SIDELOBE REDUCTION OF LINEAR ANTENNA ARRAY BY POSITION-ONLY CONTROL USING NOVEL METAHEURISTIC OPTIMIZATION METHODS

Ali DURMUS[1*], Zafer YILDIRIM[2]

[1] Kayseri University, Vocational College, Department of Electricity and Energy, Kayseri, Turkey,
ORCID No: http://orcid.org/0000-0001-8283-8496
[2] Kayseri University, Faculty of Engineering Architecture and Design, Department of Electrical Electronics Engineering, Kayseri, Turkey,
ORCID No: http://orcid.org/0000-0003-3584-6770

Keywords
Artificial intelligence
Optimization
Antenna array synthesis
Sidelobe reduction

Abstract
In this study, the positions of linear antenna array elements (LAAs) are optimized for antenna design with the desired radiation patterns by using Honey Badger Algorithm (HBA) and Chameleon Swarm Algorithm (CSA) methods, which are the novel metaheuristic algorithms. LAAs with 10, 12, 24, and 32 elements are considered in the simulations. While designing the antenna, the half power beam width (HPBW) of the antenna array is kept at a minimum level and the sidelobe level (SLL) is suppressed as much as possible. In addition, results obtained with the swarm-based metaheuristic algorithms Artificial Bee Colony (ABC) and Particular Swarm Algorithm (PSO) methods, which are quite well known in the literature are compared to test the performance and accuracy of the HBA and CSA methods. As a result of these comparisons, the antenna design with high directivity, gain and desired pattern has been successfully obtained with the HBA and CSA optimization methods.

YENİ METASEZGİSEL OPTİMİZASYON YÖNTEMLERİNİ KULLANARAK YALNIZCA KONUM KONTROLÜYLE DOĞRUSAL ANTEN DİZİLERİNİN YAN DEMET SEVİYELERİNİN BASTIRILMASI

Anahtar Kelimeler
Yapay zeka
Optimizasyon
Anten dizi sentezi
Yan demet bastırma

Öz
Bu çalışmada, istenen ışıma diyagramlarına sahip anten dizi tasarımı için yeni metasezgisel metodlardan olan Bal Porsuğu Algoritması (Honey Badger Algorithm-HBA) ve Bukalemun Sürüsü Algoritması (Chameleon Swarm Algorithm-CSA) kullanılarak doğrusal anten dizi (Linear Antenna Array-LAA) elemanlarının konumları optimize edilmiştir. Simülasyonlarda 10, 12, 24 ve 32 elemanlı doğrusal anten dizileri dikkate alınmıştır. Anten tasarımı yapılırken dizinin yarı demet güç genişliği (Half Power Beam Width-HPBW) minimum seviyede tutulmuş ve yan demet seviyesi (Sidelobe Level-SLL) mümkün olduğunca bastırılmıştır. Ayrıca HBA ve CSA yöntemlerinin performans ve doğruluğunu test etmek için literatürde oldukça iyi bilinen sürü tabanlı metasezgisel algoritmalardan olan Yapay Arı Kolonisi (ABC) ve Parçacık Sürüsü Optimizasyonu (PSO) yöntemleri ile elde edilen sonuçlar karşılaştırılmalı olarak verilmiştir. Bu karşılaştırmalar sonucunda, HBA ve CSA optimizasyon yöntemleri ile yüksek yönlülük, kazanç ve istenilen ışıma diyagramlarına sahip anten tasarımları başarılı bir şekilde elde edilmiştir.

Araştırma Makalesi		Research Article	
Başvuru Tarihi	: 02.11.2021	Submission Date	: 02.11.2021
Kabul Tarihi	: 27.01.2022	Accepted Date	: 27.01.2022

* Sorumlu yazar; e-posta : alidurmus@kayseri.edu.tr

1. Introduction

With the developing communication technology, the need for objects to communicate with each other is increasing. This increases the interest in antenna design, which is one of the most important elements of communication technology. Today, one antenna cannot achieve sufficient gain and high directivity. Therefore, different antenna arrays with a certain geometry consisting of two or more antennas are needed (Balanis, 2011). Antenna arrays make different radiation patterns in order to communicate from one point to another. In communication systems, it has become very important to design antennas with the desired radiation pattern. Thanks to the rapid developments in biomedical, Industry 4.0, Internet of Things (IoT), and 5G technologies, the need for the design of different antennas arises. It is necessary to design more effective antenna arrays for uninterrupted and high-capacity communication and to reduce electromagnetic pollution that arises with increasing communication technology. The desired radiation pattern can be obtained by optimizing the positions, phases, and amplitudes of the antenna arrays. (Banerjee and Dwivedi, 2013; Chakravarthy et al., 2018; Durmus et al., 2021; Ebrahimzade et al., 2016; Kaur and Goyal, 2017; Khodier, 2019; Laseetha and Sukanesh, 2011; Laseetha and Sukanesh, 2012; Li et al., 2013).

The optimization process is a method of finding the best solution in the search space (Mirjalili and Dong, 2020). These methods are used to optimize the gain, especially in engineering problems. By optimizing, the gain of a system can be increased, and it can be brought closer to the desired aim. Nowadays, the use of metaheuristic optimization methods in antenna design to gain the desired radiation pattern is very popular. (Mangoud and Elragal, 2009; Liang et al., 2009; Recioui, 2012; Saxena and Kothari, 2016; Shrivastava and Cecil, 2012; Shrivastava, 2013; Singh and Salgotra, 2018; Singh and Salgotra, 2017). At the same time, metaheuristic optimization methods are used in many different optimization problems due to their fast convergence curves, independence of derivatives, and ability to avoid local minimums.

The aim of this study is to obtain an effective linear antenna array (LAA) with high directivity. In order to have these properties, three parameters, which are mostly called phases, positions and amplitudes are optimized in antenna arrays. Many different methods can be used to optimize antenna arrays. In this study, positions only of the LAA elements have been optimized with the Honey Badger Algorithm (HBA) and Chameleon Swarm Algorithm (CSA) methods, which have just entered the literature. In addition, the performance of novel HBA and CSA methods are compared with Artificial Bee Colony (ABC) and Particle Swarm Algorithm (PSO) methods, which are well known in the literature. In order to test the performance of all metaheuristic methods, LAAs with different numbers of elements is determined as 10, 12, 24, and 32. By keeping half power beam width (HPBW) to a minimum for the cost of the optimization problem, the efficiency of the antenna is increased, and high directivity is provided by suppressing the sidelobe level (SLL). During the optimization process, the aim is not only to design an effective array of antennas but also to quickly solve the antenna design problem of these methods. For the 10, 12, 24, and 32 elements LAA design, the positions of the elements are optimized with ABC, PSO, HBA, and CSA. The SLL and CPU time performance indicators of these metaheuristic methods are given comparatively.

Contributions of this study can be summarized as:

- Implementation of HBA and CSA, two recent metaheuristics, are applied to LAA synthesis with position-only optimization.
- HBA and CSA performances are compared with PSO and ABC, two well-known swarm-based optimization algorithms.
- Experiments are repeated 30 times due to the random nature of the algorithms and statistical results are also tabulated.
- CPU time consumptions of the algorithms are also compared.

The article is organized as follows: Section II presents the problem formulation. In Section 3, optimization algorithms are briefly explained. Comparative statistical data obtained with simulation results and optimization algorithms are given in Section 4. Finally, In Section 5 presented the conclusion.

2. Formulation of Problem

The design of the LAA with *2N* elements placed as found values throughout the x-axis is shown in Figure 1. When the origin point is taken as a reference, the antenna array with *2N* elements contains *N* antenna elements in both regions of the x-axis.

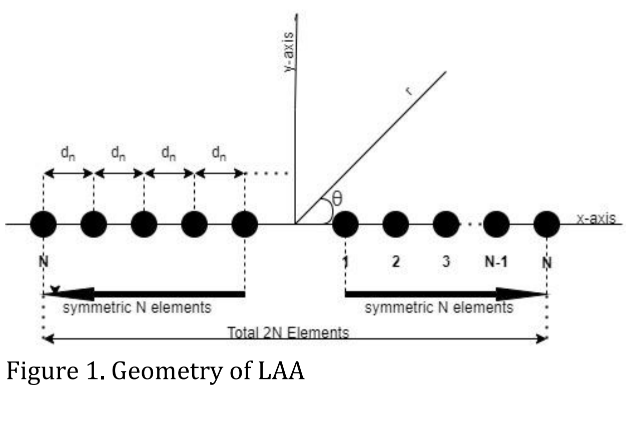

Figure 1. Geometry of LAA

The array factor (AF) of LAA with symmetrical N elements positioned throughout the x-axis is given in Equation 1 (Balanis, 2011):

$$AF = 2\sum_{n=1}^{N} I_n\, e^{j(n-1)\psi} \quad (1)$$

where I_n amplitude excitation of n_{th} element in LAA. ψ is represents as $\psi = (k\, d_n\, cos\gamma + \beta)$. β indicates the increasing phase excitation due to the previous element. The scanning angle is γ and k symbolizes the wave number and is formulated as $k=\frac{2\pi}{\lambda}$. The array factor formula can be shown as Equation 2 (Balanis, 2011):

$$AF = 2\sum_{n=1}^{N} I_n\, e^{j(n-1)(\frac{2\pi}{\lambda} dn\, cos\gamma+\beta)} \quad (2)$$

where d_n is distance inter-elements of n_{th} element in this LAA, regularly. The total number of elements in this LAA is $2N$, but since these antennas are the same and symmetrical on both sides, the parameter to be optimized is equal to half of the total number of antennas, N. In this work, phase values are taken as zero.

Only position values of LAAs with the desired radiation patterns are optimally determined by four different optimization algorithms namely PSO, ABC, HBA, and CSA. The purpose of antenna design is to transfer data to as long distance as desired without being affected by the electromagnetic pollution in the environment. Therefore, it is very important to suppress unwanted electromagnetic interferences in the environment. In order to prevent electromagnetic interferences, the SLLs of the radiation patterns should be good reduced. The aim here is to obtain a radiation pattern with a minimum SLL value and maximum performance. The HPBW value is also included in the fitness function, and this value has been determined as constant in all simulation studies to achieve desired patterns. The fitness function can be defined as Equation 3 (Durmus et al., 2021):

$$Fit_{function} = \begin{cases} I_{nf} & if\ HPBW_{obt} > HPBW_{des} \\ f_{SLL} & else \end{cases} \quad (3)$$

where $HPBW_{des}$ and $HPBW_{obt}$ are the desired HPBW value and obtained value of HPBW, respectively. I_{nf} is a penalty coefficient. f_{SLL} indicates the SLL of LAA that needs to be suppressed.

3. Metaheuristic Optimization Algorithms

3.1. Particle Swarm Algorithm (PSO)

Particle Swarm Algorithm (PSO) is one of the most well-known algorithms in the literature of optimization methods (Kennedy and Eberhart, 1995). This algorithm is inspired by the foraging behavior of bird and fish flocks in nature during migration, and the behavior of these swarms is mathematically modeled. Mathematical modeling of these swarms is made by taking into account the three characteristics of location, fitness value, and speed. The first parameter, position, specify the location of a member in this swarm in the search space. The second parameter, fitness value, is the value that indicates how suitable the birds in the flock are, determined by their proximity to the food. It updates the third parameter, its velocity, by taking into account the position of a bird in the flock in space and the position of the best bird in the flock in space. Accompanied by this information, the members of the herd share their positions with their herd and ensure that the herd moves towards the best positions. The swarm, on the other hand, moves by looking for a position where it is better than its previous position. Advantages of PSO: Derivative is not calculated in order not to get stuck in a local optimum and to reach the result faster. Because herd members share information with each other at all times, other members can use this information to better navigate towards the best positions. For more information on PSO, see the article by Kennedy and Eberhart (Kennedy and Eberhart, 1995).

3.2. Artificial Bee Colony (ABC)

The Artificial Bee Colony (ABC) (Karaboga and Basturk, 2009) optimization method is proposed by D. Karaboga in 2009. This method is one of the most well-known swarm-based algorithms in the literature. ABC method brought a different view to the foraging methods of bees by bringing some limitations in the search space. These limitations are kept equal to the number of members in the ABC and the number of bees. There are 3 types of bees, although there are 2 types of bees that have a duty and do not have a definite mission. The first type is worker bees, those who go to the source and bring food. The second type of onlooker bees determines which source they will go to in order to direct them and help collect resources faster. The third type is scout bees, bees that see their resources as insufficient, are bees that explore without being affected by any onlooker or attendant bees in order to find new resources. In addition, when the worker bees come to the hive after collecting food, they share information about the location and quality of the food source with the bees in the hive. Other bees also act in the presence of this information. By evaluating the quality of the honey in the source, the bees travel towards the best source, and thanks to this process, the bees approach the best result quickly. These limitations made by D. Karaboga have presented a unique algorithm by keeping the number of food sources equal to the members of the artificial bee colony. For more information, see D. Karaboga et al. (Karaboga and Basturk, 2009).

3.3. Honey Badger Algorithm (HBA)

Honey Badger Algorithm (HBA) (Hashim et al., 2022) has been brought to the literature by being inspired by the foraging and hunting behavior of the creature known as the honey badger in nature. The honey badger is known for its fearlessness in nature and is found in some African deserts and rainforests; it is a species of creature found in Southwest Asia and India. It finds and hunts 67 different species, including dangerous snakes, with its search strategy. Honey is one of the favorite foods of the honey badger. This animal can use some tools to locate honey. Honey badger has 2 types of foraging and hunting in nature. The first is called the digging phase. Honey badger finds prey by searching for it with their strong sense of smell in the digging phase. It approaches its prey by digging and wandering around. The second stage is called the honey stage. At this stage, the honey badger's sense of smell is not enough to search for honey so needs a guide. The honey badger, which is guided by a bird, has a common search and hunting strategy with the pilot bird. In this strategy, the honey badger follows the guided bird. The guide undertakes the task of searching for birds. Who offers a different search strategy than honey badger, finds honey thanks to the guided bird. But the bird cannot reach honey. Here, the honey badger's ability to use tools comes into play and reaches honey. In this way, the honey badger completes the missing parts in the search areas where it is insufficient, thanks to a guided bird. In Ref. (Hashim et al., 2022) shows that the HBA has been applied to many engineering problems and good results have been obtained. The unique search and exploitation ability of the honey badger is calculated as follows:

$$Div_j = \frac{1}{N}\sum_{i=1}^{N} median(x^j) - x_i^j, \qquad (4)$$

$$Div^t = \frac{1}{D}\sum_{j=1}^{D} Div_j, \ t = 1,2,\dots,t_{max} \qquad (5)$$

i and j represent the order and size of the honey badger, respectively. x_i^j indicates the honey badger solution candidate. Div_j is the average variety for dimensions. Div^t is the average of this dimensional diversity (D) taken at the end of all iterations. The following equation is used to calculate the percentage of exploitation and explore after the variety is calculated:

$$Exploration\% = \frac{Div^t}{\max(Div)} \times 100; \qquad (6)$$

$$Exploitation\% = \frac{|Div^t - \max(Div)|}{\max(Div)} \times 100 \qquad (7)$$

t_{max} is expressed as max (Div) diversity in maximum iteration. For more information, see the work of Hashim, Fatma A., et al. (Hashim et al., 2022).

Each food or prey in the HBA ecosystem represents a possible solution to the LAA synthesis problem. In this case, each food or prey position resembles the optimal position of LAA elements. The food amount of each position mimics the SLL of the designed antenna. The higher the food or prey amount the better the SLL obtained. Flowchart of HBA is given in Figure 2.

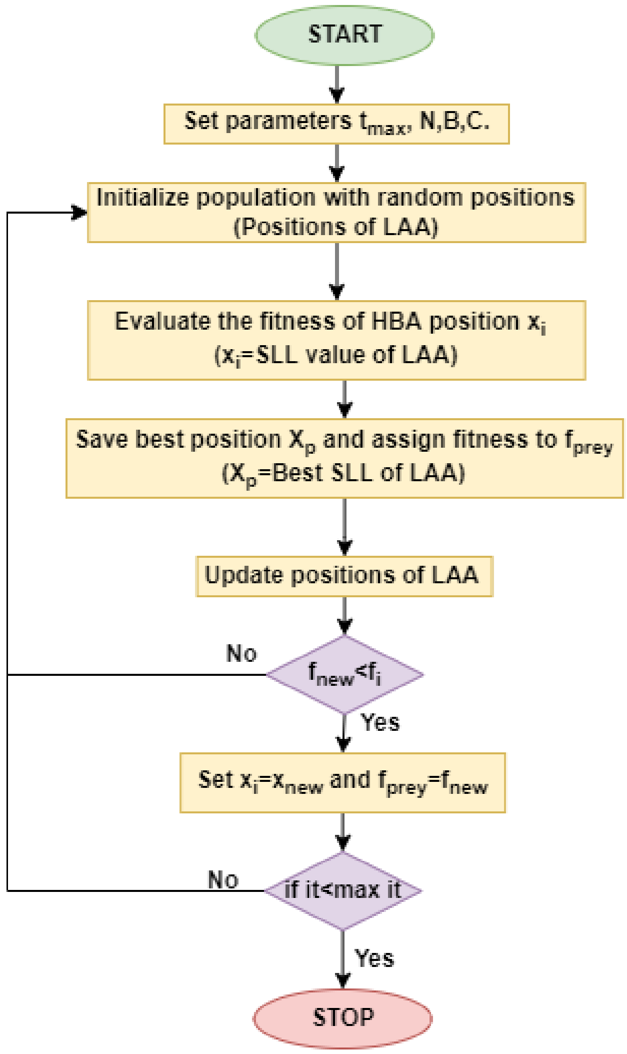

Figure 2. Flowchart of HBA

3.4. Chameleon Swarm Algorithm (CSA)

Known for their ability to change color to adapt to the environment, chameleons are good hunters. This feature makes chameleons a good explorers. The creators of the CSA mention that chameleons have a special explore the ability to discover their prey with this explore feature (Braik, 2021).

Another feature of chameleons is that their two eyes can move independently of each other. With wide-angle eyes, they can follow their prey while moving on the one hand. Chameleons can also use their tongues very quickly for hunting. Thanks to these abilities, it can be claimed to be a good hunter. The ability to use the tongue quickly can be likened to rapid convergence. They can also explore the search space well, with the ability of their eyes to explore 360 degrees. In this algorithm, two types of exploring and four types of hunting abilities are mentioned. Its exploring capabilities are desert explore and tree explore. Their first hunting ability is to turn their eyes back and converge quickly without attracting the attention of their prey. Its second ability is to turn its eye to the right and visually direct it preys. Its third ability is that it can hunt its prey from afar without getting close, thanks to its long tongue. Its fourth ability is to hunt by approaching its prey. With its exploration capability, this algorithm contributes to hunting by exploring the entire space. Dominating the entire space, the chameleon quickly converges towards its prey and has a fast convergence curve. Compared with a total of 67 non-comparison tests, including the Chameleon Swarm Algorithm (CSA) EC-2015 and CEC-2017 test suites. Researchers state that it is obtained very well from the tests in Ref (Braik, 2021). The CSA's explore can be calculated as follows:

$$Y_{t+1}^{i,j} = \begin{cases} Y_t^{i,j} + p_1(P_t^{i,j} - G_t^j)r_2 + p_2(G_t^j - Y_t^{i,j})r_1, & r_i \geq P_p \\ Y_t^{i,j} + \mu\left((u^j - l^j)r_3 + l_b^j\right) sgn(rand - 0.5), & r_i < P_p \end{cases} \quad (8)$$

i indicates the order of chameleon. t represents the number of iterations, while j symbolizes the problem size of the t. iteration. $Y_{t+1}^{i,j}$ and $Y_t^{i,j}$ refers to the position that will be in the next iteration and the current position, respectively. $P_t^{i,j}$ is the chameleon in the best position in the t. iteration. G_t^j, t. specifies the best position in all iterations up to the t. iteration. p_1 and p_2 are the numbers that exist to control the ability to reconnaissance. r_1 , r_2 and r_3 random numbers selected in the range *0* through 1. P_p represents the possibility of chameleon hunting prey. $sgn(rand - 0.5)$ is the value that controls search and discovery capability in the *range* [-1,1]. μ is the parameter defined as a function of iterations. For more information, see the article of Malik Shehadeh Braik (Braik, 2021).

Each prey to be hunted by a chameleon in the CSA shows a candidate solution to the LAA synthesis problem which denotes the optimal position of LAA elements. The quality of the prey represents the SLL of the designed antenna. The higher the quality the better the SLL obtained. Flowchart of CSA is given in Figure 3.

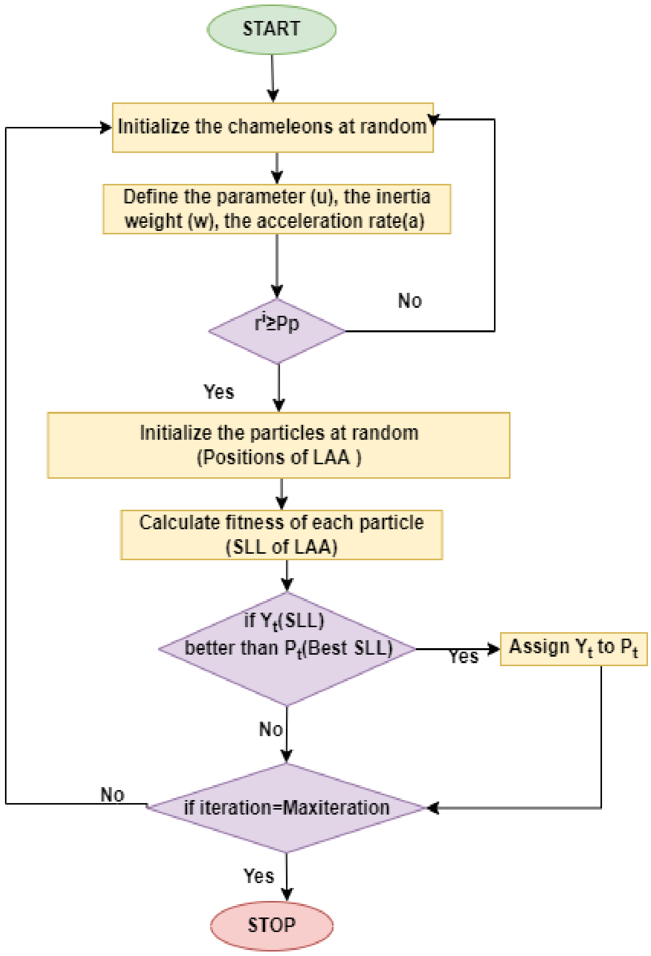

Figure 3. Flowchart of CSA

4. Numerical Results

In this study, LAA's with 10, 16, 24, and 32 elements are optimized and compared with 4 different metaheuristic optimization methods. The first two algorithms are swarm-based metaheuristic algorithms PSO and ABC. These swarm-based algorithms are well known in the literature. The other two algorithms are nature-based metaheuristic algorithms HBA and CSA. PSO and ABC algorithms are chosen because they are both swarm-based and one of the most well-known algorithms in the literature. Other algorithms are chosen because they are newly introduced to the literature. In order to test the performance of these new optimization methods in antenna array synthesis, antennas with different numbers of elements are investigated.

The aim in all simulations is to obtain the radiation pattern that achieves the minimum SLL values with fixed HPBW values. Simulations are made by a personal computer with an i5 processor at 2.5 GHz. In the optimization methods used in all simulation studies, the population size is determined as 50 and the maximum number of iterations is 500. The MATLAB software program is used for the analysis of all simulations.

Statistical comparisons of PSO, ABC, HBA, and CSA methods are made in terms of CPU time and SLL values. For this statistical analysis, all algorithms are randomly run 30 times. As a result of these runs, the best values and mean values are tabulated in terms of SLL and CPU time.

Table 1

The Statistical Values of SLL and CPU Time Obtained by Metaheuristic Optimization Methods

	Methods	PSO	ABC	HBA	CSA
10 Elements	SLL_{Best} (dB)	-21.355	-21.065	-21.430	-21.430
	SLL_{Mean} (dB)	-19.441	-20.247	-21.211	-20.809
	$CPUT_{min}$ (sec)	6.758	10.728	2.331	2.154
	$CPUT_{Mean}$ (sec)	7.149	11.148	2.448	2.232
	$CPUT_{Max}$ (sec)	7.807	12.053	2.753	2.580
16 Elements	SLL_{Best} (dB)	-24.579	-22.799	-24.826	-23.458
	SLL_{Mean} (dB)	-22.194	-22.240	-23.644	-22.643
	$CPUT_{Min}$ (sec)	7.548	11.998	3.133	3.028
	$CPUT_{Mean}$ (sec)	7.797	12.098	3.178	3.048
	$CPUT_{Max}$ (sec)	8.133	12.566	3.218	3.101
24 Elements	SLL_{Best} (dB)	-26.173	-24.086	-26.515	-26.099
	SLL_{Mean} (dB)	-23.782	-23.173	-25.872	-24.065
	$CPUT_{Min}$ (sec)	8.875	13.811	4.293	4.211
	$CPUT_{Mean}$ (sec)	9.009	13.909	4.378	4.2686
	$CPUT_{Max}$ (sec)	9.114	14.057	4.435	4.3231
32 Elements	SLL_{Best} (dB)	-25.232	-23.743	-25.423	-25.114
	SLL_{Mean} (dB)	-23.592	-22.989	-24.147	-23.166
	$CPUT_{Min}$ (sec)	10.024	15.718	5.496	5.4139
	$CPUT_{Mean}$ (sec)	10.201	15.804	5.591	5.495
	$CPUT_{Max}$ (sec)	10.299	15.909	5.852	5.569

A comparison of four methods in terms of SLL and CPU time is given in Table 1. As shown in Table 1, the CSA method is faster than the other optimization techniques. When examined for the suppression of sidelobe levels, the results obtained by the HBA method are generally better than PSO, ABC, and CSA. The best SLL value for the 10-element LAA is obtained by the CSA algorithm. At the same time, this algorithm achieved the optimum result faster than PSO, ABC, and HBA algorithms with 2.15 seconds. The HBA method found the best SLL value of -24.8261dB for the array with 16 elements. It reached this result in about 3.2 seconds. In the linear array with 24 elements, the HBA method reached the best SLL value. The HBA method reached the best SLL value in 4.29 seconds. Finally, in the 32-element linear array, the HBA method found the SLL value better than the other three optimization methods in 5.49 seconds. When we look at metaheuristic methods in general, the fastest method compared to other algorithms has been the CSA method. The method that achieved the best SLL value is the HBA method. In addition, HBA and CSA methods are much faster than PSO and ABC methods and produce better values. The position values of LAA obtained with PSO, ABC, HBA, and CSA are given in Table 2.

Table 2

The Positions of LAA Elements Obtained by Metaheuristic Optimization Methods

	Method	Position of LAA elements ($d_1, d_2, \ldots, d_n$)
10 Elements	PSO	[0.3287 0.4080 0.9853 1.3782 2.0463]
	ABC	[0.3707 0.3707 0.9882 1.3716 2.0491]
	HBA	[0.3736 0.3736 1.0010 1.3945 2.0753]
	CSA	[0.3589 0.3610 0.9644 1.3435 1.9994]
16 Elements	PSO	[0.1437 0.5472 0.6130 1.2200 1.3155 1.8745 2.3196 3.0091]
	ABC	[0.2375 0.5203 0.9902 1.3452 1.8401 2.3203 2.9808 3.7430]
	HBA	[0.3390 0.3390 0.6981 1.2578 1.2932 1.8997 2.3360 3.0464]
	CSA	[0.1072 0.4870 0.7291 1.1064 1.4357 1.8527 2.3572 2.9865]
24 Elements	PSO	[0.1417 0.4794 0.7390 1.0727 1.4707 1.6627 2.1890 2.4516 2.9830 3.4487 4.1356 4.9066]
	ABC	[0.0241 0.5402 0.7810 1.1629 1.5228 1.8772 2.3196 2.7078 3.1661 3.7285 4.3909 5.1390]
	HBA	[0.3094 0.3094 0.7306 1.0910 1.2924 1.7086 2.0190 2.4000 2.8399 3.3109 3.9545 4.7147]
	CSA	[0.2040 0.4344 0.8208 1.1062 1.4964 1.7814 2.2445 2.5812 3.0981 3.5940 4.3027 5.09809]
32 Elements	PSO	[0.1886 0.5935 0.9212 1.3643 1.6757 2.1449 2.5301 2.9816 3.3957 3.8569 4.4497 4.9975 5.5445 6.3796 7.2335 8.0756]
	ABC	[0.1480 0.6483 1.0024 1.4041 1.8720 2.3341 2.7026 3.2151 3.6673 4.2105 4.7801 5.3231 6.0406 6.8581 7.7682 8.4942]
	HBA	[0.0160 0.7334 0.7960 1.4105 1.6247 2.1738 2.5253 2.9774 3.4047 3.8513 4.4427 4.9725 5.5481 6.3888 7.2280 8.0959]
	CSA	[0.1536 0.6081 0.9402 1.3590 1.7417 2.1169 2.5645 3.0009 3.4808 3.9266 4.4680 4.9916 5.6481 6.4891 7.2552 8.1626]

As can be seen from numerical results, the newly introduced HBA and CSA optimization methods have been very successful in suppressing the SLL of LAA's. The position values given in Table 2 are the data of the optimum radiation pattern obtained by four different metaheuristic methods, namely PSO, ABC, HBA, and CSA. The comparative radiation patterns obtained with the position values of the linear array with 10, 16, 24, and 32 elements are given in Table 2 are shown in Figure 4. Figure 5 shows the convergence curves of LAAs with the different numbers of elements obtained by 30 random runs. According to Figure 5, the convergence curves obtained with HBA are better than the other three methods. As can be seen from Figure 5, HBA and CSA methods reached the optimum result faster than PSO and ABC optimization method

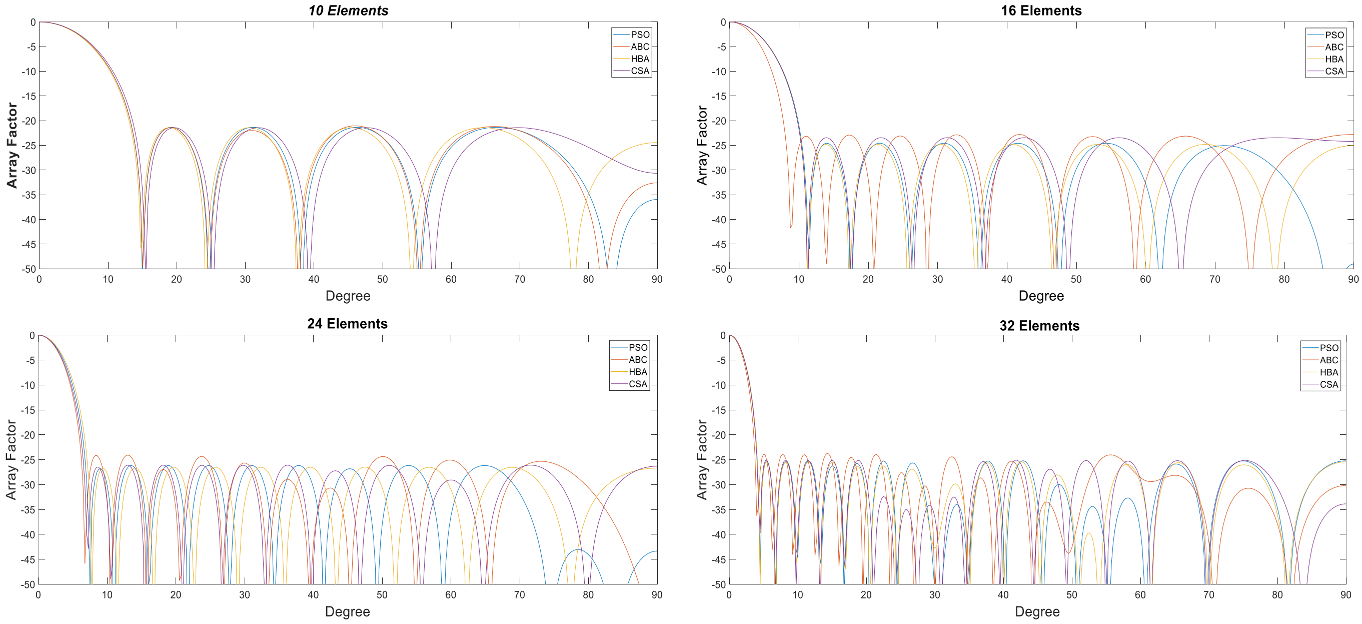

Figure 4. Radiation Patterns Obtained with PSO, ABC, HBA, and CSA for 10, 16, 24 and 32 Elements of LAA

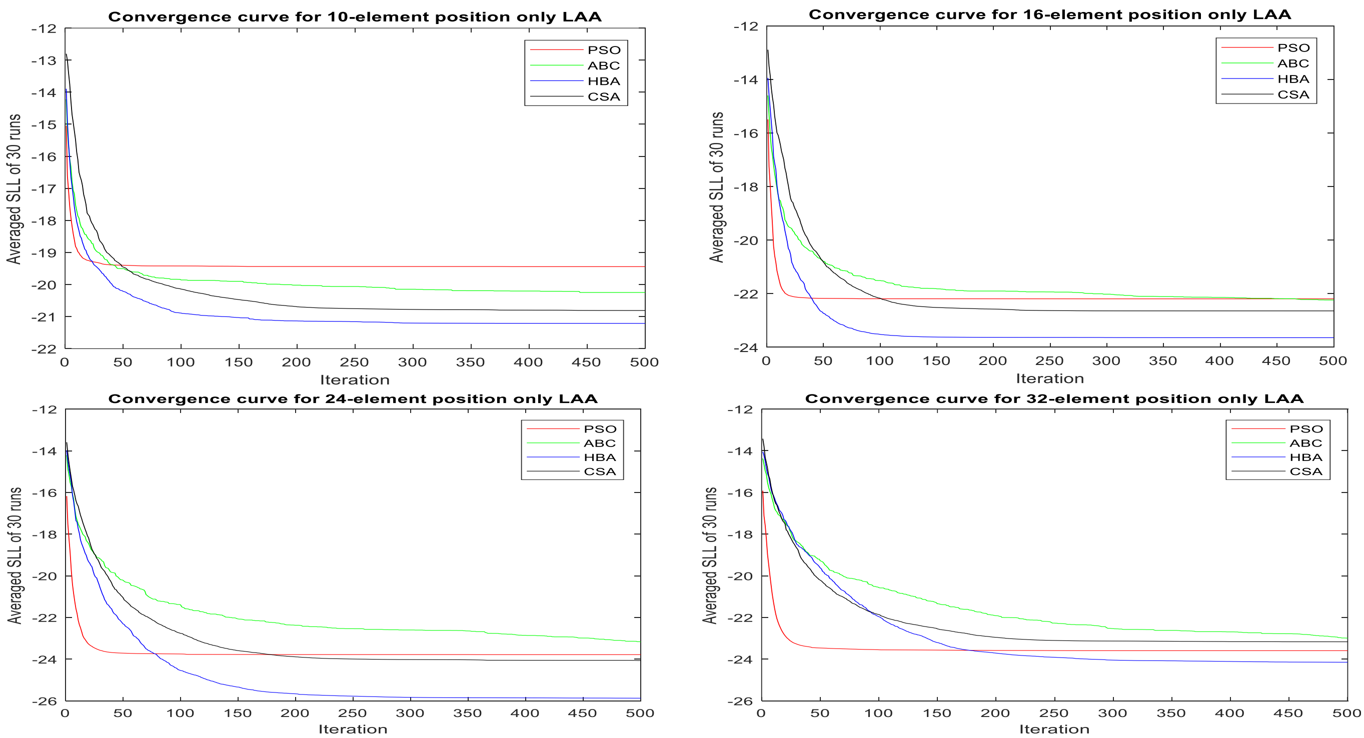

Figure 5. Convergence Curve Obtained with PSO, ABC, HBA and CSA for 10, 16, 24, and 32 Elements of LAA

5. Conclusion

In this study, the SLLs of LAAs are suppressed by four metaheuristic methods PSO, ABC, CSA, and HBA by controlling the positions only of the elements. The performances of the newly introduced HBA and CSA methods are compared with ABC and PSO. The results show that HBA and CSA methods obtained better SLL values than ABC and PSO. Moreover, HBA and CSA perform ~2x faster than PSO and ~3x faster than ABC compared to the CPU computation times. These methods have revealed that they can also be used for different antenna designs.

Contribution of Researchers

Ali DURMUS, contributed to determining the problem, revealing the original values of the study, and writing the article; Zafer YILDIRIM contributed to the literature review and the acquisition of experimental results from the computer environment.

Thanks

We would like to thank Kayseri University Scientific Research Projects Coordination Unit (KAYU/BAP). This work is supported by Kayseri University Scientific Research Projects Coordination Unit with the project numbered FYL-2021-1050.

Conflict of Interest

No conflict of interest is declared by the authors.

References

Balanis, C. A. (Ed.). (2011). *Modern antenna handbook*. John Wiley & Sons. doi: https://doi.org/ 10.1002/9780470294154

Braik, M. S. (2021). Chameleon Swarm Algorithm: A bio-inspired optimizer for solving engineering design problems. *Expert Systems with Applications*, 174, 114685. doi: https://doi.org/10.1016/j.eswa.2021.114685

Banerjee, S. M. I. T. A. & Dwivedi, V. V. (2013). Review of adaptive linear antenna array pattern optimization. *International Journal of Electronics and Communication Engineering*, 2(1), 25-42.

Chakravarthy, V. V. S. S. S. Chowdary, P. S. R. Panda, G. Anguera, J. Andújar, A. & Majhi, B. (2018). On the linear antenna array synthesis techniques for sum and difference patterns using flower pollination algorithm. *Arabian Journal for Science and Engineering*, 43(8), 3965-3977. doi: https://doi.org/10.1007/s13369-017-2750-5

Durmus, A. Kurban, R. & Karakose, E. (2021). A comparison of swarm-based optimization algorithms in linear antenna array synthesis. *Journal of Computational Electronics, 1-12*. doi: https://doi.org/10.1007/s10825-021-01711-w

Ebrahimzade, A. Abedi, H. & Hesari, M. (2016). Gravitational Search Algorithm used to synthesis a planar array antenna for nulling control and side lobe level reduction. *In 2016 8th International Symposium on Telecommunications (IST),* 391-395. IEEE. doi: https://doi.org/10.1109/ISTEL.2016.7881848

Hashim, F. A., Houssein, E. H., Hussain, K., Mabrouk, M. S., & Al-Atabany, W. (2022). Honey badger algorithm: new metaheuristic algorithm for solving optimization problems. *Mathematics and Computers in Simulation,* 192, 84-110. doi: https://doi.org/10.1016/j.matcom.2021.08.013

Karaboga, D., & Basturk, B. (2007). A powerful and efficient algorithm for numerical function optimization: artificial bee colony (ABC) algorithm. *Journal of global optimization*, 39(3), 459-471. doi: https://doi.org/ 10.1007/s10898-007-9149-x

Kaur, J. & Goyal, S. (2017). A Comparative study on linear array antenna pattern synthesis using evolutionary algorithms. *International Journal of Advanced Research in Computer Science*, 8(5).

Kennedy, J., & Eberhart, R. (1995, November). Particle swarm optimization. *In Proceedings of ICNN'95-international conference on neural networks*, 4, pp. 1942. doi: https://doi.org/10.1109/ICNN.1995.488968

Khodier, M. (2019). Comprehensive study of linear antenna array optimisation using the cuckoo search algorithm*. IET Microwaves, Antennas & Propagation*, 13(9), 1325-1333. doi: https://doi.org/10.1049/iet-map.2018.5649

Laseetha, T. J. & Sukanesh, R. (2012). Investigations on the synthesis of uniform linear antenna array using biogeography-based optimization techniques. *International Journal of Bio-Inspired Computation*, 4(2), 119-130. IEEE. doi: https://doi.org/10.1504/IJBIC.2012.047186

Laseetha, T. J. & Sukanesh, R. (2011). Synthesis of linear antenna array using genetic algorithm to maximize side lobe level reduction*. International Journal of Computer Applications,* 20(7), 27-33. doi: https://doi.org/ 10.1.1.206.4531

Li, X. Li, W. T. Shi, X. W. Yang, J. & Yu, J. F. (2013). Modified differential evolution algorithm for pattern synthesis of antenna arrays. *Progress in Electromagnetics Research,* 137, 371-388. doi: https://doi.org/ 10.2528/PIER13011207

Liang, S. Fang, Z. Sun, G. Liu, Y. Qu, G. & Zhang, Y. (2020). Sidelobe reductions of antenna arrays via an improved chicken swarm optimization approach. *IEEE Access*, 8, 37664-37683. doi: https://doi.org/10.1109/ACCESS.2020.2976127

Mangoud, M. A. A. & Elragal, H. M. (2009). Antenna array pattern synthesis and wide null control using enhanced particle swarm optimization. *Progress In Electromagnetics Research B*, 17, 1-14. doi: https://doi.org/ 10.2528/PIERB09070205

Mirjalili, S. & Dong, J. S. (2020). Multi-objective optimization using artificial intelligence techniques. *Springer*. doi: https://doi.org/10.1007/978-3-030-24835-2

Recioui, A. (2012). Side lobe level reduction in linear array pattern synthesis using particle swarm optimization. *Journal of Optimization Theory and Applications*, 153(2), 497-512. doi: https://doi.org/ 10.1007/s10957-011-9953-9

Saxena, P. & Kothari, A. (2016). Optimal pattern synthesis of linear antenna array using grey wolf optimization algorithm. *International Journal of Antennas and Propagation*. doi: https://doi.org/10.1155/2016/1205970

Shrivastava, S. & Cecil, K. (2012). Performance analysis of linear antenna array using genetic algorithm. *International Journal of Engineering and Innovative Technology (IJEIT),* 2(5). doi: https://doi.org/ 10.1.1.1065.9969

Shrivastava, S. (2013). Broadside linear antenna array synthesis using genetic algorithm. *International Journal of Scientific Research Engineering & Technology (IJSRET),* 2, 332-336. doi: https://doi.org/ 10.1.1.1058.7409

Singh, U. & Salgotra, R. (2018). Synthesis of linear antenna array using flower pollination algorithm. *Neural Computing and Applications*, 29(2), 435-445. doi: https://doi.org/10.1007/s00521-016-2457-7

Singh, U. & Salgotra, R. (2017). Pattern synthesis of linear antenna arrays using enhanced flower pollination algorithm. *International Journal of Antennas and Propagation*. doi: https://doi.org/10.1155/2017/7158752