Uluslararası İleri Doğa Bilimleri ve Mühendislik Araştırmaları Dergisi
Sayı 8, S. 216-222, 2, 2024

Araştırma Makalesi

International Journal of Advanced Natural Sciences and Engineering Researches
Volume 8, pp. 216-222, 2, 2024

Research Article

IJANSER

https://as-proceeding.com/index.php/ijanser
ISSN: 2980-0811

Performance Evaluation of the Nature Inspired Salp Swarm Algorithm on CEC 2017 Benchmark Problems

Dogukan Cevik[1], Okkes Tolga Altinoz[2]

[1]*Ankara University, Electrical and Electronics Engineering, Turkey, 19290417@ogrenci.ankara.edu.tr*
[2]*Ankara University, Electrical and Electronics Engineering, Turkey, taltinoz@ankara.edu.tr*

(Received: 04 March 2024, Accepted: 08 March 2024)

(4th International Conference on Innovative Academic Studies ICIAS 2024, March 12-13, 2024)

ATIF/REFERENCE: Cevik, D. & Altinoz, O. T. (2024). Performance Evaluation of the Nature Inspired Salp Swarm Algorithm on CEC 2017 Benchmark Problems. *International Journal of Advanced Natural Sciences and Engineering Researches, 8(2), 216-222.*

Abstract: Computational optimization algorithms are named according to the number of objectives which are single-objective; multi-objective and many-objective optimization algorithms. In addition, these algorithms can be classified according to their design differences. Single objective optimization algorithms can be classified into two categories; nature inspired and evolutionary algorithms. The nature inspired optimization algorithms design from observations on the nature especially behavior of the animal swarms. In literature there are many algorithms have been proposing to solve single objective optimization problems. Among many nature-inspired algorithms recently an algorithm called Salp Swarm Algorithm (SSA) is proposed. To evaluate the performance of this algorithm on a challenging problem; in this work, the effectiveness of the algorithm is evaluated with CEC 2017 benchmark functions. The obtained solutions are compared with other algorithms on the literature to clearly demonstrate the performance of SSA.

Keywords- Optimization, Algorithm, Salp, CEC 2017, Meta-heuristic Techniques, Optimal.

I. INTRODUCTION

Optimization can be defined as finding the best solution for the problem. This problem could be a mathematical formulation or a real-word application. Since the optimization is an important topic, there are many different approaches are proposed to solve optimization problems. Among them the computational optimization took attention of the engineering research since even the optimum could not be obtained an approximate solution can be calculated from these algorithms. The number of objectives for this approach changes the name of the problems set. If the size of the optimization problem is one the solution set called single objective optimization. Single objective optimization problems can be solved by evolutionary algorithm and nature inspired optimization algorithms.

Nature inspired algorithms are population-based algorithms that mimics the behavior of the physical phenomenon like behavior of the swarms in nature. The Salp Swarm Algorithm (SSA) is one of nature inspired optimizations algorithm which is proposed by Mirjalili, et al. in 2017. After proposal of this algorithm, it is possible to find the implementation of this algorithm on different engineering problems and improved version of the SSA. In [1], novel algorithm is proposed, and it is compared with SSA on CEC 2017 benchmark problems to clearly demonstrate the performance of the proposed algorithm. In that study the SSA is preferred as the reference results for comparing the performances.

In [6], the author proposed Salp Swarm and Grey Wolf Optimization-based technique for diagnosing breast cancer. The proposed algorithm applied to the Wisconsin Breast Cancer Dataset, and the model has an accuracy of 99.42%.
Similar study reported in [7]. The authors worked on extracting maximum photovoltaic output power and regulating the DC-bus voltage. To handle this control problem a nonlinear control algorithm sliding mode controller is evaluated. In addition, Hybrid SSA-PSO algorithm that is proposed in [7]; integrated on sliding mode controller. In the paper the results are compared with PSO, SSA, cuckoo search optimization (CSO), and grey wolf optimization (GWO). It is observed that SSA has an improve effect on this problem. Another power related topic is discussed for SSA in [10] where the diagnosis of faults in grid-connected photovoltaic (GCPV) system problem is solved with Supervised machine learning-based salp swarm algorithm. Findings of the research showed that the performance of the algorithm help to reach almost 99% accuracy. In [8], the improved version of SSA is applied to the economic dispatch problem where the renewable energy sources are included to the problem environment. The propose SSA-based algorithm is compared with Sine–cosine Algorithm, Whale Optimization Algorithm, and Back-search Algorithm. The results showed that the SSA presents better results for this engineering problems. And in [9] the feature selection problem for the data mining is investigated with the aid of SSA where multi-perspective initialization strategy, Newton interpolation inertia weight, improved followers' update model and cosine opposition-based learning (COBL) are proposed/integrated into the SSA for improving the performance. The feature selection problems and SSA implemented and compared on seven algorithms. Even SSA couldn't give the best results however the performance if it looks challenging among other algorithms.

When the papers related to SSA investigated, it can be observed that SSA clearly help to solve the engineering problem. However, from the literature still the performance of the SSA is not clear because the algorithm didn't compare on more challenging problems on a fair ground. For this reason, the SSA is compared with BIA, PSO, and SHADE algorithms on CEC 2017 benchmark problems [11] where these algorithms are more challenging and by this way it is possible to compare the performance of SSA with other well-known algorithms. This paper begins with the introduction section. Then The SSA algorithm will be discussed with a brief information with other algorithms BIA, PSO and SHADE. Then the benchmark problems will be demonstrated. Finally, the results of the implementations will be explainer. Finally, the conclusion of this research will be given.

II. SALP SWARM ALGORITHM

The Salp Swarm Algorithm (SSA) [2] is a novel nature-inspired optimization algorithm that draws inspiration from the remarkable swarming behavior of salps, marine organisms found in deep oceans. Salps exhibit an intriguing swarming behavior known as a "salp chain," where they form cooperative structures for efficient locomotion and foraging. This behavior, although not yet fully understood, has inspired the development of SSA as a powerful optimization algorithm. The mathematical model proposed for simulating the swarming behavior of salps is a key component of SSA. The salp population is divided into two groups: leaders and followers. The leaders guide the swarm, while followers emulate the movements of leaders. The positions of salps are defined in an n-dimensional search space, where n represents the number of variables in the optimization problem.

The position update equation for the leader salp involves the food source position and random coefficients. This equation ensures that the leader explores and exploits the search space effectively, balancing exploration and exploitation throughout the iterations. We begin by defining the position of salps in an ndimensional search space where n is the number of variables in each problem. We can store the position of all salps in a two-dimensional matrix called x and designate the food source as F in the search space.

$$x^1{}_j = F_j - c_1 \left((ub_j - lb_j) c_2 + lb_j \right), \quad if\ c_3 \geq 0.5 \qquad (1)$$
$$x^1{}_j = F_j + c_1 \left((ub_j - lb_j) c_2 + lb_j \right), \quad if\ c_3 < 0.5 \qquad (2)$$

Here, $x^1{}_j$ denotes the position of the leader salp in the jth dimension, an F_j denotes the position of the food source in the jth dimension. Further, ub_j and lb_j represent the upper and lower bounds of the jth dimension, respectively. Also, c_1, c_2, c_3 are random variables. As we can see, out of these coefficients, c_1 is most important as it helps in balancing between exploration and exploitation.

The positions of follower salps are updated using Newton's law of motion, facilitating gradual movements towards the leader salp.

$$x^{\mathrm{i}}{}_j = \frac{1}{2}(x^{\mathrm{i}}{}_j + x^{\mathrm{i}-1}{}_j) \ , where\ i \geq 2 \qquad (3)$$

where $x^{\mathrm{i}}{}_j$ shows the position of ith follower salp in jth dimension.

Briefly, we can summarize the flow diagram of the salp swarm algorithm as follows. We define the search space's dimension and the number of salps (individuals) exploring it. Each salp starts at a random position, and its performance is evaluated based on the objective function. The best-performing salp becomes the leader. The leader salp embarks on a random search for the "food source" (optimal solution). Its new location is evaluated, and if it improves performance, the leader stays put. Otherwise, it returns to its previous location. Each follower salp draws inspiration from the leader's location and its own past movement. It updates its position based on the distance to the leader and a random component. The new position is evaluated, and if it leads to better performance, the follower adopts it. Otherwise, it stays where it was. This cycle of leader exploration and follower adaptation continues until a predefined number of iterations occur or a desired level of performance is achieved. The final position and performance of the best-performing salp are considered the optimal solution found by the algorithm.

The SSA algorithm is compared with three other optimization algorithms: BIA, PSO and SHADE algorithms. Bison Algorithm (BIA) [1] is a nature inspired optimization algorithm. The idea of the BIA is based on exploring the area of the Bison's so that the behavior of the algorithm is divided into two parts, where divides the population divided into two groups, The First group contains stronger individuals where they exploit the search space. The second group contains weaker individuals which are explore the search space slowly. The algorithm begins with the calculation of the center of the swarm. The direction of the members is calculated with respect to this center. Then this direction is used to calculate the new position of the member. This is calculated with the first group of the Bisons. For the second group a different calculation is made. The running group direction is greatly dependent on the border of the search space and a random number.

Particle Swarm Optimization (PSO) is proposed by Eberhard and Kennedy in 1995 [12]. It is a swarm-based nature inspired optimization algorithm where it depends on the behavior of the animal swarms. The members have two properties which are position and velocity. The individual of the population updated their velocity and position. This update rules are based on the best member's position and each individual's best position through the iterations.

Differential Evolution is an evolutionary based single objective optimization algorithm. This algorithm haws a strong background but still like many of similar algorithms this algorithm lacks from the local optimum problem. The main reason of this problem for DE is the control parameters for the DE. However, the adaptivity of these parameters may help to get rid of that problem. Therefore, it is possible to adaptively change and record the best parameter history is an option to improve the performance of DE, that help to guide the selection of the parameters. Therefore, the algorithm called Success-History Based Parameter Adaptation for Differential Evolution (SHADE) algorithm is proposed [13] for this reason.

III. CEC 2017 BENCHMARK FUNCTIONS

Table 1. CEC 2017 Benchmark Functions

Type	**Id**	**Functions**	**Optimal**
Unimodal Functions	F1	Shifted and Rotated Bent Cigar Function	100
Unimodal Functions	F2	Shifted and Rotated Sum of Different Power Function	200
Unimodal Functions	F3	Shifted and Rotated Zakharov Function	300
Multimodal Functions	F4	Shifted and Rotated Rosenbrock's Function	400
Multimodal Functions	F5	Shifted and Rotated Rastrigin's Function	500
Multimodal Functions	F6	Shifted and Rotated Expanded Scaffer's F6 Function	600
Multimodal Functions	F7	Shifted and Rotated Lunacek Bi_Rastrigin Function	700
Multimodal Functions	F8	Shifted and Rotated Non-Continuous Rastrigin's Function	800
Multimodal Functions	F9	Shifted and Rotated Levy Function	900
Multimodal Functions	F10	Shifted and Rotated Schwefel's Function	1000
Hybrid Function	F11	Hybrid Function 1 (N=3)	1100
Hybrid Function	F12	Hybrid Function 2 (N=3)	1200
Hybrid Function	F13	Hybrid Function 3 (N=3)	1300
Hybrid Function	F14	Hybrid Function 4 (N=4)	1400
Hybrid Function	F15	Hybrid Function 5 (N=4)	1500
Hybrid Function	F16	Hybrid Function 6 (N=4)	1600
Hybrid Function	F17	Hybrid Function 6 (N=5)	1700
Hybrid Function	F18	Hybrid Function 6 (N=5)	1800
Hybrid Function	F19	Hybrid Function 6 (N=5)	1900
Hybrid Function	F20	Hybrid Function 6 (N=6)	2000
Composition Functions	F21	Composition Function 1 (N=3)	2100
Composition Functions	F22	Composition Function 2 (N=3)	2200
Composition Functions	F23	Composition Function 3 (N=4)	2300
Composition Functions	F24	Composition Function 4 (N=4)	2400
Composition Functions	F25	Composition Function 5 (N=5)	2500
Composition Functions	F26	Composition Function 6 (N=5)	2600
Composition Functions	F27	Composition Function 7 (N=6)	2700
Composition Functions	F28	Composition Function 8 (N=6)	2800
Composition Functions	F29	Composition Function 9 (N=3)	2900
Composition Functions	F30	Composition Function 10 (N=3)	3000
Search Range: $[-100,100]^D$			

Table 1 shows the thirty benchmark problems used in this research. These algorithms are initially proposed for CEC conference in [11]. The algorithms can be grouped as;

i)Shifted and rotated, Functions,
ii)Hybrid Functions,
iii)Composition Functions.

In addition, the optimum values of the benchmark problems are reported in the Table 1. For all benchmark problems has the 10 dimensions and the range of the search space is [-100,100].

IV. IMPLEMENTATION AND RESULTS

The SSA algorithm is the main algorithm that implemented on the CEC 2017 benchmark problems. To compare with other algorithms for a fair comparison in this research the functions are repeated15 times independently and, their statistic properties like mean and standard deviation are recorded on the Table 2. Therefore, the population is selected as 100 and maximum number of iterations is selected as 1000. To compare the performance of the SSA algorithm, SSA is compared with BIA, PSO and SHADE algorithm under the same computational resources.

Table 2. Results

Id	**SSA**	**BIA [1,3]***	**PSO [1,3]***	**SHADE [4,5]***
F1	1.40e+3 (1.62e+3)	1.6e+3 (1.8e+3)	3.9e+3 (5.1e+03)	**0 (0)**
F2	**235,59 (41,27)**	1.1e+11 (5.8e11)	5.0e+23 (3.5e+24)	1.051e+12 (4.36e+12)
F3	**300 (0)**	8.9e+1 (8.5e+1)	3.4e-4 (6.7e-4)	0 (0)
F4	**408,51 (16,99)**	1.6e+1 (2.5e+1)	9.3e+1 (2.8e+1)	4.92e+1 (4.69e+1)
F5	**521,88 (9,26)**	7.2e+1 (6.1e+1)	1.5e+2 (2.8e+1)	3.24e+1 (5.01)
F6	605,63 (4,05)	**2.8e-4 (1.0e-3)**	3.0e+1 (9.7e+0)	8.35e+4 (1e-3)
F7	**728,98 (10,36)**	1.7e+2 (3.1e+1)	1.0e+2 (2.2e+1)	8.06e+1 (3.74)
F8	**819,23 (6,90)**	8.00e+1 (5.8e+1)	1.0e+2 (1.9e+1)	5.49e+1 (7.73)
F9	901,86 (2,89)	5.5e+0 (9.0e+0)	2.2e+3 (9.6e+2)	**1.07 (9.34e-1)**
F10	**1.72e+3 (371,72)**	7.00e+3 (2.7e+2)	3.5e+3 (6.3e+2)	3.311e+3 (2.89e+2)
F11	1.16e+3 (38,17)	**2.9e+1 (2.3e+1)**	1.0e+2 (3.5e+1)	1.19e+2 (2.89e+2)
F12	1.73e+6 (1.91e+6)	**2.3e+4 (1.1e+4**)	6.5e+5 (2.0e+6)	5.1e+3 (2.85e+3)
F13	1.17e+4 (9.88e+3)	1.1e+4 (7.7e+3)	1.0e+5 (6.2e+5)	**2.6e+2 (1.43e+2)**
F14	**1.48e+3 (23,96)**	2.0e+3 (1.7e+3)	5.0e+3 (5.8e+3)	2.11e+2 (7.26e+1)
F15	1.92e+3 (322,16)	1.4e+3 (2.0e+3)	1.0e+4 (1.2e+4)	**3.2e+2 (1.39e+2)**
F16	**1.70e+3 (90,88)**	1.0e+3 (4.1e+2)	8.4e+2 (2.5e+2)	7.3e+2 (1.82e+2)
F17	**1.75e+3 (25,40)**	1.3e+2 (1.5e+2)	5.2e+2 (1.9e+2)	5.11e+2(1.08e+2)
F18	1.46e+4 (9.22e+3)	1.3e+5 (1.0e+5)	1.5e+5 (1.5e+5)	**1.85e+2 (1e+2)**
F19	**1.99e+3 (88,73)**	4.3e+3 (3.9e+3)	4.8e+3 (6.4e+3)	1.56e+2 (5.63e+1)
F20	**2.06e+3 (33,24)**	2.0e+2 (1.2e+2)	1.8e+2 (1.3e+2)	3.32e+2 (1.19e+2)
F21	**2.242e+3 (60,18)**	2.6e+2 (5.8e+1)	3.3e+2 (2.8e+1)	2.30e+2 (5.08)
F22	**2.28e+3 (59,51)**	1.0e+2 (6.6e−1)	1.9e+3 (1.9e+3)	3.12e+3 (1.52e+3)
F23	**2.62e+3 (9,85)**	3.7e+2 (1.3e+1)	6.5e+2 (1.0e+2)	4.56e+2 (8.72)
F24	**2.75e+3 (8,33)**	4.4e+2 (1.1e+1)	6.9e+2 (6.7e+1)	5.30e+2 (7.42)
F25	2.91e+3 (23,90)	**3.9e+2 (8.9e+0)**	3.9e+2 (4.0e+0)	5.05e+2 (3.61e+1)
F26	**2.91e+3 (22,45)**	8.6e+2 (6.4e+2)	2.0e+3 (1.6e+3)	1.38e+3 (9.75e+1)
F27	**3.09e+3 (2,75)**	5.3e+2 (1.1e+1)	5.7e+2 (5.0e+1)	5.46e+2 (2.71e+1)
F28	3.16e+3 (71,04)	**3.3e+2 (5.4e+1)**	4.2e+2 (4.3e+1)	4.73e+2 (2.38e+1)
F29	**3.19e+3 (78,61)**	5.6e+2 (1.2e+2)	9.3e+2 (2.2e+2)	4.84e+2 (1.02e+2)
F30	4.24e+5 (5.42e+5)	**3.6e+3 (7.8e+2)**	5.1e+3 (3.1e+3)	6.81e+5 (8.48e+4)
	19/30	**6/30**	**0/30**	**5/30**

*The optimum value for the benchmark problems is reported on Table 1. The results in [1,3,4,5] did not include the optimum value for their results. However, when comparing with our results that objective function values are reported, we consider that difference.

Table 2 shows the comparisons of the CEC 2017 benchmark problems. In the solutions of BIA, PSO, and SHADE the results are only the objective values subtracted from the optimal values. Therefore, values in Table 2 are not consistence. For this reason, when compared the results the optimum values are summed with the optimum value given in Table 1. As given in Table 2, the SSA gives the bet result statistically among other algorithms. The success rates for SSA, BIA, PSO and SHADE are 63.3%, 20%, 0%, and 16%, respectively. Even one of frequently preferred nature inspired optimization algorithm PSO, gives the worst performance since it couldn't give a best result even for a single function. But for problem id 25, BIA and PSO gives approximately same results. The unexpected result got from the first function unimodal Shifted and Rotated Bent Cigar Function; even gives better than PSO and BIA algorithms. Also, the SHADE algorithm has a better performance for the unimodal functions. Without rotation and shifting of the unimodal function; it is expected that all algorithms perform well.

V. CONCLUSION

In this research, the SSA algorithm is evaluated on CEC 2017 benchmark problems. The aim and the motivation of this research is to clearly and fair demonstration of the SSA algorithm under relatively harder challenging problem set. Even SSA is evaluated for engineering problems in the literature; still it is a missing discussion to evaluate the performance of SSA on the common problem set. For this reason, SSA is implemented on CEC 2017 benchmark problem set and compared with BIA, PSO and SHADE algorithms. The results support that SSA has a good performance on the problem set when compared other algorithms. However still there are some problems related to SSA that must be improved because the local optimum problem remains on this algorithm. Therefore, as future study SSA will be improved with hybrid usage with other optimization algorithm and improvement on the update rules.

REFERENCES

1. Kazikova A, Pluhacek M, Senkerik R (2019) “Performance of the Bison algorithm on benchmark” IEEE CEC 2017. In: Silhavy R (ed) Artificial intelligence and algorithms in intelligent systems., Advances in intelligent systems and computing, vol 764. Springer.
2. Mirjalili S, Gandomi AH, Mirjalili SZ, Saremi S, Faris H, Mirjalili SM (2017) Salp swarm algorithm: A bio-inspired optimizer for engineering design problems. Adv Eng Softw 114:163–191.
3. Bagheri Tolabi, H., Lashkar Ara, A. & Hosseini, R. (2021) An enhanced particle swarm optimization algorithm to solve probabilistic load flow problem in a micro-grid. Appl. Intell. 51, 1645–1668. https://doi.org/10.1007/s10489-020-01872-4
4. R. Salgotra, S. Mirjalili and A. H. Gandomi, (2022) "Enhancing Differential Evolution Algorithm: Adaptation for CEC 2017 and CEC 2021 Test Suites," 2022 9th International Conference on Soft Computing & Machine Intelligence (ISCMI), Toronto, ON, Canada, 2022, pp. 235-240, 10.1109/ISCMI56532.2022.10068469.
5. R. Salgotra, U. Singh, S. Saha and A. H. Gandomi, (2020) "Improving Cuckoo Search: Incorporating Changes for CEC 2017 and CEC 2020 Benchmark Problems," 2020 IEEE Congress on Evolutionary Computation (CEC), Glasgow, UK, 2020, pp. 1-7, doi: 10.1109/CEC48606.2020.9185684.
6. Rustagi, K., Bhatnagar, P., Mathur, R. et al. (2024) “Hybrid salp swarm and grey wolf optimizer algorithm-based ensemble approach for breast cancer diagnosis”. Multimed Tools Appl https://doi.org/10.1007/s11042-023-18015-9
7. Ibrahim AL-Wesabi, Fang Zhijian, Hassan M. Hussein Farh, Idriss Dagal, Abdullrahman A. Al-Shamma'a, Abdullah M. Al-Shaalan, Yang kai, (2024) “Hybrid SSA-PSO based intelligent direct sliding-mode control for extracting maximum photovoltaic output power and regulating the DC-bus voltage,” International Journal of Hydrogen Energy, Vol. 51, Part C, 348-370,
8. Ijaz Ahmed, Muhammad Rehan, Abdul Basit, Saddam Hussain Malik, Waqas Ahmed, Keum-Shik Hong, (2024) “Adaptive salp swarm algorithm for sustainable economic and environmental dispatch under renewable energy sources,” Renewable Energy, Volume 223.

9. Hongbo Zhang, Xiwen Qin, Xueliang Gao, Siqi Zhang, Yunsheng Tian, Wei Zhang, (2024) "Improved salp swarm algorithm based on Newton interpolation and cosine opposition-based learning for feature selection," Mathematics and Computers in Simulation, Vol. 219, 544-558.
10. Hichri, A., Hajji, M., Mansouri, M. et al. (2024) "Supervised machine learning-based salp swarm algorithm for fault diagnosis of photovoltaic systems", J. Eng. Appl. Sci. 71, 12. https://doi.org/10.1186/s44147-023-00344-z.
11. N.H. Awad, M.Z. Ali, J.J. Liang, B.Y. Qu, P. N. Suganthan, (2016) "Problem Definitions and Evaluation Criteria for the CEC 2017 Special Session and Competition on Single Objective Real-Parameter Numerical Optimization," Nanyang Technological University, Jordan University of Science and Technology and Zhengzhou University, Tech. Rep..
12. Kennedy, J., Eberhart, R. (1995). "Particle Swarm Optimization" Proceedings of IEEE International Conference on Neural Networks. Vol. IV. pp. 1942–1948. doi:10.1109/ICNN.1995.488968.
13. Tanabe, R., Fukunaga, A. "Success-history based parameter adaptation for Differential Evolution", IEEE Congress on Evolutionary Computation (CEC 2013), Cancún, México, 20–23 June 2013, pp. 71–78. 2013.