression Spring design is shown in figure 6. The proposed algorithm and several comparative algorithms in the literatures have been applied for searching the best global optima result/or solution of this function. The mathematical formulas involved are as follows:

$$\text{Minimize } f(x) = (x_3+2)\,x_2 x_1^2 \tag{14}$$

$$\text{Subject to: minimum deflection: } (m_1(x)) = 1 - \frac{x_2^3 x_3}{71{,}785 x_1^4} \le 0 \tag{15}$$

$$\text{shear stress}(m_2(x)) = \frac{4x_2^2 - x_1 x_2}{12{,}566\left(x_2 x_1^3 - x_1^4\right)} + \frac{1}{5108 x_1^2} - 1 \le 0 \tag{16}$$

$$\text{shear stress}(m_2(x)) = \frac{4x_2^2 - x_1 x_2}{12{,}566\left(x_2 x_1^3 - x_1^4\right)} + \frac{1}{5108 x_1^2} - 1 \le 0 \tag{17}$$

$$\text{limit on outside diameter}(m_4(x)) = \frac{x_1 + x_2}{1.5} - 1 \le 0 \tag{18}$$

where $0.05 \le x_1 \le 0.25$, $0.25 \le x_2 \le 1.30$ and $2.00 \le x_3 \le 15.0$.

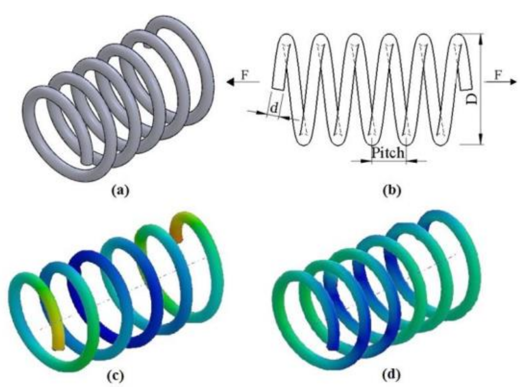


Figure 6. a) 3D view of the spring b) 2D view of the spring c) displacement heat map d) stress heat map

There are three design variables: wire diameter $d(x_1)$, mean coil diameter $D(x_2)$ and the number of active coils $N(x_3)$. Table 12 compares the best optimal solution obtained using HSSASCA algorithm with those reported in the literature. It can be easily seen that the minimum weight value (0.12533670077042) obtained by the proposed method is better than those of the others.

Table 12. Best solutions of the tension/compression spring design by different algorithms

Algorithm	x_1	x_2	x_3	$f(x)$
IHS [49]	0.05115438	0.34987116	12.0764321	0.0126706
ABC [38]	0.051749	0.358179	11.203763	0.012665
MFO [8]	0.051994457	0.36410932	10.868421862	0.0126669
GWO [13]	0.05169	0.356737	11.28885	0.012666
AFA [50]	0.0516674837	0.3561976945	11.3195613646	0.0126653049
BA [51]	0.05169	0.35673	11.2885	0.01267
LSA-SM [43]	0.05170453	0.3570899	11.26718	0.01266524
HSSASCA	**0.051591**	**0.3569458**	**11.19253**	**0.12533670077042**

7.3 Cantilever Beam Design

This Cantilever beam problem [26] has been solved in this section. The main objective is to find the best and possible minimum weight of this function. The section 4, presents all parameter settings on which a newly approach code has been run for searching the completive or best solution of this function. Figure 7, represents a schematic view of this function. The mathematical explanation of this function has been presented as following.

$$\text{Minimize: } f(x) = 0.0624\left(x_1 + x_2 + x_3 + x_4 + x_5\right) \tag{19}$$

$$\text{Subject to: } g(x) = \frac{61}{x_1^3} + \frac{37}{x_2^3} + \frac{19}{x_3^3} + \frac{7}{x_4^3} + \frac{1}{x_5^3} - 1 \le 0 \tag{20}$$

where $0.01 \le x_j \le 100;\ j = 1, 2, ..., 5.$

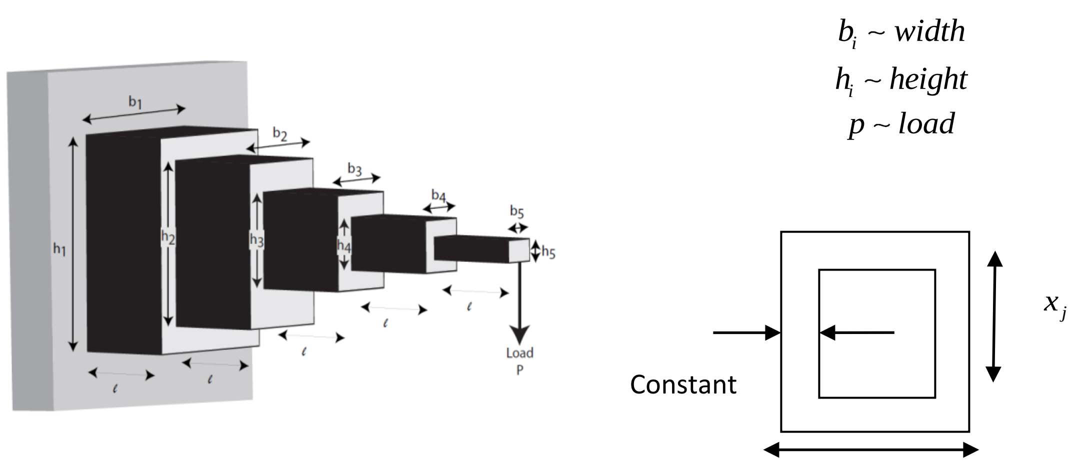


Figure 7. The cantilever beam design problem

x_j

For the purpose of comparison, we use the results of the recent comparative algorithms i.e. Method of moving asymptotes (MMA) [20], cuckoo search (CS) [20], generalized convex approximation -I (GCA-I) [10], generalized convex approximation-II (GCA-II) [10], symbiotic organisms search (SOS) [9], Moth Flame Optimizer (MFO) [33] and lightning search algorithm simplex method (LSA-SM) [26].Hence, after assessment of the results can be easily concluded that the HSSASCA algorithm provides in comparison to others the minimum value of weight of the Cantilever Beam Design problem. The minimum weight value of HSSASCA algorithm achieves the overall best design of 1.338896 (See Table 13).

Table 13. Best solutions of the Cantilever Beam Design by different algorithms

Algorithm	**I**	**II**	**III**	**IV**	**V**	**Minimum Weight Value**
	x_1	x_2	x_3	x_4	x_5	
MMA	6.0100	5.3000	4.4900	3.4900	2.1500	1.3400
CS	6.0089	5.3049	4.5023	3.5077	2.1504	1.33999
GCA-I	6.0100	5.3000	4.4900	3.4900	2.1500	1.3400
GCA-II	6.0100	5.3000	4.4900	3.4900	2.1500	1.3400
SOS	6.01878	5.30344	4.49587	3.49896	2.15564	1.33996
MFO	5.98487177	5.316726924	4.49733258	3.51361646	2.16162029	1.33998808
LSA-SM	6.021636	5.310859	4.490882	3.497403	2.152906	1.339958
HSSASCA	**6.012534**	**5.301452**	**4.491546**	**3.496582**	**2.154526**	**1.338896**

8. CONCLUSION

In this paper, we developed a new hybrid algorithm that integrates salp swarm (SSA) and sine cosine algorithms (SCA) for enhancing exploitation of the standard algorithms. The performance of the hybrid algorithm has been assessed and compared against seven feature selection approaches including SSA, PSO, MFO, SCA, DA, MVO, ALO, MGWO, DCSGWO, FWAGWO and HAGWO. Different criteria have been reported namely the minimum and maximum objective function value, standard deviation, best score, average, self and total time. On the basis of convergence performance of the proposed algorithm, we can conclude that the new hybrid approach is highly capable for maintaining balance amid exploitation and exploration. At the end, the proposed algorithm has been applied to solve three engineering design problems in reality namely three-bar truss, tension/compression spring and cantilever beam design problems. The experimental numerical and statistical results/or solutions reveal that the proposed hybrid method is better to other competitors in terms of convergence speed, quality of solutions and can serve as an efficient and capable computer aided tool for real life tasks with complex search area.

Future studies will investigate a new method to accelerate the speed of HSSASCA as well as apply it for solving other constrained nonlinear optimization functions [62-75].

ACKNOWLEDGEMENT

The authors are very grateful to the referees for their valuable suggestions, which helped to improve the quality of the paper significantly.

REFERENCES

[1]. Abido MA. Optimal power flow using tabu search algorithm. Electric Power Compon Syst. 2002;30:469–83.

[2]. Abtahi AR, Bijari A. A novel hybrid meta-heuristic technique applied to the well-known benchmark optimization problems. Journal of Industrial Engineering International. 2017:13(1):93-105.

[3]. Akay B, Karaboga D. Artificial bee colony algorithm for large-scale problems and engineering design optimization. Journal of Intelligent Manufacturing. 2012:23(4):1001–1014.

[4]. Bakirtzis AG, Biskas P, Zoumas CE, Petridis V. Optimal power flow by enhanced genetic algorithm. Power Syst, IEEE Trans.2002;17(2): 229–236.

[5]. Barraza J, Rodriguez L, Castillo O, Melin P, Valdez F. A New Hybridization Approach between the Fireworks Algorithm and Grey Wolf Optimizer Algorithm. Journal of Optimization. 2018: Article id: 6495362: 1-18.

[6]. Baykasoğlu A, Ozsoydan FB. Adaptive firefly algorithm with chaos for mechanical design optimization problems. Applied Soft Computing. 2015:l(36):152–164.

[7]. Ben-Tal A, El haoui, L, Nemirovski, A. Robust Optimization. Princeton Series in Applied Mathematics. Princeton University Press. 2009:9-16.

[8]. Bouchekara HREH. Optimal power flow using black-hole-based optimization approach. Appl. Soft Comput. 2014;24: 879–888.

[9]. Cheng MY, Prayogo D. Symbiotic organisms search: a new meta-heuristic optimization algorithm. Computers & Structures. 2014:139: 98–112.

[10]. Chickermane H, Gea HC. Structural optimization using a new local approximation method. International Journal for Numerical Methods in Engineering. 1996:39(5):829–846.

[11]. Chowdhury BH. Towards the concept of integrated security: optimal dispatch under static and dynamic security constraints. Electric Power Syst, Research.1992; 25:213-225.

[12]. Chung TS and Li YZ. A hybrid GA approaches for OPF with consideration of FACTS devices. IEEE Power Engineering Review. 2001: 47-50.

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

[14]. Derrac J, García S, Molina D, Herrera F. A practical tutorial on the use of nonparametric statistical tests as a methodology for comparing evolutionary and swarm intelligence algorithms. Swarm and Evolutionary Computation. 2011:1(1):3-18.

[15]. Duman S, Güvenç U, Sönmez Y, Yörükeren N. Optimal power flow using gravitational search

algorithm. Energy Convers, Manag. 2012;59:86–95.
[16]. Faris H, Mafarja MM, Heidari AA, Aljarah I, Ala'M AZ, Mirjalili S, Fujita H. An Efficient Binary Salp Swarm Algorithm with Crossover Scheme for Feature Selection Problems. Knowledge-Based Systems.2018.
[17]. Fouad A. A Hybrid Grey Wolf Optimizer and genetic algorithm for minimizing potential energy function. Memetic Computing. 2017:9(9):1-13.
[18]. Gandomi AH, Yang XS, Alavi AH, Talatahari H. Bat algorithm for constrained optimization tasks. Neural Computing and Applications. 2013:22(6):1239–1255.
[19]. Gandomi AH, Yang XS, Alavi AH. Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Engineering with Computers. 2013:29(1):17–35.
[20]. Gandomi AH, Yang XS, Alavi AH. Cuckoo search algorithm: a metaheuristic approach to solve structural optimization problems. Engineering with Computers. 2013:29(1):17–35.
[21]. Hsun LR, Ren TS, Tone CY. Tseng Wan-Tsun. Optimal power flow by a fuzzy based hybrid particle swarm optimization approach. Electr Power Syst Res 2011;81(7):1466–74.
[22]. Kalaiselvi K, Kumar V, Chandrasekar K. Enhanced Genetic Algorithm for Optimal Electric Power Flow using TCSC and TCPS. Proc. World. 2010.(II)
[23]. Kennedy J, Eberhart RC. Particle Swarm Optimization. In: Proceedings of IEEE International Conference on Neural Networks,.1995:1942–1948.
[24]. Liu H, Cai Z, Wang Y. Hybridizing particle swarm optimization with differential evolution for constrained numerical and engineering optimization. Applied Soft Computing, 2010:10(2): 629–640.
[25]. Liu H, Hua G, Yin H, Xu Y. An Intelligent Grey Wolf Optimizer Algorithm for Distributed Compressed Sensing. Computational Intelligence and Neuroscience. 2018: Article id: 1723191: 1-10.
[26]. Lu Y, Zhou Y, Wu X. A Hybrid Lightning Search Algorithm-Simplex Method for Global Optimization. Discrete Dynamics in Nature and Society. 2017: 2017(2017):id: 8342694:1-23.
[27]. Mahdavi M, Fesanghary M, Damangir E. An improved harmony search algorithm for solving optimization problems. Applied Mathematics and Computation. 2007:188(2):1567–1579.
[28]. Mirjalili S, Gandomi AH, Mirjalili SZ, Saremi S, Faris H, Mirjalili SM. Salp swarm algorithm: a bio-inspired optimizer for engineering design problems. Advances in Engineering Software. 2017:114:163–191.
[29]. Mirjalili S, Mirjalili SM, Hatamlou A. Multi-Verse Optimizer: a nature-inspired algorithm for global optimization. Neural Computing and Applications.2016; 2:495-513.
[30]. Mirjalili S, Mirjalili SM, Lewis A. Grey Wolf Optimization. Advance in Engineering Software. 2014;69:46-61.
[31]. Mirjalili S. Dragonfly algorithm: a new meta-heuristic optimization technique for solving single-objective, discrete, and multi-objective problems. Neural Computing and Applications. 2016;4:1053-1073.
[32]. Mirjalili S. Grasshopper Optimization Algorithm: Theory and application. Advances in Engineering Software. 2016;105:30-47.
[33]. Mirjalili S. Moth-flame optimization algorithm: A novel nature-inspired heuristic paradigm. Knowlede-Based System, Elsevier. 2015;89:228-249.
[34]. Mirjalili S. SCA: A Sine Cosine Algorithm for solving optimization problems. Knowlede-Based System, Elsevier. 2016;96:120-133.
[35]. Mirjalili S. The Whale Optimization Algorithm. Advances in Engineering Software. 2016;9:51-67.
[36]. Mirjalili, S. The Ant Lion Optimizer. Advances in Engineering Software. Elsevier. 2015; 83:80-98.
[37]. Mukherjee A, Mukherjee V. Solution of optimal power flow using chaotic krill herd algorithm. Chaos. Solutions & Fractals. 2015;78:10–21.
[38]. Pandiri, V., & Singh, A. A swarm intelligence approach for the colored traveling salesman problem. Applied Intelligence. 2018: 1-17. DOI: 10.1007/s10489-018-1216-0.
[39]. Rao MR, Babu NVN. Optimal Power Flow Using Cuckoo Optimization Algorithm. Ijareeie.2013:4213–4218.
[40]. Ray T, Liew KM. Society and civilization: an optimization algorithm based on the simulation of social behavior. IEEE Transactions on Evolutionary Computation. 2003:7(4):386–396.
[41]. Sadollah A, Bahreininejad A, Eskandar H, Hamdi M. Mine blast algorithm: a new population based algorithm for solving constrained engineering optimization problems. Applied Soft

Computing.2013:13(5): 2592–2612.

[42]. Sarbazfard S, Jafarian A. A Hybrid Algorithm based on Firefly Algorithm and Differential Evolution for Global Optimization. International Journal of Advanced Computer Science and Applications. 2016:7(6):95-106.

[43]. Sayed GI, Khoriba G, Haggag MH. A novel chaotic salp swarm algorithm for global optimization and feature selection. Applied Intelligence. 2018:6:1-20.

[44]. Sayed, G. I., Khoriba, G., & Haggag, M. H. A novel chaotic salp swarm algorithm for global optimization and feature selection. Applied Intelligence. 2018: 1-20. DOI: 10.1007/s10489-018-1158-6.

[45]. Simon D. Biogeography-based optimization. IEEE Transaction on Evolutionary Computation. 2008;12(6):702-713.

[46]. Singh N, Hachimi H. A New Hybrid Whale Optimizer Algorithm with Mean Strategy of Grey Wolf Optimizer for Global Optimization. Mathematical and Computational Applications.2018:23(14):1-32.

[47]. Singh N, Singh S, Singh SB. A New Hybrid MGBPSO-GSA Variant for Improving Function Optimization Solution in Search Space. Evolutionary Bioinformatics. 2017:13:1-13.

[48]. Singh N, Singh S, Singh SB. Half Mean Particle Swarm Optimization Algorithm. International Journal of Scientific & Engineering Research.2012:3(8):1-9.

[49]. Singh N, Singh SB. A Modified Mean Grey Wolf Optimization Approach for Benchmark and Biomedical Problems. Evolutionary Bioinformatics.2017:13:1-28.

[50]. Singh N, Singh SB. Hybrid Algorithm of Particle Swarm Optimization and Grey Wolf Optimizer for Improving Convergence Performance. Journal of Applied Mathematics.2017:ID 2030489:1-15.

[51]. Singh N, Singh SB. Personal Best Position Particle Swarm Optimization. Journal of Applied Computer Science & Mathematics. 2012:12(6):69-76.

[52]. Singh N, Singh, SB. One Half Global Best Position Particle Swarm Optimization Algorithm. International Journal of Scientific & Engineering Research. 2011:2(8):1-10.

[53]. Singh N. A Modified Variant of Grey Wolf Optimizer. International Journal of Science & Technology, Scientia Iranica. 2018. In press: http://scientiairanica.sharif.edu/?_action=article&keywords=A+Modified+Variant+of+Grey+Wolf+Optimizer

[54]. Sinsupan N, Leeton U, Kulworawanichpong T. Application of Harmony Search to Optimal Power Flow Problems 2010:219–222.

[55]. Soares J, Sousa T. Vale ZA, Morais H, Faria P. Ant colony search algorithm for the optimal power flow problem. IEEE Power Energy Soc, Gen, Meet. 2011: 1–8.

[56]. Tawhid MA, Savsani V. Multi-objective sine-cosine algorithm (MO-SCA) for multi-objective engineering design problems. Neural Computing and Applications. 2017:1-15.

[57]. Wolpert DH, Macready WG. No Free Lunch Theorems for Optimization. IEEE Transactions on Evolutionary Computation. 1997:1(1):67-82.

[58]. Yang, Y., Yang, B., &Niu, M. Adaptive infinite impulse response system identification using opposition based hybrid coral reefs optimization algorithm. Applied Intelligence. 2017: 1-18. DOI: 10.1007/s10489-017-1034-9.

[59]. Yuan X, Dai X, Zhao J, He Q. On a novel multi-swarm fruit fly optimization algorithm and its application. Applied Mathematics and Computation. 2014:233(3):260-271.

[60]. Zhang M, Luo W, Wang X. Differential evolution with dynamic stochastic selection for constrained optimization. Information Sciences.2008:178(15):3043–3074.

[61]. Zhao, X., Hwang, J. N., Fang, Z., & Wang, G. Gradient-based adaptive particle swarm optimizer with improved extremal optimization. Applied Intelligence. 2018: 1-14. DOI: 10.1007/s10489-018-1228-9.

[62]. Amal, L., Son, LH, Chabchoub, H. SGA: spatial GIS-based genetic algorithm for route optimization of municipal solid waste collection. Environmental Science and Pollution Research. 2018: 1-14. DOI: 10.1007/s11356-018-2826-0.

[63]. Son, LH, Fujita, H. (2018). Neural-Fuzzy with Representative Sets for Prediction of Student Performance. Applied Intelligence. 2018: 1-8.

[64]. Le, T., Le Son, H., Vo, M. T., Lee, M. Y., & Baik, S. W. A Cluster-Based Boosting Algorithm for Bankruptcy Prediction in a Highly Imbalanced Dataset. Symmetry (20738994), 2018: 10(7).

[65]. Son, LH, Chiclana, F., Kumar, R., Mittal, M., Khari, M., Chatterjee, J. M., & Baik, S. W. ARM–AMO: An efficient association rule mining algorithm based on animal migration

optimization. Knowledge-Based Systems, 154. 2018: 68-80.

[66]. Ngan, R. T., Son, LH, Cuong, B. C., & Ali, M. H-max distance measure of intuitionistic fuzzy sets in decision making. Applied Soft Computing, 69. 2018: 393-425.

[67]. Hemanth, D. J., Anitha, J., Son, LH. Brain signal based human emotion analysis by circular back propagation and Deep Kohonen Neural Networks. Computers & Electrical Engineering, 68. 2018: 170-180.

[68]. Pham, B. T., Son, LH, Hoang, T. A., Nguyen, D. M., & Bui, D. T. Prediction of shear strength of soft soil using machine learning methods. Catena, 166. 2018: 181-191.

[69]. Louati, A., Son, LH & Chabchoub, H. Smart routing for municipal solid waste collection: a heuristic approach. Journal of Ambient Intelligence and Humanized Computing. 2018: 1-20. DOI: 10.1007/s12652-018-0778-3.

[70]. Singh, K., Singh, K., Son, LH & Aziz, A. Congestion control in wireless sensor networks by hybrid multi-objective optimization algorithm. Computer Networks, 138. 2018: 90-107.

[71]. Giap, C. N., Son, L. H., & Chiclana, F. Dynamic structural neural network. Journal of Intelligent & Fuzzy Systems, 34. 2018: 2479 – 2490.

[72]. Ali, M., Son, LH, Thanh, N. D., & Van Minh, N. (2017). A neutrosophic recommender system for medical diagnosis based on algebraic neutrosophic measures. Applied Soft Computing. DOI: 10.1016/j.asoc.2017.10.012

[73]. Chuan, P. M., Son, LH, Ali, M., Khang, T. D., & Dey, N. Link prediction in co-authorship networks based on hybrid content similarity metric. Applied Intelligence, 48(8). 2018: 2470-2486.

[74]. Hemanth, D. J., Anitha, J., Popescu, D. E., & Son, L. H. A modified genetic algorithm for performance improvement of transform based image steganography systems. Journal of Intelligent & Fuzzy Systems, 35(1). 2018: 197-209.

[75]. Ali, M., Son, LH, Khan, M., & Tung, N. T. Segmentation of dental X-ray images in medical imaging using neutrosophic orthogonal matrices. Expert Systems with Applications, 91. 2018: 434-441.

[76]. Singh, N. and Singh, S.B. A novel hybrid GWO-SCA approach for optimization problem. Engineering Science and Technology, an International Journal, 20.2017:1586-1601.

[77]. Farnad, B. and Jafarian, A. Anew nature-inspired hybrid algorithm with a penalty method to solve constrained problem. International Journal of Computational Methods,15(08).2018.

[78]. Awais, M., Javaid, N., Mateen, A., Khan, N., Mohiuddin, A., Rehman and M.H.A. In the proceeding of 32nd International Conference on Advanced Information Networking and Applications, IEEE, 2018:882-891.

[79]. Marzie, A., Haddad, O.B. and Chu, X. Flower Pollination Algorithm (FPA). In Advanced Optimization by Nature-Inspired Algorithms, Springer, 2018:59-67.

[80]. Abdel-Basset, M., Gunasekaran, M., El-Shahat, D., Mirjalili, S. A hybrid whale optimization algorithm based on local search strategy for the permutation flow shop scheduling problem. Future Generation Computer Systems, 85.2018:129-145.

[81]. Zamli, K.Z., Din, F., S.Ahmed, B. and Bures, M. A hybrid Q-learning sine-cosine-based strategy for addressing the combinatorial test suite minimization problem. PLoS ONE, 13(5).2018.

[82]. Vidal, T., Crainic, T.G., Gendreau, M., Prins, C. A hybrid genetic algorithm with adaptive diversity management for a large class of vehicle routing problems with time- window. Computers & Operation Research, Elsevier, 40(1).2013:475-489.

[83]. Zheng, Y.J. Water wave optimization: A new nature-inspired metaheuristic. Computers & Operation Research, Elsevier, 55(1).2015: 1-11.

[84]. Yang, X.S. Nature-Inspired Optimization Algorithms. Book, Elsevier Science Publishers B. V. Amsterdam, The Netherlands, The Netherlands. 2014. https://dl.acm.org/citation.cfm?id=2655295.

[85]. Hu, C., Xia, Y. and Zhang, J. Adaptive Operator Quantum-Behaved Pigeon-Inspired Optimization Algorithm with Application to UAV Path Planning. Algorithms, MDPI. 12(1). 2018:1-16.

[86]. Talbi, E.G. A taxonomy of hybrid metaheuristics. Journal of Heuristics, 8(5).2002:541–564.

Appendix:

Table A. Uni-modal functions

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

Table B. Multi-modal functions

Function	Dim	Range	f_{min}
$F_8(x) = \sum_{i=1}^{n} -x_i \sin\left(\sqrt{\lvert x_i \rvert}\right)$	30	[-500,500]	-418.9829 $\times$ 5
$F_9(x) = \sum_{i=1}^{n} \left[x_i^2 - 10\cos(2\pi x_i) + 10 \right]$	30	[-5.12,5.12]	0
$F_{10}(x) = -20 \exp\left(-0.2 \sqrt{\frac{1}{n} \sum_{i=1}^{n} x_i^2} \right) - \exp\left(\frac{1}{n} \sum_{i=1}^{n} \cos\left(2\pi x_i\right) \right) + 20 + e$	30	[-32,32]	0
$F_{11}(x) = \frac{1}{4000} \Sigma_{i=1}^{n} x_i^2 - \prod_{i=1}^{n} \cos\left(\frac{x_i}{\sqrt{i}} \right) + 1$	30	[-600,600]	0

Function	Dim	Range	f_{min}
$F_{12}(x) = \frac{\pi}{n}\left\{10\sin(\pi y_i) + \sum_{i=1}^{n-1}(y_i - 1)^2\left[1 + 10\sin^2(\pi y_{i+1})\right] + (y_{n-1})^2\right\} + \sum_{i=1}^{n} u(x_i, 10, 100, 4)$	30	[-50,50]	0
$F_{13}(x) = 0.1\left\{\sin^2(3\pi x_i) + \sum_{i=1}^{n}(x_i - 1)^2\left[1 + \sin^2(3\pi x_i + 1)\right] + (x_n - 1)^2\left[1 + \sin^2(2\pi x_n)\right]\right\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	30	[-50,50]	0

$$y_i = 1 + \frac{x_i + 1}{4}$$

$$u(x_i, a, k, m) = \begin{cases} k(x_i - a)^m & x_i > a \\ 0 & -a < x_i < a \\ k(-x_i - a)^m & x_i < -a \end{cases}$$

Table C. Fixed-dimension multi-modal benchmark functions

Function	Dim	Range	f_{min}
$F_{14}(x) = \left(\frac{1}{500} + \sum_{j=1}^{25} \frac{1}{j + \sum_{i=1}^{2}(x_i - a_{ij})^6}\right)^{-1}$	2	[-65,65]	1
$F_{15}(x) = \sum_{i=1}^{11}\left[a_i - \frac{x_1(b_i^2 + b_i x_2)}{b_i^2 + b_i x_i + x_4}\right]^2$	4	[-5,5]	0.00030
$F_{16}(x) = 4x_1^2 - 2.1x_1^4 + \frac{1}{3}x_1^6 + x_1 x_2 - 4x_2^2 + 4x_2^4$	2	[-5,5]	-1.0316
$F_{17}(x) = \left[1 + (x_1 + x_2 + 1)^2\left(19 - 14x_1 + 3x_1^2 - 14x_2 + 6x_1x_2 + 3x_2^2\right)\right] \times \left[30 + (2x_1 - 3x_2)^2 \times \left(18 - 32x_1 + 12x_1^2 + 48x_2 - 36x_1x_2 + 27x_2^2\right)\right]$	2	[-2,2]	3
$F_{18}(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{3} a_{ij}(x_j - p_{ij})^2\right)$	3	[1,3]	-3.86
$F_{19}(x) = -\sum_{i=1}^{4} c_i \exp\left(-\sum_{j=1}^{6} a_{ij}(x_j - p_{ij})^2\right)$	6	[0,1]	-3.32

$F_{20}(x) = -\sum_{i=1}^{5}\left[(X-a_i)(X-a_i)^T + c_i\right]^{-1}$	4	[0,10]	-10.1532
$F_{21}(x) = -\sum_{i=1}^{7}\left[(X-a_i)(X-a_i)^T + c_i\right]^{-1}$	4	[0,10]	-10.4028
$F_{22}(x) = -\sum_{i=1}^{10}\left[(X-a_i)(X-a_i)^T + c_i\right]^{-1}$	4	[0,10]	-10.5363