Journal of ELECTRICAL ENGINEERING, VOL 73(2022), NO5, 325–331

sciendo

PAPERS

Sidelobe level suppression for elliptical antenna arrays using modified SALP swarm algorithm

Erhan Kurt[1*], **Suad Basbug**[2], **Kerim Guney**[1]

In this study, a modified version of salp swarm algorithm (MSSA) is used to synthesize elliptical antenna arrays (EAAs). The original salp swarm algorithm (SSA) is an optimization algorithm inspired by the behavior of salps in nature, which is used to solve engineering problems. The main purpose of the synthesis in this study is to obtain an EAA pattern with low maximum sidelobe levels (MSLs) for a fixed narrow first null beamwidth (FNBW). For different examples, the amplitude and angular position values of the antenna array elements are considered as optimization parameters. To show the effectiveness of the MSSA, eight examples of EAAs with 8, 12, and 20 elements are given. The results obtained with MSSA are compared with those of the antlion optimization, symbiotic organizations search, flower pollination algorithm, and accelerated particle swarm optimization from the literature. It is clear from the numerical results that MSSA outperforms the other algorithms in terms of the suppression of MSL.

K e y w o r d s: elliptical antenna array, salp swarm algorithm, sidelobe level suppression

1 Introduction

Antenna arrays are frequently used to improve the performance of many radar, wireless, satellite, and mobile communication systems [1]. The use of antenna arrays can provide a better signal quality, higher directivity, and increasing pattern efficiency. In many communication systems, the low sidelobe levels are required at a fixed main beam width in the radiation pattern. It is possible to obtain the desired radiation diagrams by controlling the amplitude, phase, and position values of the array elements separately or together.

Antenna arrays are generally classified as linear, circular, elliptical, or conformal antenna arrays according to their geometric structure. The linear antenna array with its one-dimensional geometry is the simplest array structure. Circular, elliptical, and planar arrays are two-dimensional antenna arrays. Three-dimensional antenna structures such as conformal, spherical, and cylindrical forms are relatively rare antenna arrays.

In this study, elliptical antenna arrays (EAAs) with different number of elements are chosen for the examples of the radiation pattern synthesis. The EAAs have some characteristic features in terms of antenna parameters. The direction of the main beam can be oriented in the desired direction, as in the circular antenna arrays. Furthermore, the structure of EAAs does not contain edge elements, thus they are less sensitive to the mutual coupling. However, EAA synthesis problems are more complex than linear and circular antenna array synthesis problems. Several methods were used in the synthesis of EAAs in the literature [2-12].

In this work, a salp swarm algorithm (SSA) with a modification is used to optimize the amplitude and angular position values of the EAA elements to achieve radiation patterns with low maximum sidelobe level (MSL) at fixed first null beamwidth (FNBW). Mirjalili *et al* . proposed SSA in 2017 as a population-based metaheuristic swarm optimization algorithm [13]. SSA is inspired by the behavior of the salp swarm that navigates and looks for food in the ocean. In [13], SSA algorithm is explained in a detailed manner and compared with a collection of state-of-the-art algorithms including particle swarm optimization, fire fly algorithm, gravitational search algorithm, and bat algorithm. In the literature, SSA is employed to solve several engineering problems having different characteristics such as electric power dispatch problem [14], design of photovoltaic cell models [15], wireless sensor networks [16], variable speed wind generators [17], and conformal antenna array design [18] owing to its simplicity and high performance.

In this paper, a modified version of SSA (MSSA) is employed to obtain the desired radiation patterns. The modification in SSA is about determining the new positions for the population members considering the whole iterative history of the member instead of the only previous and current ones. The enhancement that achieved by the modification is shown with comparative convergence curves in the first and second example. In this paper, eight different EAA optimization examples with different number of array elements are considered. In the first six examples, the amplitude values of EAA are calculated by MSSA to achieve low MSLs at fixed FNBWs. The last two example includes the optimization of the angular el-

[1] Department of Electrical and Electronics Engineering, Nuh Naci Yazgan University, Kayseri, 38170, Turkey, * corresponding author: ekurt@nny.edu.tr, [2] Department of Electrical and Electronics Engineering, Nevsehir Haci Bektas Veli University, Nevsehir, 50300, Turkey

https://doi.org/10.2478/jee-2022-0043, Print (till 2015) ISSN 1335-3632, On-line ISSN 1339-309X

ement positions in the EAA to reduce sidelobe levels at fixed FNBW. Furthermore, comparisons are presented to demonstrate the reliability of the given examples and the performance of the proposed algorithm. For this purpose, the results obtained by using MSSA are compared with those of antlion optimization (ALO) [10], symbiotic organizations search (SOS) [10], flower pollination algorithm (FPA) [11], and accelerated particle swarm optimization (APSO) [11]. The results of the examples in [10] and [11] are specifically selected for the comparison in this paper. Because the array factor of the EAA is corrected by Dib *et al* in [10] and this corrected version of the array factor is used for the EAA synthesis examples in this paper. Therefore, the comparisons are limited within the results of the examples in [10] and [11].

We briefly explain the main principles of array factor of EAA and the cost function for the optimization task and introduce the algorithmic details of MSSA. Finally, the numerical results are given.

2 Problem formulation

Figure 1 shows the EAA geometry with N isotropic elements placed in the shape of an ellipse on the $x-y$ plane ($\theta = 90$). In this geometry, θ and ϕ are the elevation and azimuth angles of the observation point at which the array factor is to be calculated, respectively. The array factor(AF) of N elements EAA can be mathematically described as follows [10]

$$F_{\mathrm{a}}(\theta,\phi) = \sum_{n=1}^{N} I_n e^{j\alpha_n} e^{jk\rho_n \sin\theta\cos(\phi-\phi_{\mathrm{n}})}, \tag{1}$$

where I_n is the excitation amplitude and α_n is the excitation phase of the n_{th} antenna element.

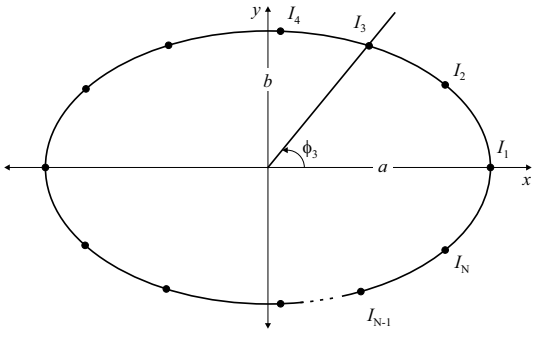

Fig. 1. Geometry of an elliptical antenna array

The array factor equation (1) used in this paper is a corrected version of EAA factor stated in [10]. The detailed information about the formulation correction can be found in [10]. While the main beam is directed to (θ_0, ϕ_0), the corresponding excitation phase is given as

$$\alpha_n = -k\rho_n \sin(\theta_0)\cos(\theta_0 - \phi_0), \tag{2}$$

where k is the wave number ($k = 2\pi/\lambda$) and ρ_n refers to the radial distance of the n_{th} element which is measured from the origin. ρ_n can be calculated as follows

$$\rho_n = \frac{ab}{\sqrt{(b\cos\phi_n)^2 + (a\sin\phi_n)^2}}. \tag{3}$$

Here, a and b are the lengths of the semi-major axis and the semi-minor axis of the ellipse, respectively. Φ_N is the angular position of the n-th element and can be given as

$$\phi_n = \frac{2\pi(n-1)}{N}. \tag{4}$$

To simplify the calculations, θ_0 is taken as 90°and ϕ_0 is taken as 0°. Finally, one of the parameters of the ellipse is ellipse eccentricity. It is fixed ($e = 0.5$) and can be defined as

$$e = \sqrt{1 - \frac{b^2}{a^2}}. \tag{5}$$

In this work, the values of the semi-major axis a are selected as 0.5λ, 1.15λ and 1.6λ for the number of 8, 12, and 20 elements EAA, respectively, as used in [10, 11]. During the optimization process, MSSA is employed to minimize the MSL values (m) while fixing FNBW values (w). For the first six examples, the optimization parameters are the only amplitude values whereas the angular position values are optimized in the last example. Thus, for the optimization in this study, the cost function is defined as follows

$$F = c_{\mathrm{m}} F_{\mathrm{m}} + c_{\mathrm{w}} F_{\mathrm{w}}, \tag{6}$$

where c_{m} and c_{w} are the weight factors, and F_{m} and F_{w} are the functions used to minimize the MSL value and to fix the FNBW value (w), respectively. The first of them

$$F_{\mathrm{m}} = \int_{-\pi}^{\phi_{\mathrm{n1}}} \delta_{\mathrm{m}}(\phi)\,\mathrm{d}\phi + \int_{\phi_{\mathrm{n2}}}^{\pi} \delta_{\mathrm{m}}(\phi)\,\mathrm{d}\phi, \tag{7}$$

where ϕ_{n1} and ϕ_{n2} are the two angles of the first nulls on the left and right sides of the main beam, and

$$\delta_{\mathrm{m}}(\phi) = \begin{cases} F_{\mathrm{a}}(\phi) - m_{\mathrm{d}}, \text{ if } F_{\mathrm{a}}(\phi) > m_{\mathrm{d}} \\ 0, \text{ elsewhere,} \end{cases} \tag{8}$$

where F_{a} is the array factor (in dB) and m_{d} is desired MSL value. The second, FNBW function can be expressed as

$$F_{\mathrm{w}}(\phi) = \begin{cases} w_0 - w_{\max}, \text{ if } w_0 > w_{\max}, \\ 0, \text{ elsewhere,} \end{cases} \tag{9}$$

where w_0 is the FNBW value obtained by MSSA and $w_{\max}$ is its desired maximum value.

3 Methods

3.1 Salp swarm algorithm

SSA is a metaheuristic optimization algorithm inspired by the swarming and navigation behaviors of salps in oceans for food [13]. These sea creatures, which have a transparent bottle-shaped body, belong to the family of Salpidae. Their tissues are very similar to jellyfishes. In nature, salp individuals hold each other and live as a swarm that looks like a chain. The swarm structure shown in Fig. 2 is called salp chain. The salp chain consists of a leader and followers. The leader is the individual positioned at the front of the salp chain.

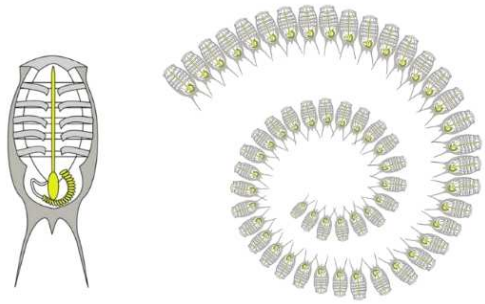

Fig. 2. salp structure of swarm: (a) – single, (b) – chain

The collective behavior of salp group forms the basis of the SSA algorithm. The population individuals are randomly distributed in the multi-dimensional search space. During the optimization process, followers follow the leader while navigating the search space to find the best solution [13]. The position of the leader salp is updated by using the following equation

$$x_j^1 = \begin{cases} F_j + c_1 \left[(ub_j - lb_j)c_2 + lb_j \right], & \text{if } c_3 \geq 0 \\ F_j - c_1 \left[(ub_j - lb_j)c_2 + lb_j \right], & \text{if } c_3 < 0, \end{cases} \tag{10}$$

where j is the number of the dimension. F_j, ub_j and lb_j are the position of the food source in the jth dimension, the upper and lower bounds of jth dimension, respectively. The factors c_1, c_2, and c_3 are random numbers [13]. Equation (10) shows that the leader only updates its position with respect to the food source. During this update process, c_1 balances exploration and exploitation which is de?ned as follows

$$c_1 = 2e^{-\left(\frac{4l}{L}\right)^2}, \tag{11}$$

where l and L denote the current iteration and maximum number of iterations, respectively [13]. In basic SSA algorithm, the positions of the followers can be calculated by using the following formula

$$x_j^i = \frac{1}{2}(x_j^i + x_j^{i-1}) \quad \text{for } i \geq 2, \tag{12}$$

where x_j^i shows the current position of ith follower in jth dimension. In this way, the overall behavior of salp chain was modeled by means of (10) and (12). The position values of the population members of SSA algorithm in the solution space are the candidate solutions. The algorithm tries to find the most optimum values for all dimensions by considering the given constraints. In this study, there are two kinds of parameters which we need to optimize. The first type of parameters are amplitude values of the array elements. In the examples related with the amplitude-controlled optimization, the positions of the salp members in the solution space correspond to the amplitude values of the array elements (I_n). Similarly, for the second type of parameter optimization in this study, the angular position values (ρ_n) are the population member positions.

3.2 Modified salp swarm algorithm

The original SSA is well performed in the optimization problems. However, as stated in the no free lunch theorem [19], different problems need different algorithms to solve. Besides, it is also seen in the literature that, even minor changes in the code may significantly improve the performance of the algorithm. Due to the simple and easy-to-understand structure of the SSA, it is possible to make changes while applying on different optimization problems. In our case, equation (12) is modified before applying SSA to array synthesis as follows

$$x_j^i = \frac{1}{M} \sum_{m=1}^{M} (x_j^m), \tag{13}$$

where M is the number of follower slaps. Thus, while calculating the position of the next follower, the position of all previous followers up to that point is considered to improve the convergence rate.

4 Numerical results

In this section, the performance of MSSA on EAA synthesis examples is evaluated in detail. To validate the capability and flexibility of the proposed MSSA, eight examples of EAAs with $N = 8, 12,$ and 20 elements have been simulated, see Tab. 2. In this article, MSSA is used to find optimum amplitude and position values of EAA e)lements to obtain low MSL with constraint FNBW. In the first six examples (EX1 to EX6), an equally spaced EAA with 8, 12, and 20 elements is used for sidelobe suppression by controlling only the element amplitudes. In the seventh and eighth example (EX7, EX8), the desired pattern with low MSL is obtained by controlling only the angular positions of the array elements for an unequally spaced EAA with 12 elements.

All simulation results achieved by using MSSA were compared with those of four different algorithms which are ALO [10], SOS [10], FPA [11], and APSO [11]. The results taken from [10] and [11] are calculated by Dib *et al* and Chakravarthy *et al* by means of a corrected version of EAA factor, respectively. In [10], Dib *et al* showed

Table 1. MSL and FNBW values for the uniform EAAs with 8, 12, and 20 elements [10]

N	FNBW	MSL (dB)	$a(\lambda)$	$b(\lambda)$	$e(\lambda)$
8	98.8	-7.76	0.5	0.433	0.5
12	42.2	-2.95	1.15	0.952	0.5
20	30.2	-6.87	1.6	1.385	0.5

that the array factor used in [2-9] has some defects related with calculating the position of array elements along the perimeters of ellipse. In this paper, we also used the corrected version of the EAA factor. For this reason, the results calculated by MSSA are only compared with those of [10] and [11].

The algorithm was run 20 times to observe the actual performance of MSSA and make a fair comparison. The best results calculated by the algorithm were chosen for the evaluation. In the optimization process, the number of population size and the number of iterations is 30 and 1000, respectively. Parameter $c_1, c_2,$ and c_3 are random numbers generated in the interval of [0, 1]. The parameter c1 is updated by using (11) during the iterations. The simulations are performed on a personal computer which has 2.1 GHz i5 processor and 16 GB RAM. The software used for the optimization applications of MSSA is MATLAB.

The MSL and FNBW values of the EAAs having uniform amplitude and angular positions are given with corresponding $a, b,$ and e values in Tab. 1.

In the first example, an 8-elements EAA is considered. During the iteration FNBW is fixed to 102°as a constraint. The radiation pattern produced by MSSA is illustrated in Fig. 3. Based on Fig. 4, one can say that the MSSA has faster convergence than that of the original SSA.

In the second example, the FNBW value was 99°and the number of array elements is the same as in EX1. The array pattern obtained from the MSSA is in Fig. 5. For a better comparison, the patterns obtained by using FPA [11] and APSO [11] are also presented in Fig. 5. The MSSA takes around 72 second to optimize the 8-elements EAA. MSL value of MSSA is better than that of FPA [11] or APSO [11]. From Fig. 6 we see, that the MSSA leads to better convergence and that 600 iterations are needed to find the optimal solutions.

The EAA considered in the third example has 12 equispaced elements. The sidelobe region is taken as $\phi = [-180°, -22°]$ and $\phi = [22°, 180°]$. The best MSL value is achieved after 1000 iterations by using MSSA. The array pattern obtained by MSSA is compared with other patterns in Fig. 7. Execution time, MSL, mean MSL, and SD values are calculated for the 20 independent runs. The MSL and SD values obtained by using MSSA are better than those of ALO [10] and SOS [10] for the same FNBW of 44°.

In the fourth example, the number of array elements is the same as in EX3. The only difference is the sidelobe region taken as $\phi = [-180°, -21°]$ and $\phi = [21°, 180°]$. Figure 8 presents the radiation pattern with low MSL value obtained by MSSA. The figure also includes other radiation patterns. The MSSA takes around 88 second to optimize the 12-elements EAA. MSL, mean MSL and SD values achieved by MSSA, FPA [11], and APSO [11]. The MSL and standard deviation (σ) achieved by MSSA is slightly better than those of the other algorithms.

Table 2. MSL (m), mean MSL $< m >$, standard deviation (σ) values, and average execution time (T) obtained by different techniques and authors, numbers of EAA elements (N), with fixed value of FNBW (w)

	$N = 8$	m (dB)	$< m >$ (dB)	σ (dB)	T (s)
EX1:	MSSA	-14.44	-14.417	0.012	65
$w =$	ALO [10]	-14.35	-14.233	0.093	60
102 °	SOS [10]	-14.28	-14.250	0.012	17
EX2:	MSSA	-13.53	-13.514	0.010	-
$w =$	FPA [11]	-13.49	-13.490	0.005	-
98.8 °	APSO [11]	-13.49	-13.480	0.016	-
	$N = 12$	m (dB)	$< m >$ (dB)	σ (dB)	T (s)
EX3:	MSSA	-8.05	-8.022	0.016	90
$w =$	ALO [10]	-7.90	-7.556	0.166	95
44 °	SOS [10]	-7.70	-7.617	0.031	26
EX4:	MSSA	-6.09	-6.061	0.016	-
$w =$	FPA [11]	-6.05	-5.920	0.048	-
42.2 °	APSO [11]	-6.06	-5.970	0.062	-
	$N = 20$	m (dB)	$< m >$ (dB)	σ (dB)	T (s)
EX5:	MSSA	-12.13	-12.067	0.035	120
$w =$	ALO [10]	-11.42	-11.364	0.058	55
32 °	SOS [10]	-11.48	-10.829	0.224	190
EX6:	MSSA	-8.92	-8.795	0.062	-
$w =$	FPA [11]	-8.59	-8.230	0.206	-
30.2 °	APSO [11]	-8.70	-8.310	0.237	-
	$N = 12$	m (dB)	$< m >$ (dB)	σ (dB)	T (s)
EX7:	MSSA	-10.36	-	-	-
$w =$	ALO [10]	-10.03	-	-	-
44 °	SOS [10]	-9.95	-	-	-
EX8:	MSSA	-7.56	-7.476	0.053	-
$w =$	FPA [11]	-7.33	-7.110	0.087	-
42.2 °	APSO [11]	-7.45	-7.230	0.114	-

Table 3 shows MSL, mean MSL and SD values of the radiation pattern obtained by using MSSA with constant FNBW value in the 20 independent runs. The MSL value of MSSA is better than those of FPA [11] and APSO [11].

In the fifth and sixth examples, the number of array elements is set to 20. For the fifth example, the

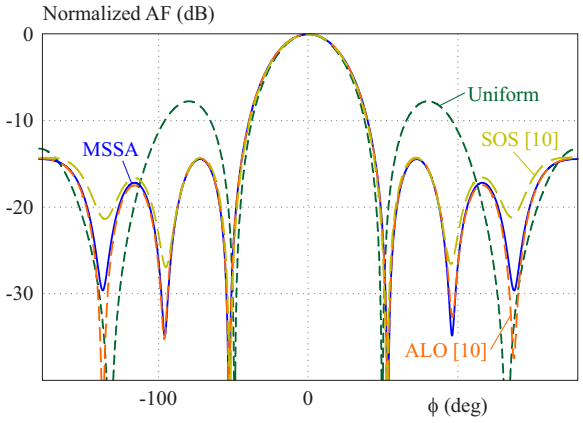

Fig. 3. Radiation patterns obtained by amplitude-only control for 8-elements EAA

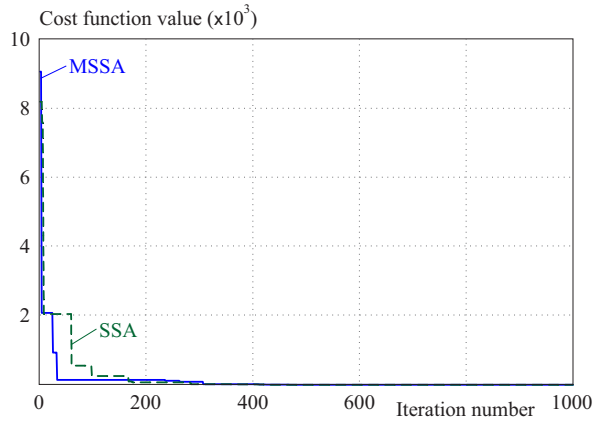

Fig. 4. Convergence curve plots of MSSA and SSA for the first example with 8-element EAA

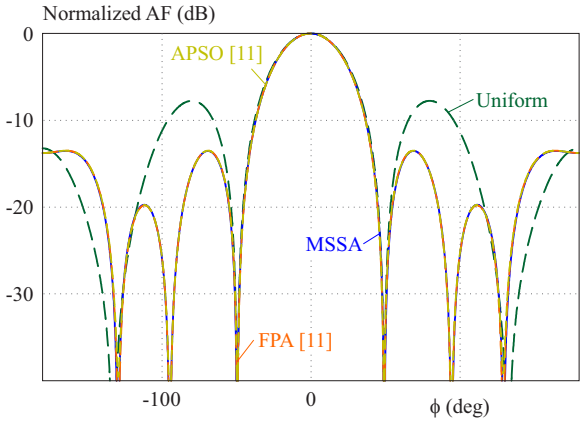

Fig. 5. Radiation patterns obtained by amplitude-only control for 8-elements EAA with fixed FNBW

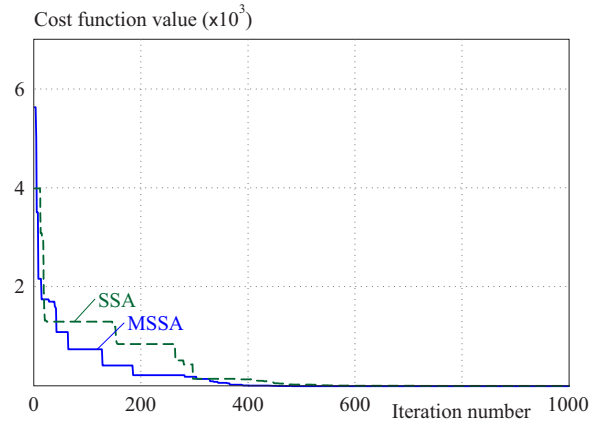

Fig. 6. Convergence curve plots of MSSA and SSA for the second example with 8-element EAA

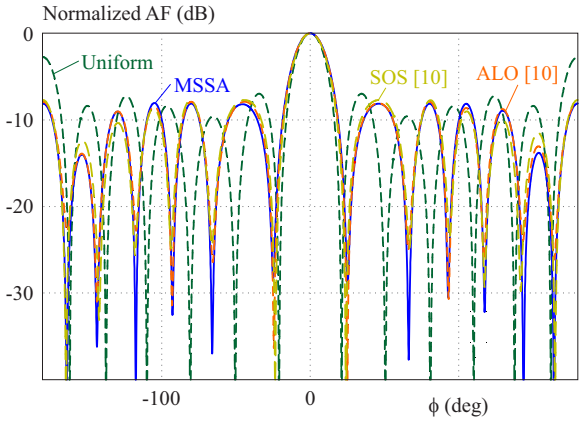

Fig. 7. Radiation patterns obtained by amplitude-only control for 12-elements EAA

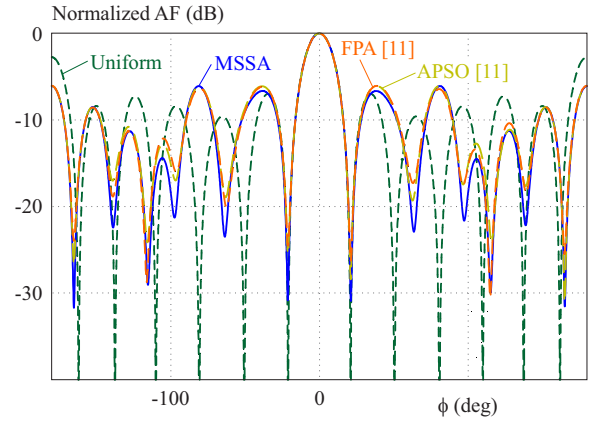

Fig. 8. Radiation patterns obtained by amplitude-only control for 12-elements EAA with fixed FNBW

sidelobe suppression region is set to $\phi = [-180°, -16°]$ and $\phi = [16°, 180°]$. The radiation patterns obtained by MSSA, ALO [10], and SOS [10] are given in Fig. 9. The best MSL value is achieved after 1000 iterations by using MSSA. The algorithm is also run 20 times. From Tab. 2 is apparent that the MSL and SD value obtained by MSSA is better than those of ALO [10] and SOS [10].

In the sixth example, the optimum amplitude values of EAA with 20-elements are calculated by MSSA to obtain low MSL values in the region $\phi = [-180°, -15°]$ and $\phi = [15°, 180°]$. Figure 10 illustrates the patterns achieved by MSSA, FPA [11], and APSO [11]. The MSSA takes around 110 second to optimize the 20-elements. As it can be seene, the MSL and SD values of MSSA is better than those of the other compared algorithms.

To verify the performance of the proposed algorithm, in the seven and eight examples, the angular positions of the array elements for a 12-elements EAA are optimized. In the seventh example, the FNBW value is chosen as the same with the examples in ALO [10] and SOS [10] for

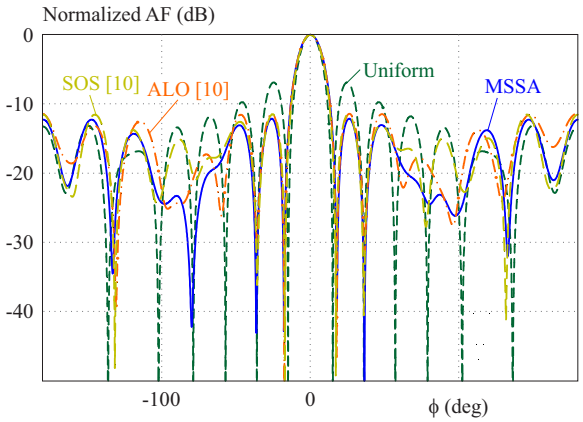

Fig. 9. Radiation patterns obtained by amplitude-only control for 20-elements EAA

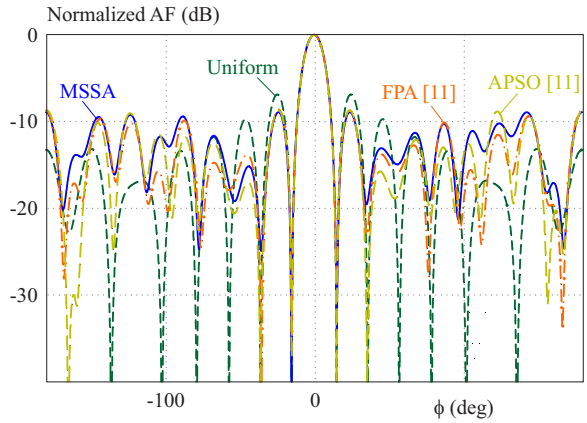

Fig. 10. Radiation patterns obtained by amplitude-only control for 20-elements EAA with fixed FNBW

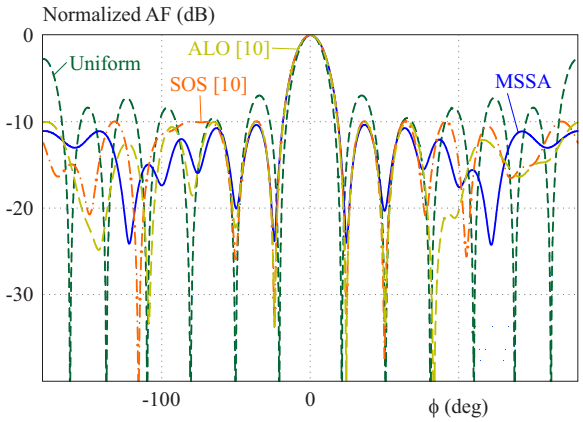

Fig. 11. Radiation patterns obtained by angular position-only control for 12-elements EAA

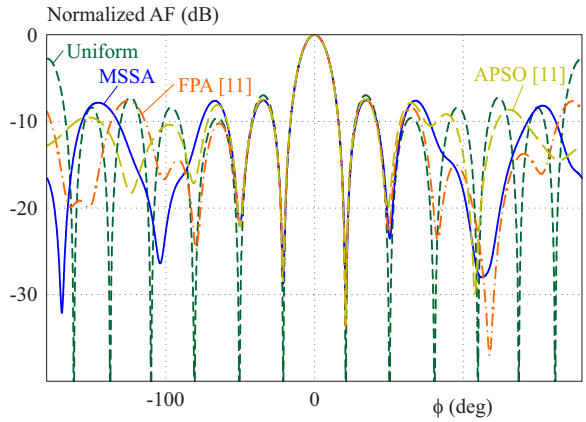

Fig. 12. Radiation patterns obtained by angular position-only control for 12-elements EAA

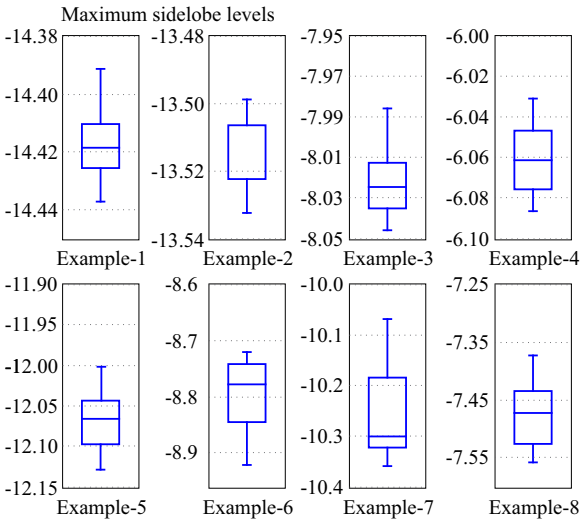

Fig. 13. Box-and-whisker plots of all design examples for 20 independent runs by the proposed algorithm

a fair comparison. Proposed algorithm is run for the 20 independent times and each run take around 96 s. The best MSL, mean, and standard deviation from these runs are obtained at -10.36 dB, -10.2619 dB, and 0.0910 dB, respectively. The patterns obtained by MSSA, ALO [10] and SOS [10] are illustrated in Fig. 11. We can say that the MSL value obtained by MSSA is better than those of ALO [10] and SOS [10].

In the last example, the FNBW value is chosen 42.2°as the same with the examples in FPA [11] and APSO [11]. The obtained patterns are shown in Fig. 12. The value of MSSA is better than the MSL values of the other compared algorithms.

The amplitude values for design examples EX1 to EX6 and position values for design EX7 and EX88 are in Tab.3 and Tab.4. Besides, box-and-whisker plots are illustrated in Fig. 13 to exhibit the overall statistical view of 20 independent runs for each design example. From this figure, it can be said that the MSSA algorithm shows little fluctuation and produces stable results.

5 Conclusions

In this study, MSSA as an improved version of the original SSA was used to synthesize eight different EAA examples with low MSL and constant FNBW values. For the first six examples, the excitation amplitude values of the antenna array elements were optimized whereas angular positions were optimized for each array element in the last two examples. Results were analyzed by using the well-known statistical parameters of best solution, mean

Table 3. The amplitude values of array elements obtained by the MSSA for design examples 1-6

	$[I_1, I_2, I_3, \ldots I_N]$
Fig. 3	[0.507 0.949 0.147 0.949 0.507 0.999 0.065 1.00]
Fig. 5	[0.292 0.996 0.307 0.996 0.291 0.999 0.300 1.000]
Fig. 7	[1.000 0.170 0.324 0.320 0.303 0.153 0.980 0.099 0.334 0.441 0.338 0.019]
Fig. 8	[0.864 0.000 0.557 0.788 0.167 0.000 1.000 0.000 0.174 0.775 0.563 0.000]
Fig. 9	[1.000 0.200 0.566 0.000 0.354 0.570 0.297 0.000 0.533 0.324 0.969 0.411 0.458 0.000 0.259 .0646 0.304 0.000 0.513 0.275]
Fig.10	[0.676 0.006 0.284 0.017 0.015 0.789 0.430 0.004 0.258 0.000 1.000 0.003 0.125 0.052 0.694 0.379 0.294 0.072 0.137 0.011]

Table 4. The position values of array elements obtained by the MSSA for design examples 7 and 8

	$[d_1, d_2, d_3, \ldots d_N]$ in $\lambda's$
Fig. 11	[16.330 16.414 40.135 82.731 120.658 164.678 194.736 239.155 277.395 319.438 343.151 343.761]
Fig.12	[15.195 72.751 91.192 105.086 151.296 171.125 190.298 210.999 260.142 264.071 285.150 344.844]

and standard deviation value of the solutions. The results were also shown by the boxplot method. To verify the achieved results, proper comparisons were performed with the results of ALO, SOS, FPA and APSO from the literature. We took care to keep FNBW values fixed for a fair comparison. From these comparisons, it is clearly seen that the proposed algorithm has obtained radiation patterns with lower MSL values than the other algorithms for all examples. It can also be said that the MSSA algorithm is suitable for the synthesis of antenna arrays with different geometrical structures thanks to its algorithmic simplicity and robustness.

References

[1] A. C. Balanis, *Antenna Theory: Analysis and Design*, Fourth Edition, New York: John Wiley & Sons, 2016.

[2] R. Bera, D. Mandal, R. Kar, and G. P. Sakti, "Optimal design of elliptical array antenna using opposition based differential evolution technique", *The Applied Computational Electromagnetics Society Journal*, no. 9, pp. 833-841, ISSN: 1054-4887, 2017.

[3] P. Das and J. S. Roy, "Thinning of elliptical antenna arrays using genetic algorithm", *International Journal of Engineering Research and Applications*, no. 4, pp. 225-229, ISSN: 2248-9622, 2014.

[4] K. Guney and A. Durmus, "Elliptical antenna array synthesis using backtracking search optimization algorithm", *Defence Science Journal*, no. 3, pp. 272–277, Doi: 10.14429/dsj.66.9583, 2016.

[5] A. Sharaqa and N. Dib, "Position-only side lobe reduction of a uniformly excited elliptical antenna array using evolutionary algorithms", *IET Microwaves, Antennas & Propagation*, no. 6, pp. 452–457, Doi: 10.1049/iet-map.2012.0541, 2013.

[6] A. Sharaqa and N. Dib, "Design of linear and elliptical antenna arrays using biogeography-based optimization", *Arabian Journal for Science and Engineering*, no. 4, pp. 2929–2939, doi: 10.1007/s13369-013-0794-8, 2014.

[7] A. Zare, "Elliptical antenna array pattern synthesis with ?xed side lobe level and suitable main beam beamwidth by genetic algorithm", *Majlesi Journal of Telecommunication Devices*, no. 4, pp. 1–8. ISSN: 2322-1550, 2013.

[8] M. Sainath, K. Chaitanya, and S. Raju, "Comparative analysis of reduction of side lobe level for concentric elliptical array and cylindrical elliptical array using differential evolution", *International Journal of Advances Electronics and Computer Science*, no. 1, pp. 62-66, Doi: IJAECS-IRAJ-DOIONLINE-6747, 2017.

[9] M. Khodier, "Optimization of elliptical antenna arrays using the cuckoo search algorithm", *IEEE-APS Topical Conference on Antennas and Propagation Wireless Communications (APWC)*, pp. 143-147, Doi: 10.1109/APWC.2019.8870399, 2019.

[10] N. Dib, A. Amaireh, and A. Al-Zoubi, "On the optimal synthesis of elliptical antenna arrays", *International Journal of Electronics*, no. 1, pp. 121-133, Doi: 10.1080/00207217.2018.1512658, 2019.

[11] V. V. S. S. S. Chakravarthy, P. S. R. Chowdary, N. Dib, and J. Anguera, "Elliptical antenna array synthesis using evolutionary computing tools", *Arabian Journal for Science and Engineering*, 1-18. Doi: 10.1007/s13369-021-05852-9, 2021.

[12] A. Amaireh, N. Dib, and A. Al-Zoubi, "The optimal synthesis of concentric elliptical antenna arrays", *International Journal of Electronics*, no. 3, pp. 461-479, Doi: 10.1080/00207217.2019.1661028, 2020.

[13] S. Mirjalili *et al* "Salp swarm algorithm: a bio-inspired optimizer for engineering design problems", *Advances Engineering Software*, pp. 163–191, Doi: 10.1016/j.advengsoft.2017.07.002, 2017.

[14] V. Kansal and J. S. Dhillon, "Emended salp swarm algorithm for multiobjective electric power dispatch problem", *Applied Soft Computing*, pp. 1-26, Doi: 10.1016/j.asoc.2020.106172, 2020.

[15] R. Abbassi, "An efficient salp swarm-inspired algorithm for parameters identification of photovoltaic cell models", *et al*, vol. 179, pp. 362-372, Doi: 10.1016/j.enconman.2018.10.069, 2019.

[16] H. M. Kanoosh, E. H. Houssein, and M. M. Selim, "Salp swarm algorithm for node localization in wireless sensor networks", *Journal of Computer Networks and Communications*, pp. 1-12, Doi: 10.1155/2019/1028723, 2019.

[17] M. H. Qais, H. M. Hasanien, and S. Alghuwainem, "Enhanced salp swarm algorithm: Application to variable speed wind generators", *Engineering Applications of Artificial Intelligence*, pp. 82-96, Doi: 10.1016/j.engappai.2019.01.011, 2019.

[18] D. Prabhakar and M. Satyanarayana, "Side lobe pattern synthesis using hybrid SSWOA algorithm for conformal antenna array", *Engineering Science and Technology-An International Journal*, no. 6, pp. 1169-1174, Doi: 10.1016/j.jestch.2019.06.009, 2019.

[19] D. H. Wolpert and W. G. Macready, "No free lunch theorems for optimization", *IEEE Transactions on Evolutionary Computation*, no. 1, pp. 67-82, Doi: 10.1109/4235.585893, 1997.

Received 1 October 2022