PMT-SSA : Parallel Mirror Technique Based Salp Swarm Optimization Algorithm For Engineering Applications

Sudhir Sharma
HOD Electrical Engg.
DAVIET, Jalandhar, India

Shivani Mehta
Associate Prof. Electrical Engg.
DAVIET, Jalandhar, India

Nitesh Kumar Joshi
Dept. of Electrical Engg.
DAVIET, Jalandhar, India

Abstract—*This paper proposes improved version of existing salp swarm algorithm (SSA) using recently developed opposition-based learning method called parallel mirrors technique (PMT). The improved version of SSA i.e. PMT-SSA is tested on 7 unimodal and 6 multimodal mathematical benchmark functions. The developed algorithm is further applied to solve economic load dispatch problem in power system. The results compared with SSA algorithm verify the superiority of PMT-SSA in solving optimization problems.*

Keywords—**SSA, PMT-SSA, Metaheuristic, optimization**

I. INTRODUCTION

Metaheuristics are gradually becoming popular for handling difficult optimization problems due to their flexibility, simplicity,derivative-free method, and local-optima avoidance. SSA is the newly introduced swarm based metaheuristic method that imitate the characteristics of sea tunicates also called as salps [1]. To explore the entire search space, SSA generally forms a salp chain in oceans. Mirjalili et al. [1] analyzed the efficiency of SSA on standard benchmark functions and found that the salp swarm optimization algorithm outperform the other metaheuristic methods i.e.cuckoo search [2], BA (Bat algorithm) [3], grey wolf optimization algorithm (GWO) [4].

In SSA, position all the individuals (follower salps) are updated in respect of each other in such a way that they travel gradually in the direction of leading salp and SSA uses parameter C1 to control the exploration and exploitation capabilities. However, SSA sometimes stick to some local solution due to hasten exploitation at later iterations. Hence, the convergence is not ensured in some instances. Hence, In this paper SSA has been modified using new opposition based learning method called parallel mirror technique (PMT) as presented in[5].

The rest of the paper is structured as follows: Sections 2 and 3 introduce the SSA and the PMT opposition-based learning method, respectively. PMT-SSA algorithm is presented in Section 4. The performance of the PMT-SSA is assessed on benchmark functions in Section 5 and finally, the results are concluded in Section 6.

II. SALP SWARM ALGORITHM

The salp chains are mathematically modelled by dividing the the population in two groups: leader and followers. The salp at the front of the chain is called Leader, while the rest of salps are well-thought-out as followers. As the term "salp" suggests, the leader directors swarm and the followers follow each other (and leader directly or indirectly). Correspondingly, to different swarm-based techniques, the placement of salps is defined in an n - dimensional seek space wherein n is the count of variables of a given problem. Therefore, the position of all salps are stored in a two-dimensional matrix known as x. It is also assumed that there may be a food supply known as F within the seek area because the swarm's target. To update the position of the leader the following equation is proposed:

$$x_j^1 = \begin{cases} F_j + c_1((ub_j - lb_j)c_2 + lb_j] & c_3 \geq 0 \\ F_j - c_1((ub_j - lb_j)c_2 + lb_j] & c_3 < 0 \end{cases} \quad \ldots (1)$$

Where x_j^1 shows the position of the first salp (leader) in the j th dimension, F_j is the position of the food source in the j^{th} dimension, ub_j indicates the upper bound of j^{th} dimension, lb_j indicates the lower bound of j^{th} dimension, c_1, c_2, and c_3 are random numbers.

Eq. (1) shows that the leader only updates its position with respect to the food source. The coefficient c_1 is the most important parameter in SSA because it balances exploration and exploitation defined as follows:

$$C_1 = 2e^{-\left(\frac{4l}{L}\right)^2} \quad \ldots(2)$$

where "l" is the current iteration and "L" is the maximum number of iterations. The parameter c_2 and c_3 are 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 in- finity as well as the step size. To update the position of the followers, the following equations is utilized (Newton's law of motion):

$$x_j^i = \frac{1}{2} a t^2 + v_0 t \quad (.3)$$

were i ≥2, x_j^i shows the position of i th follower salp in j th dimension, t is time, v_0 is the initial speed, and a = v_{final}/v_0 where v = $(x - x_0)/t$.

Because the time in optimization is iteration, the discrepancy between iterations is equal to 1, and considering v 0 = 0, this equation can be expressed as follows:

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

where i ≥2 and x_j^i shows the position of i^{th} follower salp in j^{th} dimension. With Eqs. (1) and (4), the salp chains can be simulated. Flowchart of SSA is given below.

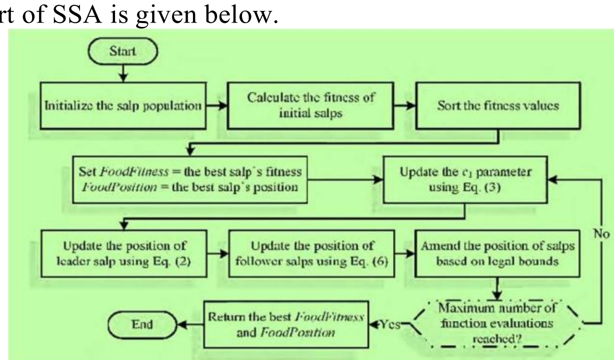

Fig.1. SSA flowchart

III. PARALLEL MIRROR TECHNIQUE

Parallel mirror technique is a novel opposition based learning approach for improving the performance of existing metaheuristics[5]. In PMT, a candidate solution is positioned between two parallel mirrors. Afterward the creation of first image, new images are created continuously into the opposite mirror by previous image, which causes an infinite number of similar images in the virtual space. These images are new candidate solutions. Due to distribution of candidates in objective space, the probability of reaching the global best solution is increased and local optima is avoided. In contrast to traditional opposition-based learning approach, the PMT uses broader ideas to explore in more than one opposite direction and create more than one new candidate solution.

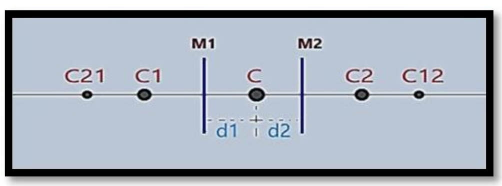

Fig.2 Example of parallel mirror.

The parallel mirrors illustration in Fig.2 shows candidate solution c is bounded by two mirrors (M1, M2) and the distance between each mirror and c equals (d1, d2). Primarily, the first image c1 is produced by mirror M1 from the original value c, at the same time another image c2 from the original value c is produced by M2. Thus, each image of the new images (c1, c2) will produce more new images into the opposite mirrors (c12, c21), respectively. Then each value of these new values (c12, c21) will yield a countless number of images into these two mirrors.

The c is bounded between upper and lower limits of solution space. The location of the mirrors M1 and M2 are defined as follows:

$$\text{Position}(M1) = c - d1; \quad d1 > 0 \quad (6)$$

$$\text{Position}(M2) = c + d2; \quad d2 > 0 \quad (7)$$

Let us assume the current candidate value is c_0. Then the next generated image c_1, c_2, … ci is defined as:

$$c_i = c_{i-1} \pm 2[i * d_1 + (i-1) d_2] \quad (8)$$

```
Input: the population X = {X1, X2... Xn}
Output: Xbest and the updated population X ' = {X1', X2' ... Xn'}
Begin
  Set MI the maximum number of generated images
  Set MF the maximum failure images
  Set initial Sbest = f(X)
  Set initial Xbest = X0
  Set initial mirror d1 = random()
  Set initial mirror d2 = random()
  For i = 0; i < MI; i ++
    Xi = generateImage(X)
    If f(Xi)< f(Xbest) then
        Sbest = f(Xi)
        Xbest = Xi
    Else
        Update d1 = random()
        Update d2 = random()
        MF = MF - 1
    EndIf
    If MF == 0
      Break; //Stop the for loop
    EndIf
  EndFor
  Return the best population, X and the best result (Xbest)
End
```

Fig 3 Pseudo code for PMT[15].

Two stopping factors are used in PMT to limit the number of images generated. These are the maximum number of images (MI) and the maximum number of failure (MF) images. The MI is the maximum number of images that can be generated for each candidate and the MF is the maximum number of failure-generated images (i.e., failed to reach a better solution than the current candidate solution).The procedure of applying PMT is shown as pseudocode in Fig.3.

IV. PMT-SSA

Like many meta-heuristic algorithms, SSA suffers from a low convergence rate and local optima stagnation. Hence, engaging the PMT to improve the SSA should give a chance for the SSA to overcome some of its shortcomings. The modified PMT-SSA algorithm pseudocode is shown in fig.3.

V. RESULTS & DISCUSSION

A. PMT-SSA testing on benchmark functions

The PMT-SSA algorithm is tested on 7 unimodal and 6 multi-modal mathematical benchmark functions adopted from [1]. The algorithm was run for 30 times with maximum number of iterations set at 200 and population size of 30. For PMT-SSA, MI and MF are set at 6 and 3 respectively.

As per the results in Table 1, PMT-SSA delivers very viable results. This algorithm outperforms SSA in F1, F2,F4,F5,F6 and F7 in terms of best result. As the appropriateness of unimodal functions for benchmarking exploitation, these outcomes validate the advantage of PMT-SSA over SSA for exploiting the optimal results.

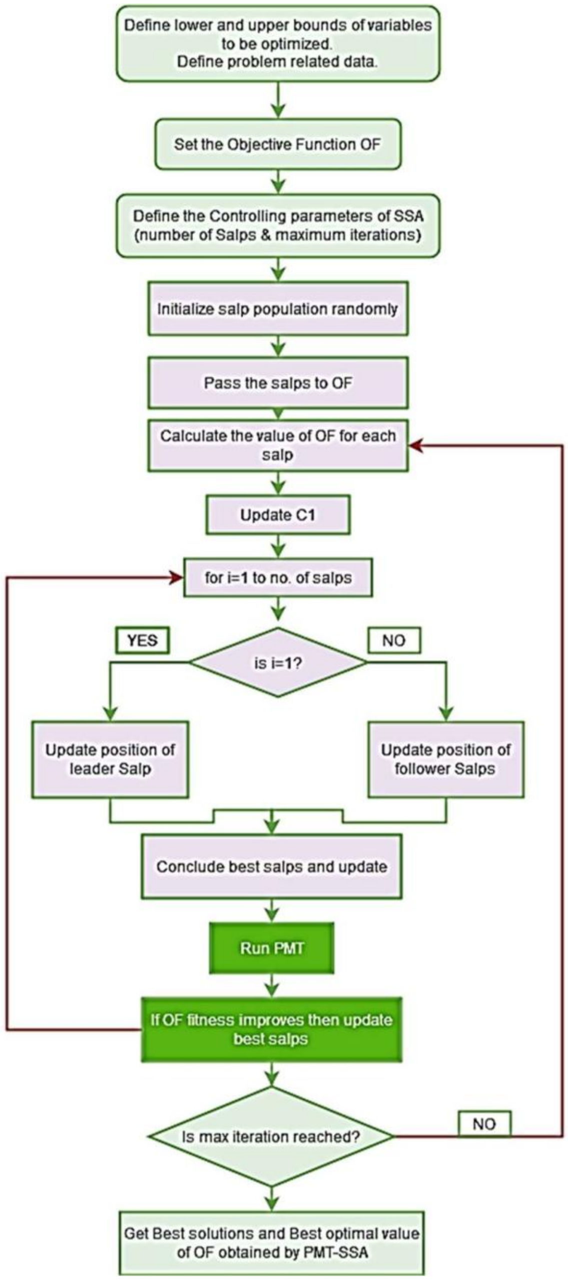

Fig 4 PMT SSA flowchart

In contrast to unimodal functions, multimodal functions consist of multiple local optima which makes them suitable for testing the exploration competence of an algorithm. As per the results in Table 1, PMT-SSA performs well on multimodal functions too. These outcomes establish the fineness of the PMT-SSA algorithm in terms of exploration.

B. PMT-SSA testing on economic load dispatch(ELD) problem

The objective function of the ELD problem is to minimalize the total generation cost while satisfying the different constraints, when the required load of power system is being supplied [6]. The objective function to be minimized is given by the following equation:

$$F(P_g) = \sum_{i=1}^{n} (a_i P_{gi}^2 + b_i P_{gi} + c_i) + |d_i \sin(e_i (P_{gi}^{\min} - P_{gi}))| \quad \ldots(9)$$

TABLE I. RESULTS ON UNIMODAL & MULTIMODAL BENCHMARK FUNCTIONS

Function	Algorithm Indices	Best	Mean	Worst	SD
F1	SSA	3.98E-08	2.54E-07	1.09E-06	2.74E-07
	PMT-SSA	6.3E-22	1.48E-14	1.41E-13	3.42E-14
F2	SSA	5.51E-06	0.006136	0.100815	0.021068
	PMT-SSA	1.28E-10	9.81E-09	3.4E-08	8.83E-09
F3	SSA	3.11E-09	4.06E-07	1.12E-05	2.04E-06
	PMT-SSA	3.92E-18	6.54E-15	6.06E-14	1.22E-14
F4	SSA	1.27E-05	2.2E-05	6.91E-05	1.06E-05
	PMT-SSA	1.36E-10	3.39E-08	2.55E-07	5.26E-08
F5	SSA	4.232761	52.48037	258.0132	79.73837
	PMT-SSA	7.341271	7.782304	8.204915	0.20053
F6	SSA	2.2E-10	9.72E-10	1.77E-09	3.39E-10
	PMT-SSA	5.66E-10	1.17E-09	2.15E-09	4.36E-10
F7	SSA	0.002524	0.015504	0.041252	0.010475
	PMT-SSA	6.03E-06	0.000129	0.000352	0.000106
F8	SSA	-3321.27	-2705.16	-2227.8	305.4258
	PMT-SSA	-4066.26	-3375.49	-2501.56	326.9185
F9	SSA	4.974795	18.04189	50.74273	9.27382
	PMT-SSA	0	4.74E-16	1.42E-14	2.59E-15
F10	SSA	7.35E-06	0.496231	2.579928	0.819514
	PMT-SSA	6.69E-11	1.46E-08	5.41E-08	1.49E-08
F11	SSA	0.041881	0.269808	0.68609	0.170643
	PMT-SSA	0	8.67E-15	6.48E-14	1.51E-14
F12	SSA	1.35E-11	0.758591	4.044616	1.036615
	PMT-SSA	6.75E-12	4.59E-11	1.54E-10	3.12E-11
F13	SSA	6.5E-11	0.003965	0.021024	0.006585
	PMT-SSA	2.22E-11	0.001465	0.010987	0.003799

For testing we adopted 13 generator system from [8] and power demand of 1800 MW was adopted to run dispatch.

Generator	SSA	PMT-SSA
Pg1	628.3185	628.1327
Pg2	74.79987	299.1400

Pg3	360	223.7534
Pg4	60.00002	60.0000
Pg5	60	109.3692
Pg6	159.7331	60.0000
Pg7	109.8665	60.0000
Pg8	60	109.4924
Pg9	60.00002	60.0000
Pg10	40	40.0198
Pg11	40.00363	40.0188
Pg12	92.23303	55.0036
Pg13	55	55.0698
Total power generation	1800	1800
Minimum Cost(Rs)	18101.15496	17974.7393

Method	Minimum Cost(Rs)	Mean Cost(Rs)	Maximum Cost(Rs)
CEP[8]	18048.21	18190.32	18404.04
FEP[8]	18018.00	18200.79	18453.82
MFEP[8]	18028.09	18192.00	18416.89
IFEP[8]	17994.07	18127.06	18267.42
PSO[9]	18030.72	18205.78	--
SSA	18101	18205.23	18632
PMT-SSA	17974.73	18195.49	18491

VI. CONCLUSION

In this paper a modified version of existing SSA algorithm is presented i.e. PMT-SSA. The proposed algorithm utilizes PMT opposition-based learning strategy to improve the performance of original SSA algorithm. The developed algorithm was tested on different unimodal and multimodal benchmark functions to test its performance and compare it with SSA. The PMT-SSA was further applied to solve economic load dispatch problem. The results comparisons show that PMT-SSA outperforms SSA algorithm.

REFERENCES

1. S. Mirjalili, A. H. Gandomi, S. Z. Mirjalili, S. Saremi, H. Faris, and S. M. Mirjalili, “Salp swarm algorithm: A bio-inspired optimizer for engineering design problems,” Advances in Engineering Software, vol. 114, pp. 163–191, 2017
2. Yang X-S , Deb S . Cuckoo search via Lévy flights. In: Nature & biologically in- spired computing, 2009. NaBIC 2009. world congress on; 2009. p. 210–14 .
3. Yang X-S . A new metaheuristic bat-inspired algorithm. In: Nature inspired co- operative strategies for optimization (NICSO 2010). Springer; 2010. p. 65–74 .
4. S. Mirjalili, S. M. Mirjalili, and A. Lewis, “Grey Wolf Optimizer,” Adv. Eng. Softw., vol. 69, pp. 46–61, 2014, doi: https://doi.org/10.1016/j.advengsoft.2013.12.007.
5. H. S. Alamri and K. Z. Zamli, “PMT: Opposition-Based Learning Technique for Enhancing Meta-Heuristic Performance,” IEEE Access, vol. 7, pp. 9765397672, 2019, doi: 10.1109/ACCESS.2019.2925088.
6. Sharma, Sudhir, Shivani Mehta, and Nitish Chopra. "Economic load dispatch using grey wolf optimization." International Journal of Engineering Research and Applications 5, no. 4 (2015): 128-132.
7. Sharma, Sudhir, Shivani Mehta, and Nitish Chopra. "Grey wolf optimization for solving non-convex economic load dispatch." Engineering Research 3, no. 3 (2015): 17-21.
8. Nidul Sinha, R. Chakrabarti, and P. K. Chattopadhyay. Evolutionary programming techniques for economic load dispatch. IEEE Transactions on Evolution Computation. 2003; 7: 83-94.
9. T. Aruldoss Albert Victoire and A. Ebenezer Jeyakumar. Hybrid PSO-SQP for economic dispatch with valve-point effect. Electric Power Systems Research. 2004; 71: 51-9.