Journal for Research in Applied Sciences and Biotechnology

www.jrasb.com

ISSN: 2583-4053

Volume-3 Issue-3 || June 2024 || PP. 38-46

https://doi.org/10.55544/jrasb.3.3.8

Evaluation the Performance of Sine Cosine Algorithm in Solving Pressure Vessel Engineering Design Problem

Ghulam Ali Sabery[1], Ghulam Hassan Danishyar[2] and Mohammad Arman Osmani[3]

[1,2]Assistant Professor, Department of Mathematics & Physics, Faculty of Engineering, Balkh University, Mazar-e-Sharif, Balkh, AFGHANISTAN.

[3]Assistant Professor, Department of Civil and Industrial Engineering, Faculty of Engineering, Balkh University, Mazar-e-Sharif, Balkh, AFGHANISTAN.

[1]Corresponding Author: gh.alisabery@gmail.com

www.jrasb.com || Vol. 3 No. 3 (2024): June Issue

Received: 14-05-2024 **Revised:** 24-05-2024 **Accepted:** 02-06-2024

ABSTRACT

The Sine Cosine Algorithm (SCA) is one of the population-based metaheuristic optimization algorithms inspired by the oscillation and convergence properties of sine and cosine functions. The SCA smoothly transits from exploration to exploitation using adaptive range change in the sine and cosine functions. On the other hand, pressure vessel design is a complex engineering structural optimization problem, which aims to find the best possible design for a vessel that can withstand high pressure. This typically involves optimizing the material, shape, and thickness of the vessel to minimize welding, the material, and forming cost while ensuring it meets safety and performance requirements. This paper evaluates the performance of SCA for solving pressure vessel design problems. The result produced by SCA is compared with the results obtained by other well-known metaheuristic optimization algorithms, namely; ABC, ACO, BBO, CMA-ES, CS, DE, GA, GSA, GWO, HSA, PSO, SSO, TLBO and TSA. The experimental results demonstrated that SCA provides a competitive solution to other metaheuristic optimization algorithms with the advantage of having a simple structured search equation. Moreover, the performance of SCA is checked by different numbers of populations and the results indicated that the best possible population size should be 30 and 40. In addition to this, the SCA search agent success rate is checked for different numbers of populations and results show that the search agent success rate do not exceed 4.2%.

Keywords- Metaheuristic Algorithm, Optimization, Pressure Vessel Design, Search Agent Success Rate, Sine Cosine Algorithm.

I. INTRODUCTION

Optimization refers to the process of finding the best possible solution to a problem or achieving the highest level of performance within given constraints. It involves maximizing or minimizing an objective function by adjusting variables or parameters. Optimization is paramount in many applications, including engineering, economics, computer science, business activities and industrial designs. Pressure vessels are complex engineering structures, subject to many uncertainties which are considered to be a challenging task for optimization algorithms. The aim of the pressure vessel engineering design problem involves finding the best possible design for a vessel that can lead to optimizing the material, shape, and thickness of the vessel to minimize welding, the material, and forming cost while ensuring it meets safety and performance requirements. Since many mechanical engineering optimization problem are non-linear, non-continuous, and non-differentiable with many constraints therefore, the classical optimization algorithms will not able to solve such kind of problem, so there is a serious need to employ metaheuristic optimization algorithms [1]. Nature-inspired/Metaheuristic optimization algorithms have gained significant attention in solving complex optimization problems across various domains. These algorithms are a class of computational methods that

ISSN: 2583-4053
Volume-3 Issue-3 || June 2024 || PP. 38-46

https://doi.org/10.55544/jrasb.3.3.8

mimic the behavior of natural systems to solve complex optimization problems. Nature-inspired/Metaheuristic optimization algorithms draw inspiration from the principles and processes observed in various natural phenomena, such as evolution, swarm intelligence, and the behavior of organisms. These algorithms are designed to explore and exploit search spaces efficiently by iteratively improving candidate solutions. They often involve population-based approaches, where a set of potential solutions (population) evolves over generations through iterative processes of selection, reproduction, and mutation [2]. Metaheuristic optimization algorithms can be classified into five classes as follows:

1. Evolutionary techniques take inspiration from biology. In evolutionary algorithms, there is an initial random population that evolves over generations to produce new solutions by means of crossover and mutation and eliminate the worst solutions in order to improve the fitness value. The most popular algorithms are Evolutionary Strategy (ES) [3], Genetic Algorithm (GA) [4], Differential Evolution (DE) [5], Biogeography-Based Optimization (BBO) [6], Genetic Programming (GP) [7], and Covariance Matrix Adaption Evolutionary Strategy (CMA-ES) [8].
2. The swarm intelligence technique takes inspiration from the behavior of social insects or animals. In swarm intelligence, every individual has its own intelligence and behavior, but the integration of the individuals gives more power to solve complex problems. The most popular algorithms are Particle Swarm Optimization (PSO) [9], Ant Colony Optimization Algorithm (ACO) [10], Artificial Bee Colony Algorithm (ABC) [11,12], Spider Monkey Optimization algorithm (SMO) [13], Grey Wolf Optimizer (GWO) [14], Firefly Algorithm (FA) [15], Cuckoo Search (CS) algorithm [16,17] and Cuckoo Optimization Algorithm (COA) [18].
3. Physics-based techniques that take inspiration from the rules governing a physical phenomenon. The most popular algorithms are Gravitational Search Algorithm (GSA) [19], Harmony Search Algorithm (HSA) [20], Simulated Annealing (SA) [21], Big-Bang Big-Crunch (BBBC) [22], Charged System Search (CSS) [23], and Central Force Optimization (CFO) [24].
4. Human techniques are inspired from human activities. Every individual does physical activities (body activities) that affect his performance and nonphysical activities like thinking and behavior (mind activities). The most popular algorithms are the Teaching-Learning Based Optimization algorithm (TLBO) [25], Harmony Search (HS) [26], Group Search Optimizer (GSO) [27], and Tabu (Taboo) Search (TS) [28].
5. Mathematics techniques are adopting geometric, trigonometric, analytic functions and other mathematical expressions in their search equations in order to direct the solutions towards the promising area of search space. The most popular algorithms are Sine-Cosine Optimization algorithm (SCA) [29], Spherical Search Optimizer (SSO) [30], The Arithmetic Optimization Algorithm (AOA) [31], Stochastic Fractal Search (SFS) [32], and Tangent Search Algorithm (TSA) [33].

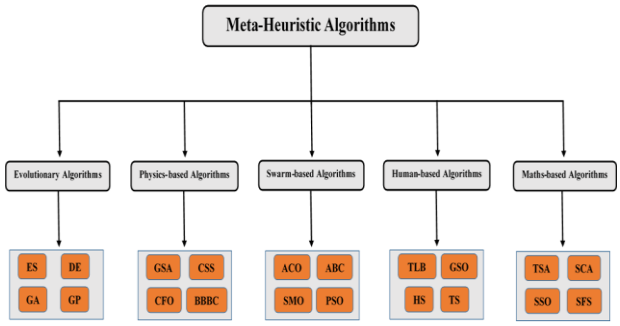

Figure 1. Classification of meta-heuristic optimization algorithms.

This paper aims to evaluate the performance and suitability of the Sine Cosine Algorithm in solving pressure vessel engineering design problems. The Sine Cosine Algorithm (SCA) is a metaheuristic optimization algorithm based on the trigonometric functions sine and cosine. It was proposed by Mirjalili, as a population-based algorithm for solving engineering optimization problems [29]. The algorithm is inspired by the oscillation and convergence properties of sine and cosine functions, and it has been shown to be effective in solving a wide range of optimization problems. This algorithm attracted researchers' attention and has received 3914 citations since it was published in 2016. Fig 2 shows the citation of SCA received for each year. This work evaluates the performance of SCA in three areas; solution quality, convergence ability and the search agent success rate. The Search agent success rate shows how many times a search agent gets updated successfully over the course of iterations. If the maximum number of iteration is T and a search agent, X_i, get updated successfully T_{is} times out of the course of iterations, the ratio $\frac{T_{is}}{T} \times 100\%$ is called search agent success rate. To evaluate the suitability of SCA, the performance of SCA is compared to other well-known metaheuristics optimization algorithm. The detailed description is given in the experimental results in section 5.

The rest of this paper is organized as follows: The second section describes the methodology. In the third section, a detailed introduction of the sine cosine optimization algorithm is provided. In the fourth section, a brief explanation of the pressure vessel engineering design problem, including the objective function and its constraints, is provided. In the fifth section, experimental results and solution analysis are presented and finally, the sixth section summarizes and concludes the work which has been done in this paper.

Journal for Research in Applied Sciences and Biotechnology

www.jrasb.com

ISSN: 2583-4053

Volume-3 Issue-3 || June 2024 || PP. 38-46

https://doi.org/10.55544/jrasb.3.3.8

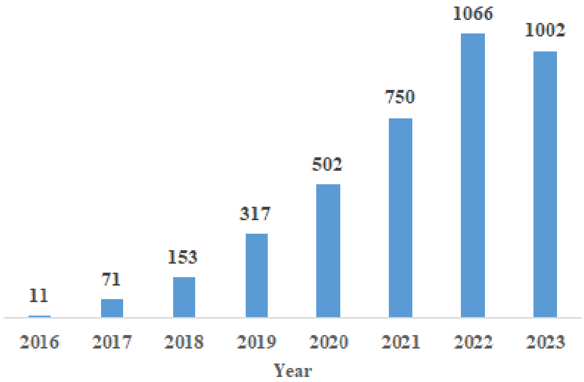

Figure 2. SCA citations over the years [38].

II. METHODOLOGY

This paper aims to evaluate the performance of the Sine Cosine Algorithm (SCA) when solving pressure vessel design problems. This work is done in two steps: In the first step, the detailed description of the Sine Cosine Algorithm and the pressure vessel design problem are provided in section 3 and section 4. Then, in the second step, the SCA with certain parameters is employed to solve the problem and other experiments are conducted to evaluate the performance of SCA. In this study, the experiments conducted using MATLAB language, and the detailed background experiment is provided in Table 1. In order to conduct the experiments, each algorithm runs 10 times independently to find the optimal solution, then the average of the best solution, the fitness values and search agent success rate out of ten runs stored in tables 3, 4, and 5. Moreover, in order to visualize the performance of SCA, the results produced be SCA is compared with the results obtained by other well-known metaheuristic optimization algorithms, namely, Genetic Algorithm (GA), Differential Evolution (DE), Teaching Learning Based Optimization Algorithm (TLBO), Grey Wolf Optimizer (GWO), Jaya Algorithm (JA), Spherical Search Optimization Algorithm (SSO), Covariance Matrix Adaption Evolutionary Strategy (CMA-ES), Artificial Bee Colony Algorithm (ABC), Ant Colony Optimization (ACO), Tangent Search Algorithm (TSA), Particle Swarm Optimization (PSO), Biogeography Based Optimization Algorithm (BBO), Gravitational Search Algorithm (GSA), Harmony Search Algorithm (HSA) and Cuckoo Search algorithm(CS). For ease of readability, the best solution obtained from the algorithms is highlighted in the boldface. At the end, the convergence curves of algorithms and search agents success rate are plotted.

III. SINE COSINE ALGORITHM (SCA)

The Sine Cosine Algorithm (SCA) is a metaheuristic optimization algorithm inspired by the sine and cosine functions behavior. It was proposed by Mirjalili [29], as an efficient and effective optimization technique for solving complex optimization problems. The algorithm is based on the concept of simulating the behavior of sine and cosine waves to explore the search space. It utilizes a population-based approach, where a set of candidate solutions, known as search agents, are iteratively updated to find the optimal solution. In SCA, each search agent represents a potential solution to the problem being solved. The algorithm starts by randomly initializing the positions of the search agent within the search space. The position of a search agent is represented by a vector of real numbers. During each iteration, the search agents' position updates using a mathematical equation based on sine and cosine functions. In SCA, the position of search agent is updated using the following search equation:

$$X_i^{t+1} = \{X_i^t + r_1 \times sin\, sin\,(r_2) \times |r_3 P_i^t - X_i^t| \quad r_4 < 0.5 \quad X_i^t + r_1 \times cos\, cos\,(r_2) \times |r_3 P_i^t - X_i^t| \quad r_4 \geq 0.5 \quad \ldots(1)$$

where X_i^t is the position of the current search agent in i^{th} dimension at t^{th} iteration, P_i^t is the position of the best solution (distention point) in i^{th} dimension at t^{th} iteration, $|\cdot|$ indicates the absolute value and $r_i, i = 1,2,3,4$ are uniformly distributed random numbers [29].

The random numbers, r_1, r_2, r_3 and r_4 are the main parameters in SCA. The parameters r_1 and r_3 are uniformly distributed random numbers between 0 and 2, the parameter r_2 is a uniformly distributed random number between 0 and 2π, and r_4 is a uniformly distributed random number between 0 and 1. When the parameter r_1 is greater than 1, the algorithm conducts an exploration search, otherwise the algorithm conducts an exploitation search. The parameter r_2 defines how far the movement should be towards or outwards the destination. The parameter r_3 provides random weights for global best position (distention) in order to stochastically emphasize when $r_3 > 1$ or deemphasize when $r_3 < 1$ the effect of desalination in defining the distance. Finally, the parameter r_4 equally switches between the sine and cosine components in Eqs.(1). The effect of Sine Cosine and random numbers are illustrated in Fig. 2. Fig 2 shows that the search equation of SCA is capable of making the search agents move toward or outward to global best position (destination). Although a two dimensional model is provided in Fig. 3, it should be noted that it can be extended in higher dimension. In SCA, the search agents are doing exploration when the parameter r_1 is greater than 1 and the search doing exploitation when it is less than 1. So the control parameter r_1 plays a crucial role in the global exploration, which controls the transition of the algorithm from exploration mode to exploitation mode. The control parameter r_1 adopts the linear decreasing method of Eqs. (2) to guide the algorithm transit from the exploration to the exploitation [29].

ISSN: 2583-4053
Volume-3 Issue-3 || June 2024 || PP. 38-46
https://doi.org/10.55544/jrasb.3.3.8

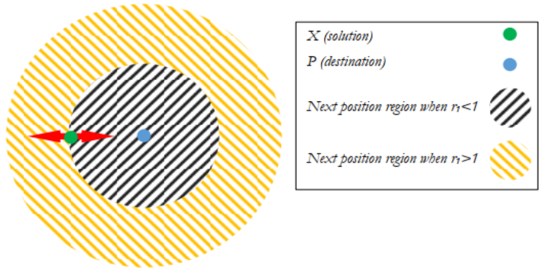

Figure 3. Effects of Sine, Cosine and random numbers in Eq. (1) on the next position [29].

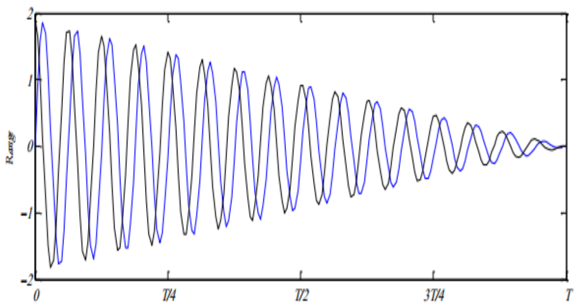

Figure 4. Decreasing pattern for range of Sine and Cosine [29].

The pseudo code of SCA is given as follows:

Algorithm: Pseudo code of SCA

1. Introduce the variable bounds and required parameters.
2. Generate the initial populations.
3. Calculate the fitness
While t <= T
Find the best solution
For i=1:Population size
Update the parameters r_1, r_2, r_3 and r_4
Update the position X_i using (1)
End if
Check the bounds
Calculate the fitness value
Apply the greedy selection
End for
End while
4. Report the best solution and fitness

IV. PRESSURE VESEL DESIGN PROBLEM

Pressure vessels are complex engineering structures, subject to many uncertainties which are used to store and transport fluids or gases at high pressures. A well-designed pressure vessel ensures the safety of workers and the surrounding environment by preventing leaks, ruptures, or explosions is less prone to failure, which minimizes downtime and production losses in industrial processes, can improve operational efficiency by minimizing energy loss, reducing maintenance requirements, extending the lifespan of the equipment, can help reduce material and manufacturing costs while ensuring optimal performance and longevity of the pressure vessel. Fig 5 demonstrates the schematic view of pressure vessels. The pressure vessel design optimization problem involves finding the best possible design for a vessel that can withstand high pressure. This typically involves optimizing the material, shape, and thickness of the vessel to minimize weight and cost while ensuring it meets safety and performance requirements. Key factors to consider in this optimization problem include:

- Material selection: Choosing the right material with high strength and corrosion resistance while minimizing weight and cost.
- Shape optimization: Determining the optimal shape of the vessel to distribute stress evenly and minimize material usage.
- Thickness optimization: Finding the optimal thickness of the vessel walls to ensure safety while minimizing weight and material cost.
- Performance requirements: Ensuring that the optimized design meets all necessary performance criteria, such as maximum allowable stress, fatigue life, and pressure containment.
- Cost considerations: Balancing the trade-off between material cost, manufacturing complexity, and weight to achieve an economically viable design.

To solve this optimization problem, engineers typically use advanced computational tools such as finite element analysis (FEA) and optimization algorithms to iteratively evaluate different design configurations and identify the best solution based on predefined objectives and constraints. Additionally, considering real-world factors such as manufacturing limitations, inspection requirements, and regulatory standards is crucial in achieving a practical and safe pressure vessel design.

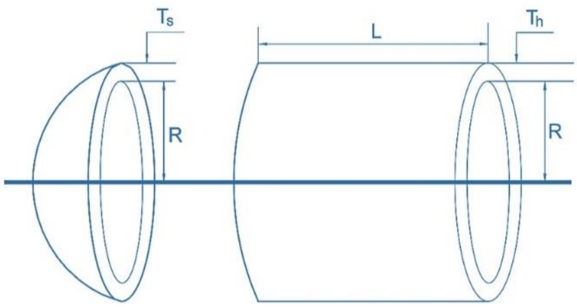

Figure 5. The schematic view of presser vessel [37].

Figure 5 illustrate the schematic view of pressure engineering vessel design problem. The purpose of this problem is to minimize the welding, the material, and forming cost [34]. There are four decision variables in the problem, namely, thickness of the head, T_h, thickness of the shell, T_s, length of the cylindrical section without

ISSN: 2583-4053
Volume-3 Issue-3 || June 2024 || PP. 38-46
https://doi.org/10.55544/jrasb.3.3.8

considering the head, L and the inner radius, R and containing four constraints while complete mathematical description of this constrained problem is provided in [35]. The best obtained solutions of the problem by optimization algorithms are presented in Table 3 and Table 4 and the convergence graph of the objective function, $f(\vec{x})$, is plotted in Fig. 6-8. The mathematical model of the problem is given as follows:

Consider $[x_1, x_2, x_3, x_4] = [T_s, T_h, R, L]$

$$f(\vec{x}) = 0.6224x_1x_3x_4 + 1.7781x_2x_3^2 + 3.1661x_1^2x_4 + 19.84x_1^2x_3$$

Subject to

$$g_1(\vec{x}) = -x_1 + 0.0193x_3 \le 0,$$
$$g_2(\vec{x}) = -x_2 + 0.00954x_3 \le 0,$$
$$g_3(\vec{x}) = -\pi x_3^2 x_4 - \frac{4}{3}\pi x_3^3 + 1296000 \le 0,$$
$$g_4(\vec{x}) = x_4 - 240 \le 0,$$

Variable range $0 \le x_1, x_2 \le 99, 10 \le x_3, x_4 \le 200.$

V. EXPERIMENTAL RESULTS

This section aims to validate the performance and suitability of the Sine Cosine Algorithm in solving pressure vessel design problems. This problem is a constrained engineering design which is considered to be a challenging task for optimization algorithms. To tackle the design constraints, a constraint handling method must be integrated into the optimization algorithms. There are several methods of constraint handling in the literature, but this work used the simplest constraint handling method called the death penalty to conduct the experiments [36]. The parameter settings and background details of the experiments are described in subsection 5.1 and the results provided by SCA and other metaheuristic optimization algorithms are provided in subsection 5.2.

a) Parameter Settings

In experiments in this section, the maximum number of iterations is 500 and the population size is 30. It's worth mentioning here that the population size in CMA-ES is $N =$ (4+round (3×$\ln \ln D$) ×5, which is equal to 50. Moreover, only the performance of SCA is checked for different numbers of populations. Where D is the dimension of the problem, which is 4. In order to conduct the experiments, each algorithm runs 10 times to find the optimal solution. In this study, the experiment was conducted using MATLAB language and a detailed background experiment is provided in Table 1. Other specific control parameters of the algorithms are presented in Table 2.

Table 1. Experimental background details.

Name	Settings
System Manufacturer	Acer
Processor	AMD A4-7210 (2.2GHz) APU with AMD Radeon R3 Graphics, 1800 Mhz, 4 Core(s), 4 Logical Processors (s)
HDD	1000GB
RAM	4GB
Operation System	Windows 10, x64-Based PC
Language	MATLAB R2014a

Table 2. The parameter settings of algorithms.

Algo	Parameter Settings
ABC	Limit = (population size × problem dimension)/2.
ACO	Sample size =40, Intensification Factor =0.5 and Deviation-Distance Ratio =1.
BBO	Emigration Rates are the same as original form of the algorithm, Habitat Keep Rate =0.4, Habitat Keep Size = round (Habitat Keep Rate× Population size), alpha =0.9 and mutation rate is 0.15. step size is [0.0099randn.*(Upper-Lower) bound] in mutation.
CMA-ES	Lambda= (4+round (3×log(problem dimension))) ×5, mu =lambda/2, and alpha_mu =2.
CS	Discovery rate of alien eggs =0.25, beta =3/2, and the rest are as original form of CS
DE	Crossover rate =0.9 and Scaling coefficient factoris 0.5.
GA	Uniform Crossover, Mutation rate =0.4, pc =beta =1 and sigma =1.6
GSA	ElitistCheck=1, Rpower=1, alpha=20, G_0=100 and Final Percentage =2.
GWO	a =2, Coefficient Vector $\vec{A}$ =2×a× $rand(0,1) - a$ and $\vec{A}$ =2× $rand(0,1)$
HSA	HMCR =0.8, PAR =0.9, BW=1 and BWF=0.99
PSO	Inertia weight =1, Dumping ratio of the inertia =0.99 and Acceleration Coefficients =2.
SCA	a =2, r_2 =2$\pi \times rand(0,1)$, r_3 =2× $rand(0,1)$ and r_4 =$rand(0,1)$
SSO	Tournament selection and x = 0.5+0.03 * randn
TLBO	Tf =round(1+rand(0,1))
TSA	Pswitch=0.3, Pesc=0.8 and the rests are the same as the original form of TSA

b) Results and Discussions

This section describes the experimental results obtained by SCA and other algorithms. To see the suitability of SCA, the solution produced by SCA is compared with the results obtained by the original form of well-known metaheuristic optimization algorithms,

ISSN: 2583-4053
Volume-3 Issue-3 || June 2024 || PP. 38-46
https://doi.org/10.55544/jrasb.3.3.8

namely; ABC, ACO, BBO, CMA-ES, CS, DE, GA, GSA, GWO, HSA, PSO, SSO, TLBO and TSA. Table 4 summarizes the comparison results between SCA and other algorithms. Table 3 shows that DE achieved the best solution among all algorithms, and SCA obtained the fourth-best solution after GWO, CS and SSO. Although SCA did not achieve the best results among all algorithms, still provides a competitive solution to other metaheuristic optimization algorithms with the advantage of having a simple structured search equation. Fig 8 and Fig 10 compares the convergence curve of SCA and other algorithms, which shows that SCA is smoothly convergent to the near optimum solution. At the beginning, SCA starts exploration and gradually transfers to exploitation, as we can see its convergence curve in the boxplot in Fig 9. Fig 5 compares the convergence curve of SCA for best, mean and worst fitness values. Initially, there is a big difference between the best, mean and worst fitness, but gradually they become close to each other and finally, almost they converge to the near optimum, which implies the convergence ability of SCA in solving such kind of problems.

Table 3. The optimum table of pressure vessel design problem.

Algo	**Decision Variables**				$f(\vec{x})$
	T_s	T_h	R	L	
ABC	53.414	43.323	59.826	118.526	6742.04
ACO	17.532	8.931	56.602	60.8887	6814.89
BBO	17.340	8.917	56.048	63.042	6840.89
CMAES	15.080	8.370	47.118	136.588	6932.20
CS	12.961	7.190	42.392	173.519	6070.14
DE	**13.024**	**6.968**	**42.098**	**176.636**	**6059.71**
GA	16.196	8.072	52.351	86.1028	6531.35
GSA	16.148	12.139	50.149	98.2968	7671.27
GWO	13.076	6.937	42.4143	173.147	6064.99
HAS	19.312	9.356	56.705	58.280	7452.66
JA	14.204	7.432	45.235	150.894	6310.27
PSO	16.923	8.360	53.538	86.768	6735.20
SCA	13.872	7.415	44.246	156.280	6240.96
SSO	13.772	7.177	44.007	156.230	6159.71
TLBO	14.736	7.704	47.269	136.561	6321.79
TSA	17.970	9.015	57.438	62.2124	6987.91

Table 4 summarizes the solutions obtained by SCA by different numbers of populations (N), which are 5, 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100. This experiment is launched to evaluate the performance of SCA for different numbers of populations and also to find the best possible population size for SCA to solve such kind of problem. It is well known that the population size, in an algorithm is a quantitative parameter. Therefore, if the number of populations are increased, it is obvious that the algorithms will provide a better solutions. However, we need to find the best possible population size for SCA in order to prevent time-consuming, extra-functional evolution and find the best solution. Hence, according to the Table 4, the best possible population size should be between 30 and 40 for SCA to solve such kind of problems. Fig 6 compares the convergence graph of SCA with different numbers of populations, 5, 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100.

Table 4. The optimum table of the problem obtained by SCA for different number of population.

Pop	**Decision Variables**				$f(\vec{x})$
	T_s	T_h	R	L	
$N=5$	15.593	8.3299	48.4690	125.672	6736.219
$N=10$	13.670	7.2278	43.3166	165.681	6260.783
$N=20$	13.657	7.1859	43.1133	166.461	6223.428
$N=30$	13.872	7.4158	44.2461	156.280	6240.962
$N=40$	**13.421**	**6.9667**	**43.0076**	**167.35**	**6138.421**
$N=50$	13.140	7.1701	42.297	175.930	6150.086
$N=60$	13.677	7.0951	43.6583	159.976	6146.684
$N=70$	13.270	6.9318	42.543	172.739	6148.738
$N=80$	13.235	7.0495	42.334	174.937	6129.074
$N=90$	13.153	7.0915	42.436	173.727	6117.679
$N=100$	13.361	7.1144	42.761	170.001	6120.564

Table 5 summarizes the search agent success rate of SCA, GWO, Jaya, SSO and DE for different number of populations. At each iteration in the algorithms, the search agents trying to get updated, but because of the greedy selection in many algorithms, they fail to update successfully. Therefore, there is a need to check how many times a search agent gets updated during the course of iteration, and that demonstrates how dynamic an algorithm is. In order to calculate the search agent success rate, we assigned a trial counter to each search agent. If the search agent gets updated successfully, the trial counter increases by one, otherwise it will remain the same. If the maximum number of iteration is T and a search agent gets updated T_{is} times out the course of iteration, the ratio $\frac{T_{is}}{T} \times 100\%$ is called search agent

ISSN: 2583-4053

Volume-3 Issue-3 || June 2024 || PP. 38-46

https://doi.org/10.55544/jrasb.3.3.8

success rate. In Fig 9 the horizontal line demonstrates the population size and the vertical line shows the success rates of algorithms including SCA. Table 5 shows that, the SCA search agent success rate does not exceed 4.2% , while the DE search agent success rate is almost 16% on average. Fig 9 compares the SCA, GWO, SSO, Jaya, and DE Search agent success rate for different population sizes.

Table 5. Search agent success rate

Pop	Optimization Algorithms				
	SCA	GWO	Jaya	SSO	DE
$N = 5$	4.178	3.576	28.08	4.008	13.54
$N = 10$	3.914	3.496	12.77	4.164	25.85
$N = 20$	3.898	2.955	8.628	4.179	17.155
$N = 30$	3.784	3.046	4.892	4.064	16.18
$N = 40$	3.928	2.913	3.489	4.221	15.83
$N = 50$	3.802	2.8372	3.584	4.1388	15.57
$N = 60$	3.764	2.798	2.972	4.1503	15.43
$N = 70$	3.806	2.644	1.894	4.36	15.46
$N = 80$	3.762	2.715	2.001	4.069	15.21
$N = 90$	3.802	2.613	2.057	4.191	15.30
$N = 100$	3.7834	2.6044	1.9772	4.1136	15.4038

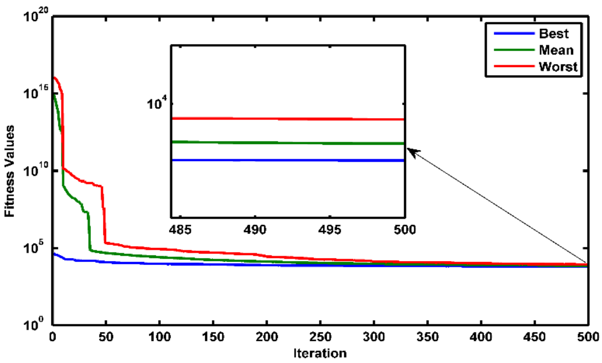

Figure 6. Convergence curve of SCA

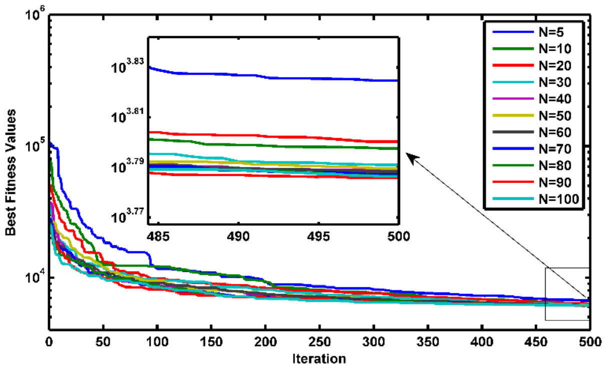

Figure 7. Convergence curve of SCA for different number of populations

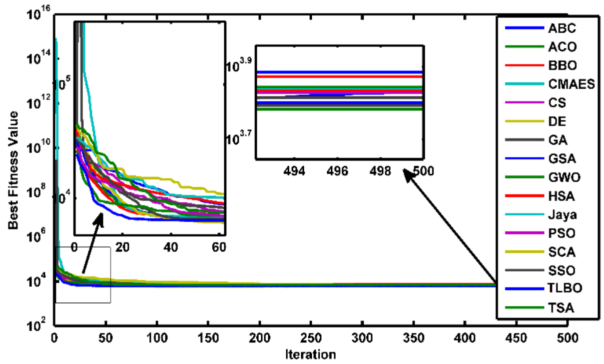

Figure 8. Convergence curve of algorithms

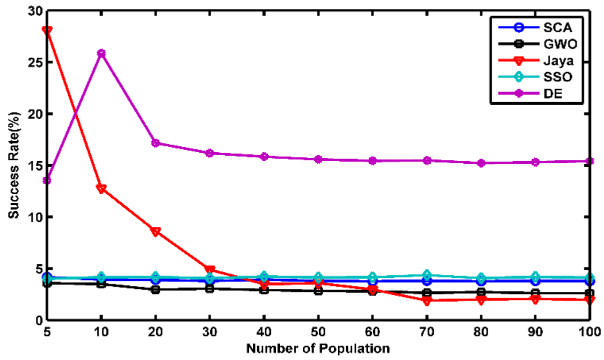

Figure 9. SCA search agent success rate

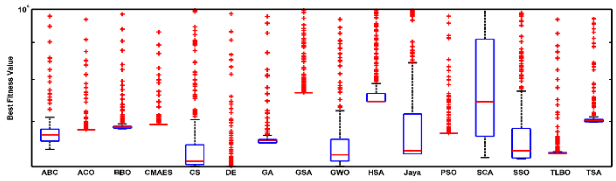

Figure 10. The boxplot of finesses provided by algorithms.

VI. CONCLUSION

This paper evaluated the performance of SCA in solving pressure vessel engineering design problem and the result produced by SCA is compared with the results obtained by other well-known metaheuristic optimization algorithms, namely; ABC, ACO, BBO, CMA-ES, CS, DE, GA, GSA, GWO, HSA, PSO, SSO, TLBO and TSA. The experimental results show that DE achieved the best solution among all algorithms, and SCA obtained the fourth-best solution after GWO, CS and SSO. Although SCA did not achieve the best results among all algorithms, still it provided a competitive solution to other metaheuristic optimization algorithms with the advantage of having a simple structured search equation. Moreover, the performance of SCA in solving the problem is checked by different numbers of populations and the experimental results recommends, the population size should be between 30 and 40. It is obvious if the population size increases, the algorithm will obtain a better solution, but it will impose extra function evaluation and time-work on the algorithm. Therefore,

ISSN: 2583-4053
Volume-3 Issue-3 || June 2024 || PP. 38-46
https://doi.org/10.55544/jrasb.3.3.8

SCA will have the best performance takes place for the population size between 30 and 40. In addition, the SCA search agent success rate is checked by different number of populations, which are 5, 10, 20, 30, 40, 50, 60, 70, 80, 90 and 100, and the experimental results show that, the SCA search agent success rate does exceed 4.2%, and it indicates that SCA is not as dynamic as DE. However, it has the ability to provide a competitive solution to other metaheuristic optimization algorithms. Overall, having a very simple search equation with controlled exploration and exploitation ability, makes the SCA look special.

REFERENCES

[1] Bansal, Jagdish Chand, Pramod Kumar Singh, and Nikhil R. Pal, eds. Evolutionary and swarm intelligence algorithms. Vol. 779. Cham: Springer, 2019.

[2] Yang, Xin-She. "Nature-inspired optimization algorithms: Challenges and open problems." Journal of Computational Science 46 (2020): 101104. https://doi.org/10.1016/j.jocs.2020.101104.

[3] Rechenberg, Ingo. "Evolutionsstrategie." Optimierung technischer Systeme nach Prinzipien derbiologischen Evolution (1973).

[4] Holland, John H. Adaptation in natural and artificial systems: an introductory analysis with applications to biology, control, and artificial intelligence. MIT press, 1992.

[5] Storn, Rainer, and Kenneth Price. "Differential evolution–a simple and efficient heuristic for global optimization over continuous spaces." Journal of global optimization 11 (1997): 341-359. https://doi.org/10.1023/A:1008202821328

[6] Simon, Dan. "Biogeography-based optimization." IEEE transactions on evolutionary computation 12, no. 6 (2008): 702-713. **DOI:** 10.1109/TEVC.2008.919004

[7] Koza, John R., and James P. Rice. "Automatic programming of robots using genetic programming." In AAAI, vol. 92, pp. 194-207. 1992.

[8] Hansen, Nikolaus, Andreas Ostermeier, and Andreas Gawelczyk. "On the Adaptation of Arbitrary Normal Mutation Distributions in Evolution Strategies: The Generating Set Adaptation." In ICGA, pp. 57-64. 1995.

[9] Kennedy, James, and Russell Eberhart. "Particle swarm optimization." In Proceedings of ICNN'95-international conference on neural networks, vol. 4, pp. 1942-1948. IEEE, 1995. **DOI:** 10.1109/ICNN.1995.488968

[10] Dorigo, M., and T. Stützle. "Ant Colony Optimization, Bradford Publisher." (2004).

[11] Karaboga, Dervis, and Bahriye Basturk. "A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm." Journal of global optimization 39 (2007): 459-471. https://doi.org/10.1007/s10898-007-9149-x

[12] Karaboga, Dervis. An idea based on honey bee swarm for numerical optimization. Vol. 200. Technical report-tr06, Erciyes university, engineering faculty, computer engineering department, 2005.

[13] Bansal, Jagdish Chand, Harish Sharma, Shimpi Singh Jadon, and Maurice Clerc. "Spider monkey optimization algorithm for numerical optimization." Memetic computing 6 (2014): 31-47. https://doi.org/10.1007/s12293-013-0128-0

[14] Mirjalili, Seyedali, Seyed Mohammad Mirjalili, and Andrew Lewis. "Grey wolf optimizer." Advances in engineering software 69 (2014): 46-61. https://doi.org/10.1016/j.advengsoft.2013.12.007

[15] Yang, Xin-She. "Firefly algorithm, stochastic test functions and design optimisation." International journal of bio-inspired computation 2, no. 2 (2010): 78-84. https://doi.org/10.1504/IJBIC.2010.032124

[16] Yang, Xin-She, and Suash Deb. "Cuckoo search via Lévy flights." In 2009 World congress on nature & biologically inspired computing (NaBIC), pp. 210-214. Ieee, 2009. **DOI:** 10.1109/NABIC.2009.5393690

[17] Yang, Xin-She, and Suash Deb. "Engineering optimisation by cuckoo search." International Journal of Mathematical Modelling and Numerical Optimisation 1, no. 4 (2010): 330-343. https://doi.org/10.1504/IJMMNO.2010.03543

[18] Rajabioun, Ramin. "Cuckoo optimization algorithm." Applied soft computing 11, no. 8 (2011): 5508-5518. https://doi.org/10.1016/j.asoc.2011.05.008

[19] Rashedi, Esmat, Hossein Nezamabadi-Pour, and Saeid Saryazdi. "GSA: a gravitational search algorithm." Information sciences 179, no. 13 (2009): 2232-2248. https://doi.org/10.1016/j.ins.2009.03.004

[20] Yang, Xin-She. "Harmony search as a metaheuristic algorithm." Music-inspired harmony search algorithm: theory and applications (2009): 1-14. https://doi.org/10.1007/978-3-642-00185-7_1

[21] Kirkpatrick, Scott, C. Daniel Gelatt Jr, and Mario P. Vecchi. "Optimization by simulated annealing." science 220, no. 4598 (1983): 671-680. DOI: 10.1126/science.220.4598.671

[22] Erol, Osman K., and Ibrahim Eksin. "A new optimization method: big bang–big crunch." Advances in engineering software 37, no. 2 (2006): 106-111.

ISSN: 2583-4053
Volume-3 Issue-3 || June 2024 || PP. 38-46
https://doi.org/10.55544/jrasb.3.3.8

https://doi.org/10.1016/j.advengsoft.2005.04.005

[23] Kaveh, A., and Siamak Talatahari. "A novel heuristic optimization method: charged system search." Acta mechanica 213, no. 3-4 (2010): 267-289. DOI: https://doi.org/10.1007/s00707-009-0270-4

[24] Formato, Richard. "Central force optimization: a new metaheuristic with applications in applied electromagnetics." Progress in electromagnetics research 77 (2007): 425-491. doi:10.2528/PIER07082403

[25] Rao, R. Venkata, Vimal J. Savsani, and D. P. Vakharia. "Teaching–learning-based optimization: an optimization method for continuous non-linear large scale problems." Information sciences 183, no. 1 (2012): 1-15. https://doi.org/10.1016/j.ins.2011.08.006

[26] Zong Woo Geem, Joong Hoon Kim, Loganathan GV. A New Heuristic Optimization Algorithm: Harmony Search. SIMULATION. 2001;76(2):60-68. doi:10.1177/003754970107600201

[27] S. He, Q. H. Wu and J. R. Saunders, "Group Search Optimizer: An Optimization Algorithm Inspired by Animal Searching Behavior," in IEEE Transactions on Evolutionary Computation, vol. 13, no. 5, pp. 973-990, Oct. 2009, doi: 10.1109/TEVC.2009.2011992.

[28] Fred Glover, (1989) Tabu Search—Part I. ORSA Journal on Computing 1(3):190-206. https://doi.org/10.1287/ijoc.1.3.190

[29] Mirjalili, Seyedali. "SCA: a sine cosine algorithm for solving optimization problems." Knowledge-based systems 96 (2016): 120-133. https://doi.org/10.1016/j.knosys.2015.12.022

[30] Zhao, J., Tang, D., Liu, Z. et al. Spherical search optimizer: a simple yet efficient meta-heuristic approach. Neural Comput & Applic 32, 9777–9808 (2020). https://doi.org/10.1007/s00521-019-04510-4

[31] Abualigah, Laith, Ali Diabat, Seyedali Mirjalili, Mohamed Abd Elaziz, and Amir H. Gandomi. "The arithmetic optimization algorithm." Computer methods in applied mechanics and engineering 376 (2021): 113609. https://doi.org/10.1016/j.cma.2020.113609

[32] Salimi, Hamid. "Stochastic fractal search: a powerful metaheuristic algorithm." Knowledge-based systems 75 (2015): 1-18. https://doi.org/10.1016/j.knosys.2014.07.025

[33] Layeb, A. Tangent search algorithm for solving optimization problems. Neural Comput & Applic 34, 8853–8884 (2022). https://doi.org/10.1007/s00521-022-06908-z

[34] Rao, Singiresu S. Engineering optimization: theory and practice. John Wiley & Sons, 2019.

[35] Gandomi, A.H., Yang, XS., Alavi, A.H. et al. Bat algorithm for constrained optimization tasks. Neural Comput & Applic 22, 1239–1255 (2013). https://doi.org/10.1007/s00521-012-1028-9

[36] Corne, David, Marco Dorigo, Fred Glover, Dipankar Dasgupta, Pablo Moscato, Riccardo Poli, and Kenneth V. Price, eds. New ideas in optimization. McGraw-Hill Ltd., UK, 1999.

[37] Varaee, Hesam, Naser Safaeian Hamzehkolaei, and Mahsa Safari. "A hybrid generalized reduced gradient-based particle swarm optimizer for constrained engineering optimization problems." Journal of Soft Computing in Civil Engineering 5, no. 2 (2021): 86-119.

[38] URL:https://scholar.google.com/citations?view_op=view_citation&hl=en&user=TJHmrREAAAAJ&citation_for_view=TJHmrREAAAAJ:k_IJM867U9cC (Accessed, December 4, 2023).

[39] URL: https://forgedcomponents.com/what-are-some-of-the-uses-for-pressure-vessels/ (Accessed, December 4, 2023).