International Journal of Microwave and Wireless Technologies

cambridge.org/mrf

Chebyshev-based array for beam steering and null positioning using modified ant lion optimization

Hrudananda Pradhan[1], Biswa Binayak Mangaraj[1] and Santanu Kumar Behera[2]

[1]Department of ETCE, Veer Surendra Sai University of Technology, Burla, Odisha 768018, India and [2]Department of ECE, National Institute of Technology, Rourkela, Odisha 769008, India

Research Paper

Cite this article: Pradhan H, Mangaraj BB, Behera SK (2022). Chebyshev-based array for beam steering and null positioning using modified ant lion optimization. *International Journal of Microwave and Wireless Technologies* **14**, 143–157. https://doi.org/10.1017/S1759078721000295

Received: 28 August 2020
Revised: 11 February 2021
Accepted: 11 February 2021
First published online: 12 March 2021

Key words:
Chebyshev array; smart antenna array; beam-steering; null positioning; modified ant lion optimization

Author for correspondence:
Biswa Binayak Mangaraj,
E-mail: bbmangaraj@yahoo.co.in

Abstract

A modified ant lion optimization (MALO) algorithm is proposed in this article, for the synthesis of Chebyshev-based arrays by optimizing amplitudes and phases of excitations, and element spacings. Modification in ant lion optimization is achieved by hybridizing it with chaotic particle swarm optimization. The optimization process is employed to obtain an array pattern with the least possible sidelobe level. Close-in sidelobe level minimization for optimum pattern synthesis is suggested. Instead of only steering the main beam towards the desired direction presented by some popular optimization methods, the beam steering along with null positioning in other specified direction is also achieved employing MALO. Considering the arrays with the same design parameters and the results of other optimization algorithms, the performance of MALO is evaluated. The results show that MALO provides considerable improvements in an array pattern compared to the arrays optimized using other optimization algorithms and the uniform array.

Introduction

In recent years, smart antenna technology is becoming increasingly popular for mobile communication systems. Enhanced directivity desired by this technology is achieved through an antenna array. The controlling mechanism to the elements of an array provides maximum radiation/reception in specific directions (beam steering). Along with this null steering (no radiation/reception) is accomplished in interfering directions by suppressing the sidelobe levels (SLLs). These are the vital requirements in smart antenna-based modern wireless communication system. In most wireless applications, enhanced communication is achieved through minimization of close-in SLL (CISLL), adjacent to the main lobe. SLL, CISLL minimization, beam steering, and null positioning are the current area of research and highly demanding in nature in the domain of antenna array design.

Conventional methods such as recursive least squares, least mean square, constant modulus algorithm, and many more are used for null positioning and beam-steering applications in smart antenna arrays [1, 2]. Many linear and nonlinear design equations are solved to achieve optimal results with the help of these methods. When the array size increases, it becomes challenging to solve those equations to get the desired solutions. The difficulties of solving such design equations can be overcome by the use of different evolutionary meta-heuristic optimization algorithms. In the past, several optimization algorithms are implemented in the synthesis of various antenna arrays for different applications. In [3–9], genetic algorithm (GA) and non-dominated sorting GA are employed in optimizing linear, planar, and circular array structures. Phase and amplitude excitations of the elements are optimized using GA to achieve radiation patterns of the desired SLL. Element positions of a linear array are optimized in [10], using ant colony optimization (ACO) for SLL minimization and null positioning. Spider monkey optimization (SMO) algorithm is introduced in [11]. Inter element positions of a linear array and the structural dimensions of an E-shaped microstrip patch antenna are optimized for synthesizing their array of factors. In [12], strawberry algorithm (SBA) is used for linear and circular array synthesis. The algorithm optimizes the amplitude excitations, and positions of the arrays for SLL minimization. Particle swarm optimization (PSO) in [13, 14] effectively deals with amplitude excitations, phases, and positions for the synthesis of linear and circular arrays. SLL, CISLL minimization, beam steering, null placement, and minimization of transmission power are achieved. Comprehensive learning PSO is implemented to suppress SLL and null control in linear array and Yagi–Uda array [15]. Mikki and Kishk [16] introduced quantum PSO (QPSO) for optimization of amplitude current distribution of a linear array. Optimal dipoles that yield radiation patterns equivalent to a cylindrical dielectric resonator antenna are also found using QPSO. In [17], QPSO is implemented for radiation pattern correction when some elements of a uniform linear array

are entirely defective. Yue *et al.* in [18] applied chaotic PSO (CPSO) for estimation of the better angle of arrival for mobile positioning application. The CPSO is also considered for finding an optimal transmission power of a planar array in [19]. Through this method, suppression of the SLL is achieved by optimizing the aperture, number of elements, and the minimum spacing between the elements. PSO is combined with discrete Green's function in [20] to accomplish appropriate dual-band patch antenna topologies. Ant lion optimization (ALO) algorithm is used in [21–23] for optimization of element positions, and current amplitudes of linear, elliptical, and circular arrays to achieve SLL, CISLL minimization, and null positioning. Other optimization algorithms such as Taguchi's optimization [24], biogeography-based optimization (BBO) [25], cat swarm optimization (CSO) [26], simulated annealing (SA) [27], symbiotic organisms search (SOS) [28], flower pollination algorithm (FPA) [29], whale optimization algorithm (WOA) [30], and moth flame optimization (MFO) [31] are implemented for SLL reduction and null control applications. In [32–37], the algorithms like MFO, BBO, imperialist competitive algorithm, GA, firefly algorithm (FA), harmony search algorithm, PSO are used to optimize linear, planar, and circular arrays that generate shaped beams (such as flat-top, isoflux, and cosecant beam). Amplitudes, phases, and powers of excitations, element spacings, radius of the rings are optimized to generate desired shape beams with low peak SLL. In recent years, researchers are showing interest in the implementation of hybrid algorithms, which provide more accuracy of system parameters. Two different types of optimization algorithms are merged to form a hybrid algorithm. Li *et al.* in [38] developed HIGAPSO algorithm by hybridizing improved GA (IGA) and improved PSO (IPSO). Excitation amplitudes and phases of the signals applied to elements of the spherical conformal array are optimized to minimize average maximum SLL. Genetical swarm optimization (GSO) is introduced in [39] for the optimization of unequally-spaced annular ring arrays. Half of the total population in GSO is created by PSO and the remainder is created by GA. In [40], hybrid algorithm HBMO/TS built on honey bees mating optimization (HBMO) and tabu search (TS) is applied to find the complex excitation weight factors of an adaptive array for steering the main beam in the desired direction. Circular array optimization is carried out by the hybrid optimization algorithm formed by merging ALO and grasshopper optimization algorithm in [41]. The number of array elements, current excitations, phases, inter-element spacings along the circumference is optimized to achieve minimum SLL. Salp swarm WOA is suggested by Prabhakar and Satyanarayana in [42]. Amplitude excitations and phases of the conformal array elements are optimized for pattern synthesis. Hybrid algorithms eliminate the drawbacks of an individual algorithm by accomplishing the advantages of their constituent algorithms. The benefits of both algorithms are considered to increase optimization performance by achieving improved accuracy. Improved performances of hybrid algorithms inspired us to propose a new hybrid algorithm, modified ALO (MALO), based on ALO and CPSO.

ALO is proposed in [43] by Mirjalili. It is applied to various benchmark functions, and the performances of it are evaluated. The performances are compared with famous optimization techniques such as bat algorithm (BA), FA, GA, CS, PSO, and FPA. ALO provides improved results, and in most of the test functions, it performs better than the other algorithms. Many engineering problems are solved using ALO. ALO is applied in [21–23] to obtain optimal linear, elliptical, and circular arrays. The results obtained using ALO are compared with uniform array and with the arrays optimized utilizing ACO, GA, PSO, CSO, BBO, SOS, and MFO. The performance of ALO-based array designs is much better than the arrays designed using other optimization algorithms and the uniform array. Hence, ALO is a suitable algorithm for antenna array optimization problems. On the other hand, CPSO improves accuracy and searching capability of basic PSO. CPSO is an improved PSO by embedding chaotic mapping to the basic PSO. PSO [44] was developed by Eberhart and Kennedy in 1995 and was modified by Shi and Kennedy in 1998 [45], whereas chaos-embedded PSOs were proposed by Alatas *et al.* in 2009 [46]. In CPSO, the paths of the particles are guided by the chaotic factors. Chaotic factor helps CPSO to avoid local minimums more easily than the basic PSO and provides improved results. Several variants of PSO are also evolved by modification of the basic PSO. Optimization problems in almost all fields are solved using traditional PSO and its variants. This way, PSO became most popular among all the evolutionary algorithms. In the past works, ALO and CPSO provided impressive performances. The advantages of both of these algorithms motivated to combine them for achieving a hybrid algorithm. The prime objective of framing the hybrid algorithm is to expect still better performance by merging the advantages of ALO and CPSO. The proposed algorithm is applied to some unimodal and multimodal benchmark functions available in [43]. The performance indices (mean and standard deviations) and characteristic curves of the benchmark functions are analyzed. Performance indices of MALO on the benchmark functions are compared with that of other optimization algorithms available in [43]. The indices demonstrate that the performance of MALO is superior than other algorithms. The convergence curves of the unimodal functions depict the high exploitation behavior of this hybrid algorithm. The multimodal function-characteristics shows that the exploration ability of MALO is also high. High exploration and exploitation behavior are the main advantage of MALO. These features enable the MALO algorithm to reach the global optimum, avoiding local optima. The results and convergence curves of the MALO on benchmark functions suggest that MALO can be employed as a substitute algorithm for different optimization problems.

Linear arrays are considered for optimization by most of the researchers. These arrays are the best suitable option to implement some new algorithms on them to validate the effectiveness of the algorithms. This is because the results of the numerous algorithms are readily available in the literature for comparison. Linear arrays, in general, are of three distribution types as uniform, binomial, and Chebyshev. Out of these three, Chebyshev array produces the smallest possible SLL for a given beam width between the first nulls. In this array, element excitation coefficients are associated with Chebyshev polynomials. The polynomials and their evaluations are available in [1]. In this article, Chebyshev-based linear arrays are considered for optimization utilizing the proposed MALO. Four examples of Chebyshev-based linear arrays are optimized. Out of these four examples, three examples are the same array problems presented in [13] using ACO, in [14] using SMO, in [12] using SBA, in [13] using PSO, in [21] using ALO, in [24] using Taguchi, in [25] using BBO, in [26] using CSO, in [28] using SOS, and in [31] using MFO. The outcomes of these arrays are compared with that of the conventional uniform arrays and with the algorithms cited above. The comparisons consider isotropic elements, the same parameter specifications (spacing between the elements,

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

amplitudes, and phases), and without a constant first-null beamwidth (FNBW) as in the previous works. In this way, appropriate comparisons between MALO and other algorithms are performed. The examples illustrate the benefits of higher exploration and exploitation behavior of MALO. The MALO provides the optimum parameters which offer a considerable reduction in peak SLL and CISLL. Superior beam steering in a specific direction is another advantage of this algorithm. The fourth design example introduces a new approach in the field of smart antenna technology. The significant contribution of this example is that the main beam can be steered in the signal of interest (SOI) direction while, placing the nulls in the signal not of interest (SNOI) directions.

The combination of CPSO with ALO to form MALO is described in the section "MALO algorithm". In this section, validation of the MALO is also carried out by verifying its performance on two unimodal and three multimodal benchmark functions. The section "Problem formulation" presents the problem formulation. Results and discussion are demonstrated in the section "Results and discussion". SLL, CISLL reduction, beam steering, and beam steering, along with nulls placing in specific directions, are analyzed. The results obtained are compared with that of the uniform arrays and with the previous works such as ACO, PSO, Taguchi, SMO, SBA, BBO, CSO, SOS, ALO, and MFO. The conclusion of the article is presented in the section "Conclusion", followed by a list of references.

MALO algorithm

The original ALO is modified by embedding CPSO to achieve a highly promising new hybrid algorithm. The robustness of this modified algorithm is verified by some standard benchmark functions. These are explained as follows.

Ant lion optimization

Mirjalili explains the ALO algorithm in depth in [43], so the detailed description is excluded here. The interactions of the antlions and ants, demonstrated in the algorithm include the following rules [43]:

1. Ants and antlions are the search agents, and the ants change their positions in random directions all over the search space.
2. Traps by antlions have impacts on the movements of the ants.
3. Traps are constructed in proportion to the fitness of antlions. Higher is the fitness, larger is the trap, and higher is the possibility of catching the ants.
4. In each iteration, an antlion and the most fitting antlion (elite) can capture each ant.
5. The random walk adaptively decreases, and the probability of achieving a solution increases by increasing the sliding of ants to the antlions.
6. An ant is hunted by the antlion when the ant is fit. This indicates that an anticipated solution is found.
7. Antlion relocates itself to the recent hunting position and constructs a conical hole to optimize the ability to catch another ant.

The stochastic movements of the ants, throughout the optimization process, are defined as:

$$R_i = [0,\ cums(2\gamma(t_1) - 1),\ \ldots,\ cums(2\gamma(t_n) - 1),\ \ldots,\ cums(2\gamma(t_M) - 1), \tag{1}$$

where *cums* computes the cumulative sum. Parameter t_n is the iteration number and M is the maximum number of iterations. $\gamma(t)$ is a random function described as follows:

$$\gamma(t) = \begin{cases} 1 & \text{if } rand > 0.5 \\ 0 & \text{if } rand \leq 0.5 \end{cases}, \tag{2}$$

where *rand* within the interval [0, 1] is a random number.

At every step, ants update their locations on the basis of equation (1). The random walks are modified based on normalization using the following equation:

$$R_{i\_norm} = \left(\frac{R_i - a_i}{b_i - a_i}\right) \times (d_i^t - c_i^t) + c_i^t, \tag{3}$$

where a_i is the minimum value and b_i is the maximum value of R_i. c_i^t is the minimum value and d_i^t is the maximum value of R_i at the tth iteration.

The hunting ability of an antlion is modelled through a roulette wheel operator. The roulette wheel chooses an antlion and assumes that ants are captured only by the selected antlion. Movement of the ants is affected by the position of the antlions. This is described by the following equations:

$$c_i = c^t + Al_j^t, \tag{4}$$

$$d_i = d^t + Al_j^t, \tag{5}$$

where c_i, d_i are the minimum and maximum values among all random walks for the ith ant. c^t, d^t indicate the minimum and maximum values among all random walks at the tth iteration. Al_j^t represents the position of the chosen jth antlion at the tth iteration.

When the ants slid into the trap, they try to escape from it. On the other hand, the sliding of ants towards the antlions is updated adaptively. Hence the values of c^t and d^t are updated using the following equations:

$$c^t = \frac{c^t}{10^w \times (t/n)}, \tag{6}$$

$$d^t = \frac{d^t}{10^w \times (t/n)}, \tag{7}$$

where w is a constant determined according to the present iteration. $w = 2$ at $t > 0.1T$, $w = 3$ at $t > 0.5T$, $w = 4$ at $t > 0.75T$, $w = 5$ at $t > 0.9T$, and $w = 6$ at $t > 0.95T$.

An antlion also updates the hunting position to increase its chance of capturing new ant. This phenomenon is demonstrated in the following equation:

$$Al_j^t = A_i^t \text{ if fitness of } A_i^t > \text{fitness of } Al_j^t, \tag{8}$$

where A_i^t = position of the ith ant at the tth iteration.

The best antlion in the optimization cycle is considered as the elite. During the optimization process, the elite affects the random walks of all ants. Hence the position of each ant is updated by the simultaneous effect of the elite and the roulette wheel. This is

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

expressed in equation (9) as:

$$A_i^t = \frac{X_{al}^t + X_e^t}{2}, \tag{9}$$

where X_{al}^t and X_e^t represent the random walk nearby the antlion selected through a roulette wheel and nearby the elite obtained within the optimization process at the tth iteration, respectively.

Modification of ALO

The advantage of ALO is that it avoids optimal local values. In the original ALO, CPSO is integrated to improve the optimization functionality to provide more accurate design parameters. The working procedure of CPSO is focused on the behavior of searching for foods by a group of birds or schooling of fish. The searching agents (birds/fishes) search for optimal solutions in adequate space and thus increase the probability of obtaining optimum values. Each agent moves randomly towards a global best position with its velocity and position vectors which are updated by equations (10) and (11) [45] as:

$$v_i = w \times v_i + c_1 \times rand_1 \times (p_i - x_i) + c_2 \times rand_2 \times (p_g - x_i), \tag{10}$$

$$x_i = x_i + v_i. \tag{11}$$

The variables of equation (10) are well-known in the domain of optimization through PSO and are described in [45]. The traditional PSO algorithm sometimes trapped prematurely in local optimums, which is a shortcoming. As described in [46], chaotic mappings help it to escape from the local optima. Chaotic maps constructed by mutating its initial state are apparently random deterministic and reproducible sequences. Several chaotic mappings, such as a logistic, tent, sinusoidal iterator, Gauss, etc. are adopted to enhance the global convergence of PSO. Here the CPSO considers the chaotic mutation operation based on logistic mapping to obtain the best optimal values. Logistic mapping shows vital usefulness in the improvement of the searching process. Mapping is employed in each iteration, and the position can be further updated using the formula as [46]:

$$x_i(d) = 4 \times x_{i-1}(d) \times (1 - x_{i-1}(d)), \tag{12}$$

where i is the current iteration. $d = 1, 2, \ldots, dim$ and dim is the dimension of searching range. The x_i is converted to the real positions as:

$$x_i(d) = lb(d) + x_i \times (ub(d) - lb(d)), \tag{13}$$

where lb is the lower bound and ub is the upper bound of the particles.

The optimization process of MALO involves the searching and updating strategy of both ALO and CPSO. In each iteration, the ALO explores the searching space first, by its search agents, the ants and the antlions. The positions of antlions define the optimizing parameters. After the exploration of searching space using ALO, the positions of antlions are updated and optimized employing CPSO. The positions of best-fitted antlions (elites) in each iteration are kept in memory. After the final iteration, the best elites are selected and the corresponding positions yield the global optimum parameters. The operational flow diagram of the MALO is shown in Fig. 1. The steps involved in the optimization process are as follows:

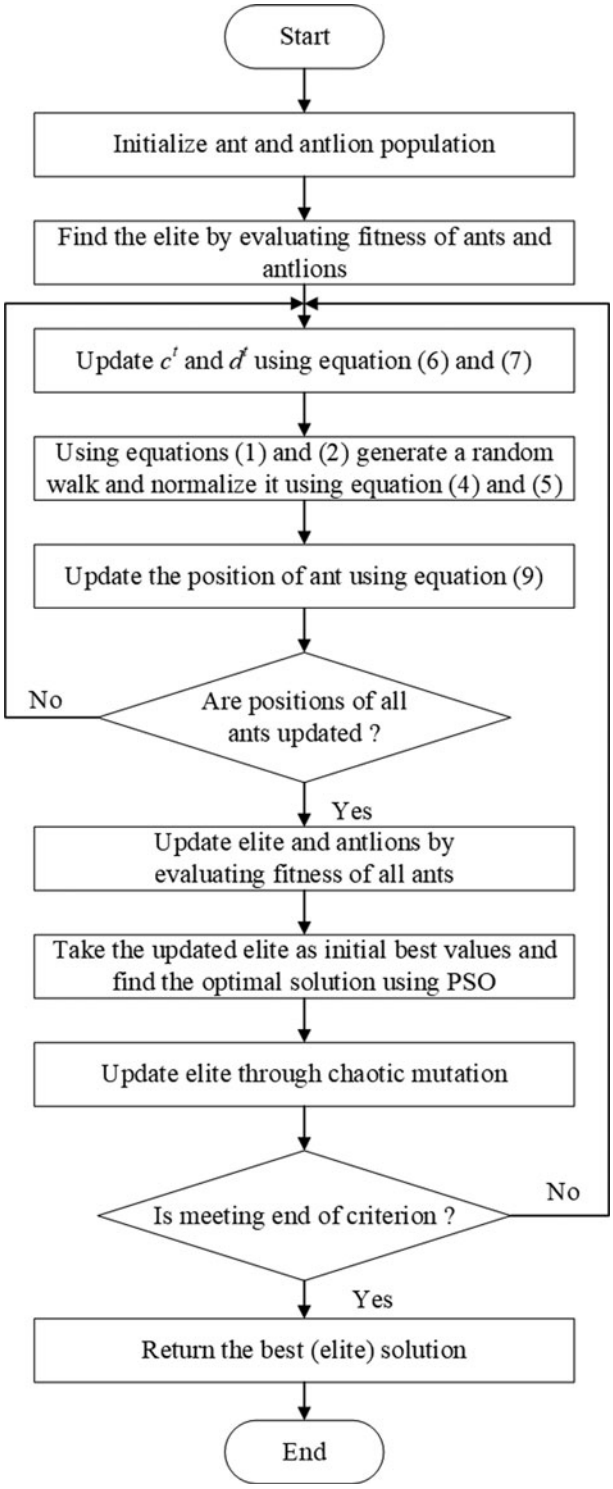

Fig. 1. Flowchart of the MALO.

Step 1: The positions of antlions and ants are initialized randomly.

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

Table 1. Benchmark functions [43].

Function	Expression	Dim (n)	Range	f_{min}
F_1	$\sum_{i=1}^{n} x_i^2$	30	$[-100, 100]^n$	0
F_2	$\sum_{i=1}^{n-1} [100(x_{i+1} - x_i^2)^2 + (x_i - 1)^2]$	30	$[-30, 30]^n$	0
F_3	$\sum_{i=1}^{n} -x_i \sin(\sqrt{\lvert x_i \rvert})$	30	$[-500, 500]^n$	−12 569.5
F_4	$\sum_{i=1}^{n} [x_i^2 - 10\cos(2\pi x_i) + 10]$	30	$[-5.12, 5.12]^n$	0
F_5	$0.1\{\sin^2(3\pi x_i) + \sum_{i=1}^{n-1}(x_i - 1)^2[1 + \sin^2(3\pi x_i + 1)] + (x_n - 1)^2[1 + s^2(2\pi x_n)]\} + \sum_{i=1}^{n} u(x_i, 5, 100, 4)$	30	$[-50, 50]^n$	0

Step 2: The fitnesses of the antlions are calculated and the elite is selected.
Step 3: An antlion is selected utilizing the roulette wheel. The random walks nearby the elite and the selected antlion are calculated.
Step 4: The positions of the ants are updated using equations (1)–(7) and (9).
Step 5: The positions of the antlions are updated using equation (8).
Step 6: The fitnesses of the updated antlions are calculated and compared with that of the elite. If the fitness of the updated antlion is better, then it becomes the elite.
Step 7: Better antlions are searched employing CPSO using equations (10) and (11).
Step 8: The elite is updated using equations (12) and (13).
Step 9: Steps 3–8 are repeated until the final iteration is reached.

Validation of MALO algorithm

The proposed algorithm is validated by verifying its performance on five benchmark functions (two unimodal (single-peak) functions and three multimodal (multi-peak) functions). Table 1 shows the benchmark functions taken from [43]. MALO is applied to optimize the above five functions by executing each for 30 runs with 30 dimensions and 1000 iterations per each term. Performance indices (mean and standard deviations) of the obtained minimums are computed for each function. The comparisons of the performance indices obtained using MALO and other algorithms, such as ALO, PSO, FPA, CS, FA, BA, and GA in [43] are shown in Table 2. From the results shown in Table 2, it is worth noting that the performance of MALO is superior than other algorithms.

The characteristics of the benchmark functions and their resultant convergence curves are plotted in Figs 2–6. In [43], it is observed that the performance of ALO is better than various other optimization algorithms. The performance is evaluated by comparing the convergence characteristics of ALO with that of the other algorithms. As MALO is the modification of ALO, the convergence characteristics of this are compared with that of ALO and PSO instead of all other algorithms. The comparison

Table 2. Mean (μ) and standard deviation (σ) of the benchmark functions and their comparison with other algorithms in [43].

	GA		BA		FA		FPA	
F	μ	σ	μ	σ	μ	σ	μ	σ
F_1	0.118842	0.125606	0.773622	0.528134	0.039615	0.01449	1.06346×10^{-7}	1.27×10^{-7}
F_2	0.13902	0.121161	0.334077	0.300037	0.049273	0.019409	0.781200043	0.000176
F_3	−2091.64	2.47235	−1065.88	858.498	−1245.59	353.2667	−1842.42621	50.42824
F_4	0.659278	0.815751	1.233748	0.686447	0.263458	0.182824	0.273294621	0.068583
F_5	1.29×10^{-1}	0.068851	0.386631	0.121986	0.00213	0.001238	3.67×10^{-6}	3.51×10^{-6}
	CS		PSO		ALO		MALO	
F	μ	σ	μ	σ	μ	σ	μ	σ
F_1	6.50×10^{-3}	2.05×10^{-4}	2.70×10^{-9}	1.00×10^{-9}	2.59×10^{-10}	1.65×10^{-10}	**2.6488×10^{-11}**	**1.056×10^{-11}**
F_2	2.12×10^{-1}	2.14×10^{-2}	0.123401	0.216251	0.34677239	0.109584	**0.0983125**	**0.030564**
F_3	−2094.91	0.007616	−1367.01	146.4089	−1606.27643	314.4302	**−3577.7893**	**425.6523**
F_4	0.127328	0.002655	0.278588	0.218991	7.71411×10^{-6}	8.45×10^{-6}	**6.2896×10^{-10}**	**4.8235×10^{-10}**
F_5	4.88×10^{-6}	6.09×10^{-7}	1.35×10^{-7}	2.88×10^{-8}	2.00222×10^{-11}	1.13×10^{-11}	**2.3746×10^{-13}**	**1.0095×10^{-13}**

The values in bold indicate that they belong to our method.

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

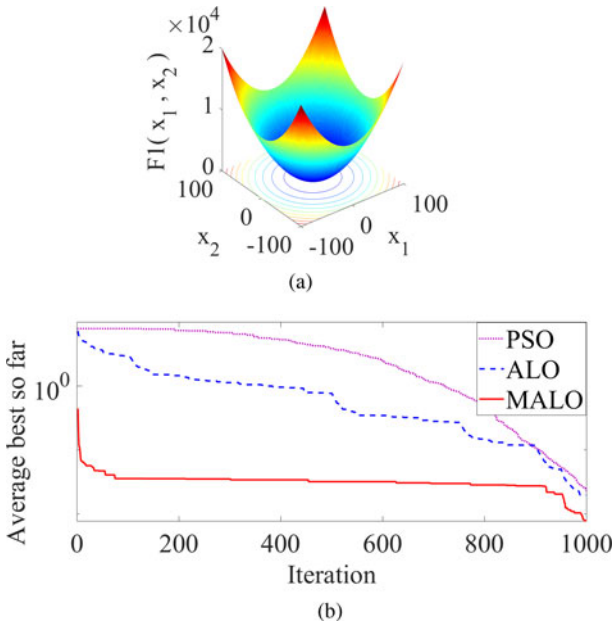

Fig. 2. (a) Function F_1. (b) Convergence curve of function F_1.

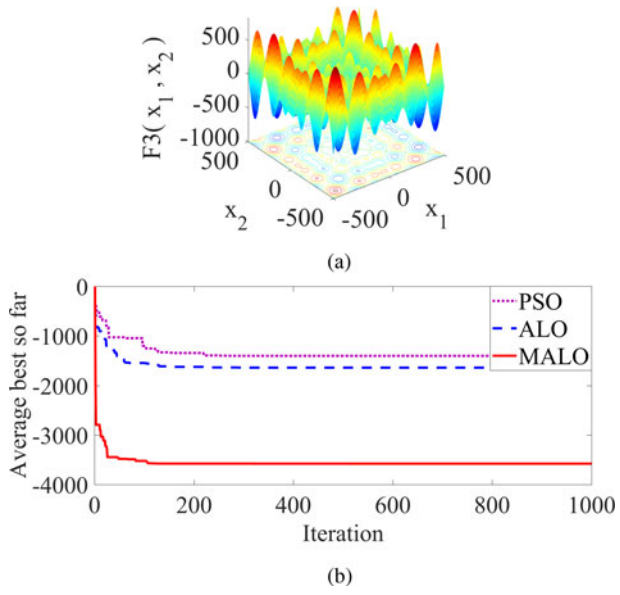

Fig. 4. (a) Function F_3. (b) Convergence curve of function F_3.

with other algorithms is deliberately excluded, as it is already available in [43]. The characteristics of the two unimodal test functions F_1 and F_2 are shown in Figs 2 and 3, respectively. These figures depict the benefits of high exploitation behavior of MALO. High exploitation enables the MALO algorithm to converge rapidly towards the optimum. Figures 4–6 show the characteristics of the multimodal functions F_3, F_4, and F_5, respectively. From these characteristics, it is clearly observed that the exploration ability of MALO is also high. This high level of exploring ability enables MALO to explore the desirable search domain. Due to the high level of exploration and exploitation ability of MALO the local optima or premature convergence are avoided and the global optima are attempted. Attempting the global optima by MALO can be clearly observed from the convergence curve of each benchmark function. The curve illustrates that the MALO exhibits superior convergence functionality, though the simulation time is a little longer than that of PSO. The simulation time for ALO is also longer than PSO and is closer to MALO. The time in case of MALO is longer, because of the hybridization of two algorithms. Embedding one algorithm (CPSO) to another (ALO) makes the hybrid algorithm (MALO) to some extent complex as compared to the individual one. That complexity in the new modified algorithm makes the simulation time longer. The simulation time of the algorithms PSO, ALO, and MALO for 1000 iterations is presented in Table 3. In spite of the long run time, the higher convergence functionality and ability to avoid local optima and attempting global optima makes use of the

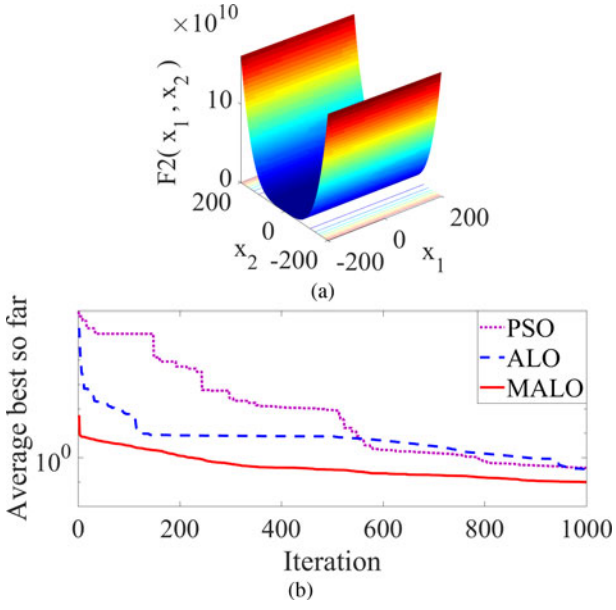

Fig. 3. (a) Function F_2. (b) Convergence curve of function F_2.

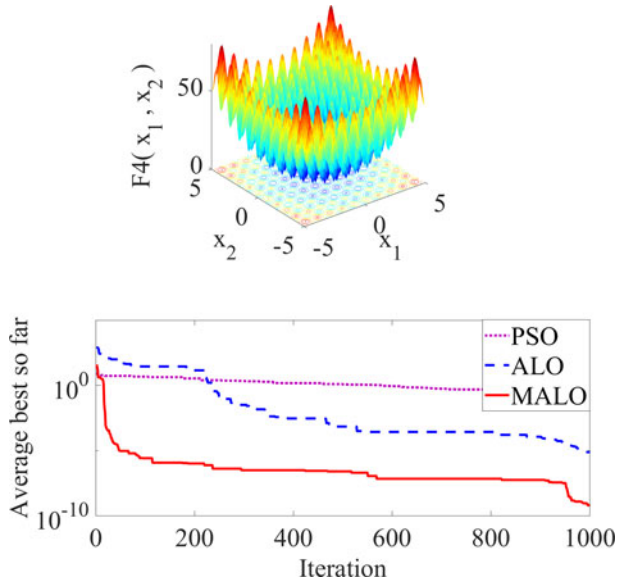

Fig. 5. (a) Function F_4. (b) Convergence curve of function F_4.

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

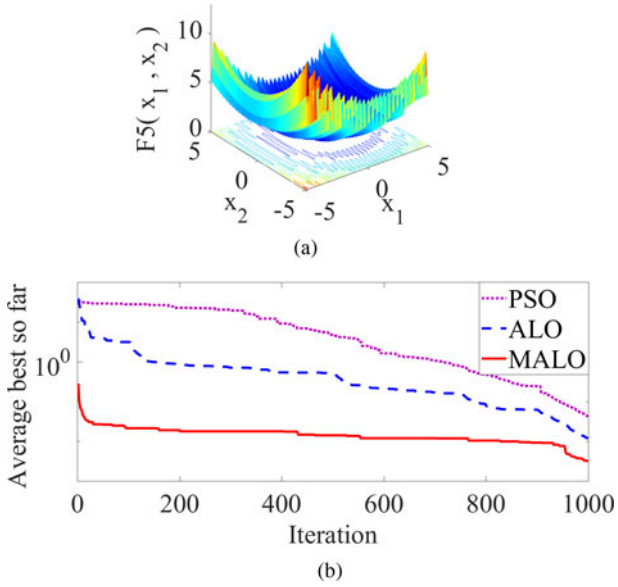

Fig. 6. (a) Function F_5. (b) Convergence curve of function F_5.

proposed MALO algorithm for antenna array synthesis and beamforming in smart antenna technology.

Problem formulation

Various linear antenna arrays based on Chebyshev distribution, as shown in Fig. 7 are considered in this work. In the figure, $2N$ isotropic elements are located along the x-axis same as the consideration of other authors. Out of $2N$, N elements are located on the right side of the origin and named as 1, 2, …, N. Similarly, other N elements are placed to the left-hand side and are assigned as $1'$, $2'$, …, N'. The elements 1 and $1'$ are placed at $x = \lambda/4$ on both sides of the origin. Array factor for the geometry is given by [1]:

$$AF(\phi) = 2\sum_{n=1}^{N} I_n \cos\left(kx_n \cos\left(\phi\right) + \varphi_n\right), \tag{14}$$

where the amplitude excitation of the nth element is I_n. k is wave number and is equal to $2\pi/\lambda$. Parameters x_n and φ_n represents the position and phase excitation of the nth element, respectively.

The array factor of the Chebyshev distributed linear array is the summation of cosine terms with symmetric amplitude excitations. Each cosine term is an integer multiple of a fundamental frequency and can be rewritten as a series of cosine functions

Table 3. Simulation time (min) of the algorithms for 1000 iterations: PSO, ALO, and MALO.

Function	PSO	ALO	MALO
F_1	0.01326	0.8259	**0.8381**
F_2	0.01641	0.8242	**0.8378**
F_3	0.01417	0.9436	**0.9471**
F_4	0.01538	0.9092	**0.9392**
F_5	0.03845	0.9558	**0.9673**

The values in bold indicate that they belong to our method.

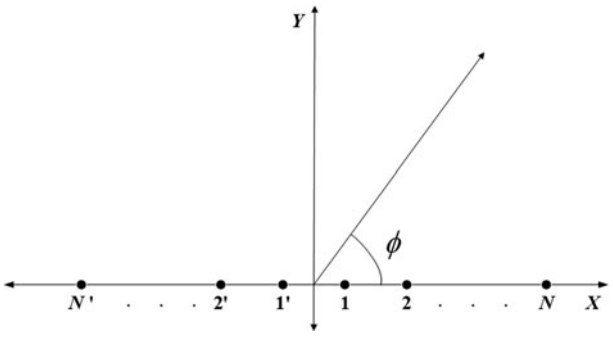

Fig. 7. $2N$ elements linear array geometry.

correlated to the Chebyshev polynomials. Chebyshev polynomial is denoted as $T_m(z)$ and is formulated as [1]:

$$T_m(z) = 2zT_{m-1}(z) - T_{m-2}(z), \tag{15}$$

where m is the order of the Chebyshev polynomial, which is one less than a number of elements. $z = \cos(u)$ and $u = (\pi x_n/\lambda)\cos\phi$.

Since the excitation distribution of a Chebyshev array is symmetrical, the amplitudes of the left-hand side elements are the same as that of the right-hand side elements. Hence amplitudes of only the right-hand side elements (i.e. 1, 2, …, N) are shown in the respective tables of different design examples. MALO is implemented in the optimization of four design examples of linear arrays to optimize input parameters like amplitudes, positions, and phases of the elements. The design specifications, such as the number of elements, the input parameters, and the SLL region of each design examples are kept the same as the structures in the literature. These structures optimized with other optimization algorithms are also available in the literature. These literatures are appropriate to compare the results related to MALO with that of other optimization techniques.

In order to have better insight into our problems, a parameter mapping between the hybrid algorithm and the array is presented in Table 4. Each array example is demonstrated using a different set of parameters. The link between these parameters and the algorithm can be easily understood from Table 4. The proposed algorithm is executed taking into account the number of search agent as 50 and the total iterations as 1000. These numbers are helpful to achieve better parameters. The optimized parameters are achieved through the appropriate positions of the antlions. The updated elite positions are found following the steps of the optimization process presented in the subsection "Modification of ALO".

Example-I: amplitude optimization

The use of MALO for the synthesis of two linear arrays (i.e. Example-Ia: $2N = 10$ elements array and Example-Ib: $2N = 16$ elements array) is considered here. The objective of this example is to obtain a Chebyshev array pattern as well as the peak SLL suppression. Peak SLL in a particular region can be minimized by considering the objective function formulated as:

$$FObj = \min(\max(20\log|AF(\phi)|)) \tag{16}$$

where $\max(20\log|AF(\phi)|)$ provides the peak SLL. Considering equation (16) as the objective function for MALO, the current

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

Table 4. Mapping between the MALO and the array in the optimization process.

Term related to MALO algorithm	Mapped term related to the antenna array
Antlions	Parameters (spacing between the elements, amplitudes, and phases) to be optimized.
Dimensions of the antlions or ants	Number of parameters (element number, amplitude number, etc.)
Trap (conical hole) and its nearby space	Searching space (rang between the lower and upper limits)
Position of antlions	Value of parameters
Elite (best-fitted antlions)	Best solutions so far
Updating the position of the antlions	Keeping the best solution found so far
Elite of the final iteration	Optimum values of the parameters
Random walks through the roulette wheel	Selection of a set of parameters
Minimum fitness value	Best combination of AF, minimum SLL or CISLL, and considerable FNBW.
Sliding of ants to the antlions	Converging towards the best solution
An ant is hunted	An expected solution is found.
Antlion relocates itself to the recent hunting position and constructs a conical hole	Choosing a new combination of the optimizing parameters

amplitude excitations of the antenna elements are optimized. The search region for the current amplitudes is spread from 0 to 1. Putting the optimal values found through MALO in equation (14), $AF(\phi)$ for the said linear arrays with suppressed peak SLL is obtained. In this example, the positions of each element and their phases are kept fixed (i.e. $x_n = 0.5\lambda$ and $\varphi_n = 0^0$). The SLLs are suppressed in the regions, $\phi = [0^0, 76^0]$ and $\phi = [104^0, 180^0]$.

Example-II: position optimization

In this example, MALO is implemented for the optimization of the positions of the elements from the origin, as shown in Fig. 7. A 2 N =10 elements non-uniformly spaced linear array is considered as Example-IIa. It is optimized to take the same objective function in equation (16). For a uniform array of 0.5λ spaced elements the total length is 4.5λ. Therefore, the last elements' positions are fixed at 2.25λ on each side of the y-axis. The minimum spacing between two consecutive elements is taken as 0.25λ. Similar to Example-I, peak SLL suppressed $AF(\phi)$ is obtained by putting the optimum position values found through MALO in equation (14). Here the amplitudes of the elements and their phases are kept constant (i.e. $I_n = 1$ and $\varphi_n = 0^0$). The SLL regions for this example are defined by $\phi = [0^0, 76^0]$ and $\phi = [104^0, 180^0]$.

In a smart antenna-based communication system, there are many applications that need minimization of CISLL adjacent to the main lobe. This can also be achieved by the elements position optimization. For this, the objective function is taken as [13]:

$$FObj = \min\left[\propto_1 \max\{20\log|AF(\phi_{AS})|\} + \propto_2 \max\{20\log|AF(\phi_{NS})|\}\right] \tag{17}$$

where ϕ_{AS} are the SLL regions the same as Example-IIa defined by $\{[0^\circ,\ 76^\circ]$ and $[104^\circ,\ 180^\circ]\}$. ϕ_{NS} represents the CISLL region, $\{[69^\circ,\ 76^\circ]$ and $[104^\circ,\ 111^\circ]\}$. $\propto_1$ and $\propto_2$ are constants with values 1 and 2, respectively.

Keeping the same conventions, the 10-elements linear array is again optimized to take the objective function as in equation (17) to realize the CISLL suppressed $AF(\phi)$. This is referred to as Example-IIb.

Example-III: phase optimization for beam steering

Nowadays, wireless and mobile technologies are widely applied in various wireless monitoring applications. Some of the wireless monitoring applications are water level and its quality monitoring, health monitoring of bridge pillar, building wall structures [47, 48]. These wireless monitoring applications include a set of omnidirectional and directional antenna systems. The omnidirectional antenna transmits information to the distant terminal units. The directional antenna system collects the data from the antennas embedded in different places on the concrete structures. The antenna system adopts the beam-steering process for the data collection from the remote embedded antennas. Beam steering is the procedure of guiding an antenna array's main beam towards a specific direction.

Beam-steering application by the implementation of MALO in the linear array is demonstrated in this example. The simplest method of achieving beam steering for a linear array is through the optimization of phases of the elements. The positions of the elements and their amplitudes are kept fixed (i.e. $x_n = 0.5\lambda$ and $I_n = 1$). Keeping the phase of the first element fixed at 0° as a reference, optimal phases for rest of the elements are found by MALO and considering the $AF(\phi)$ as [13]:

$$AF(\phi) = \sum_{n=2}^{2N} \exp(j[n\pi\cos(\phi) + \varphi_n]) + 1. \tag{18}$$

The SLL regions are defined by $\phi = [0^\circ,\ (\phi_s - (\Delta\phi_s/2))^0]$ and $\phi = [(\phi_s + (\Delta\phi_s/2))^0,\ 180^\circ]$. ϕ_s represents the steering angle and it lies in the band $\Delta\phi_s$. In this example, phases of a 20-elements linear array are optimized for steering angle $\phi_s = 45^\circ$ and the band of steering angle $\Delta\phi_s = 14^\circ$. The $AF(\phi)$, as in equation (18) with reduced SLL is obtained by putting the optimum phase values found applying MALO.

Example-IV: beam steering and null positioning by simultaneous optimization of amplitude and phase

In this example, MALO is employed for simultaneous optimization of amplitude and phase for smart antenna technology. The smart antenna application involves steering the main beam

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

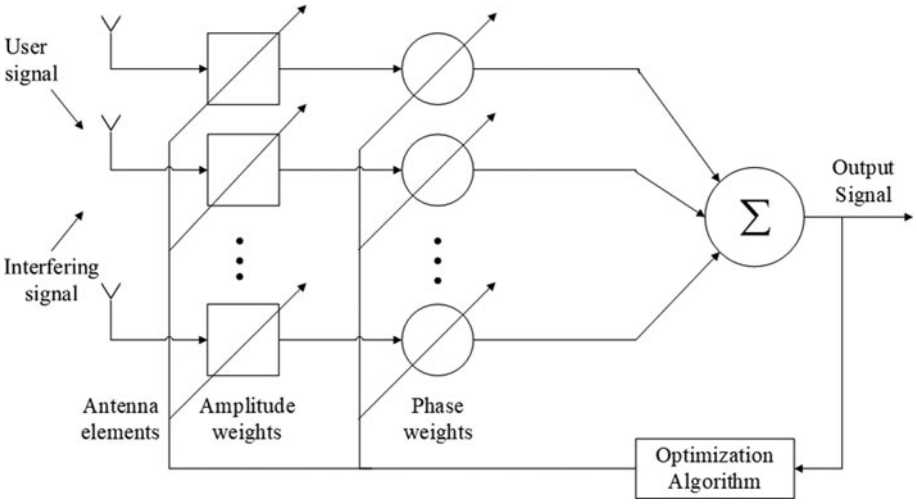

Fig. 8. Adaptive antenna array geometry.

along with null positioning in the specific directions. The steering angle is characterized as the direction of the SOI angle, and the angles of nulls are characterized as SNOI angles. The beamforming here is achieved by an adaptive process. Figure 8 shows the block diagram of an adaptive array. Elements of antenna array receive the combined signal of subscriber's signal at the SOI angle and the interfering signals at SNOI angles. The combined signal is then multiplied with a complex-valued weight having the elements amplitude weights and the phase weights. These amplitude and phase weights are optimized in order to obtain the optimum output signal. Considering the self-developed MATLAB code, the main beam of the radiation pattern is directed at any specific SOI angle along with the placement of one or more nulls at any SNOI angles. In this section, a 20-elements array is considered to steer the main beam at the SOI angle $\phi_s = 45^\circ$ and nulls at SNOI angles $\phi_{nl1} = 25^\circ$ and $\phi_{nl2} = 65^\circ$. The $AF(\phi)$ with reduced SLL is obtained by the optimum amplitude and phase excitations values found applying MALO. $AF(\phi)$ for this case is formulated as [1]:

$$AF(\phi) = \sum_{n=1}^{N} w_n \exp(j[(n-1)kx_n \sin(\phi) + \varphi_n]), \qquad (19)$$

where w_n is the complex weight of the nth element expressed as $a_n \exp(jb_n)$. The amplitude weight of the nth element is a_n, and b_n is its phase weight.

Results and discussion

MALO is implemented in four examples of linear arrays to optimize element amplitudes, positions, and phases of the elements either in single or in combined form. The optimized parameters provide various outcomes such as suppression of peak SLL and CISLL, beam steering, and beam steering, along with null positioning. The obtained outcomes are compared with that of the conventional uniform arrays and with the previous works, such as ACO, PSO, Taguchi, SMO, SBA, BBO, CSO, SOS, ALO, and MFO. The comparisons are made considering assumptions like isotropic elements, the same parameter specifications, and without a constant first null beamwidth (FNBW) as in the previous works.

The optimal amplitudes of current for design Examples-Ia and Ib are shown in Tables 5 and 6, respectively. Figure 9(a) presents the optimized radiation pattern of the design Example-Ia. It shows that all the algorithms provide nearly the same FNBWs. The FNBWs are greater than that of the uniform array. In spite of this, the peak SLL can be better suppressed by the use of optimization algorithms. The peak SLL of −27.6 dB is achieved by using MALO. The peak SLL of uniform array, PSO [13], Taguchi [24], BBO [25], SOS [28], ALO [21], and MFO [31] is 14.63, 2.9783, 2.73, 2.39, 2.32, and 1.52 dB higher than that compared to MALO, respectively. MALO provides the highest peak SLL suppression, taking into account the same parameter

Table 5. Optimal amplitudes of current for Example-Ia.

	Element amplitude				
Optimization	1	2	3	4	5
PSO [13]	1.0000	0.9010	0.7255	0.5120	0.4088
Taguchi [24]	1.0000	0.8999	0.7228	0.5077	0.3994
BBO [25]	1.0000	0.8988	0.7189	0.5025	0.3862
SOS [28]	1.0000	0.8985	0.7189	0.5017	0.3856
ALO [21]	1.0000	0.8959	0.6957	0.4935	0.2966
MFO [31]	1.0007	0.8962	0.6966	0.4935	0.2965
Proposed MALO	**1.0000**	**0.8909**	**0.6822**	**0.4826**	**0.2878**

The values in bold indicate that they belong to our method.

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

Table 6. Optimal amplitudes of current for Example-Ib.

	Element amplitude							
Optimization	1	2	3	4	5	6	7	8
PSO [13]	1.0000	0.9521	0.8605	0.7372	0.5940	0.4465	0.3079	0.2724
Taguchi [24]	1.000	0.9500	0.8575	0.7317	0.5861	0.4381	0.2988	0.2552
BBO [25]	1.000	0.9402	0.8487	0.7104	0.5596	0.4115	0.2697	0.2035
SOS [28]	1.000	0.9466	0.8475	0.7137	0.5624	0.4094	0.2697	0.2088
ALO [21]	1.0000	0.9344	0.8521	0.7044	0.6000	0.4000	0.3003	0.2002
MFO [31]	1.0004	0.9340	0.8129	0.6534	0.4857	0.3197	0.1932	0.1002
Proposed MALO	**1.0000**	**0.9286**	**0.8445**	**0.6935**	**0.5989**	**0.3912**	**0.2985**	**0.1956**

The values in bold indicate that they belong to our method.

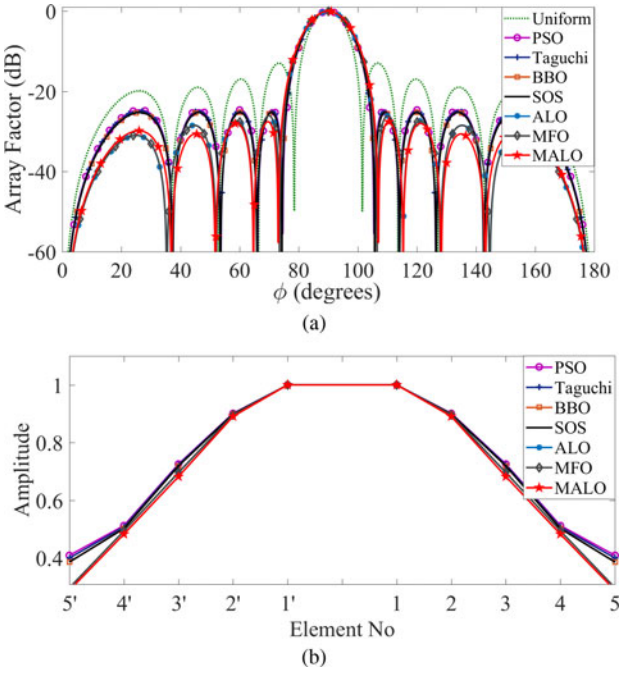

Fig. 9. (a) Radiation pattern of Example-Ia. (b) Amplitude distribution of Example-Ia.

specifications such as a number of elements, amplitude excitation, and SLL region. This result is summarized in Table 7. The amplitude distributions of the array elements of the design Example-Ia are shown in Fig. 9(b). From this figure, it is noted that the amplitudes of the current decrease from the central element to the outermost element. Hence power dividers can be easily used for such amplitude distribution. It also provides the information that by using MALO, peak SLL is better suppressed even with smaller amplitudes. Similarly, the radiation pattern and the

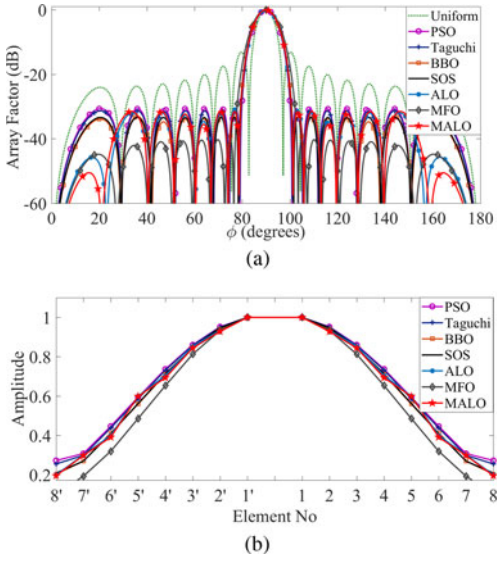

Fig. 10. (a) Radiation pattern of Example-Ib. (b) Amplitude distribution of Example-Ib.

corresponding amplitude distribution for design Example-Ib are illustrated in Figs 10(a) and 10(b), respectively. From Fig. 10(a), it can be noticed that the FNBW of MFO is greater than that of other algorithms. Here also the optimization algorithms except MFO provide nearly the same FNBW. Peak SLL achieved using MALO for this case is −31.6 dB. Uniform array, PSO [13], Taguchi [24], and ALO [21] provide peak SLLs of 18.45, 0.97, 0.39, and 0.75 dB higher than that compared to MALO, respectively. The peak SLL of MALO is 1.46, 8.65 and 1.79 dB higher than that of BBO [25], MFO [31], and SOS [28], respectively. The MFO displays a better SLL peak suppression, but it reveals a higher half-power beamwidth (HPBW) than other methods. Higher HPBW is not desirable in smart antenna beam-steering

Table 7. Peak SLL (dB) of Example-I.

Design	Uniform array	PSO [13]	Taguchi [24]	BBO [25]	SOS [28]	ALO [21]	MFO [31]	Proposed MALO
Example-Ia	−12.97	−24.6217	−24.87	−25.21	−25.28	−26.08	−26.07	**−27.6**
Example-Ib	−13.15	−30.63	−31.21	−33.06	−33.39	−30.85	−40.25	**−31.6**

The values in bold indicate that they belong to our method.

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

Table 8. Optimal positions of elements for Example-IIa.

	Element position ± $x_n(\lambda)$				
Optimization	1	2	3	4	5
Uniform array	0.25	0.75	1.25	1.75	2.25
ACO [10]	0.25	0.55	1.05	1.55	2.15
PSO [13]	0.2605	0.5105	1.0186	1.4694	2.1407
Taguchi [24]	0.2142	0.5989	1.0597	1.5861	2.25
SMO [11]	0.236	0.528	1.007	1.471	2.126
SBA [12]	0.230	0.515	0.981	1.460	2.120
CSO [26]	0.1516	0.4115	0.7899	1.1048	1.6843
BBO [25]	0.21451	0.60006	1.0610	1.5870	2.25
ALO [21]	0.1259	0.3751	0.7515	0.9994	1.5652
Proposed MALO	**0.1281**	**0.3627**	**0.7430**	**0.9841**	**1.5597**

The values in bold indicate that they belong to our method.

Table 9. Optimal positions of elements for Example-IIb.

	Element position ± $x_n(\lambda)$				
Optimization	1	2	3	4	5
Uniform array	0.25	0.75	1.25	1.75	2.25
PSO [13]	0.1685	0.5461	0.9364	1.5107	2.25
Taguchi [24]	0.1642	0.5509	0.9362	1.5199	2.25
BBO [25]	0.15085	0.5567	0.93079	1.5157	2.25
ALO [21]	0.1792	0.5407	0.9452	1.5097	2.25
Proposed MALO	**0.1782**	**0.5245**	**0.9368**	**1.5057**	**2.25**

The values in bold indicate that they belong to our method.

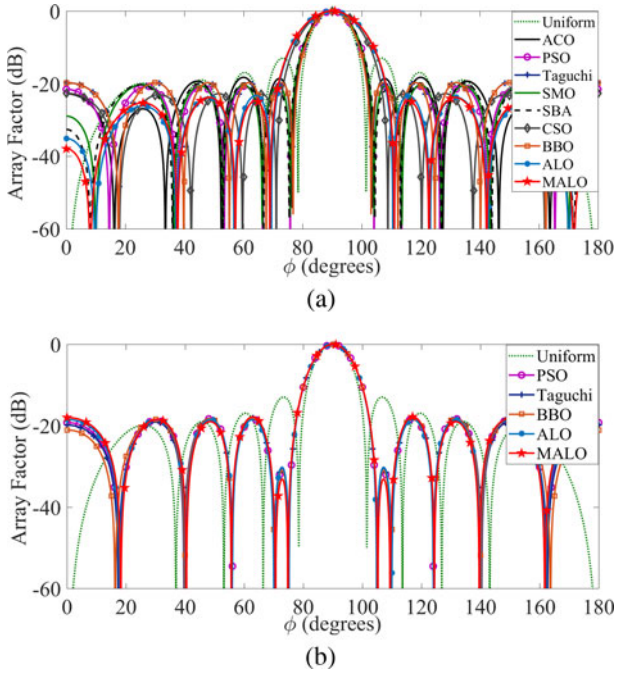

Fig. 11. (a) Radiation pattern of Example-IIa. (b) Radiation pattern of Example-IIb.

applications. This is because a higher HPBW represents a wider main beam and hence the lower directive radiation pattern. In our case, the MALO provides the highest peak SLL suppression, considering the equal HPBW. The findings of the design Example-Ib are also outlined in Table 7.

The optimum position values obtained for design Examples-II are presented in Tables 8 and 9. Table 8 depicts the optimal positions of Example-IIa. Figure 11(a) illustrates the radiation pattern of the Example-IIa, in which reduction of peak SLL is presented. From Fig. 11(a), it can be observed that the CSO, ALO, and MALO provide higher FNBW. BBO provides the smallest FNBW. In spite of this, all the algorithms offer peak SLL suppression. The SLL suppression using all algorithms by optimizing position is lower than that by amplitude optimization. For this example, the peak SLL achieved by MALO is −24.69 dB which is 11.72, 2.03, 3.97, 6.61, 4.44, 1.96, 1.80, 4.99, and 1.40 dB lower than that compared to uniform array, ACO [13], PSO [13], Taguchi [24], SMO [14], SBA [12], CSO [26], BBO [25], and ALO [21], respectively. It implies that the MALO among other algorithms has the best peak SLL suppression. The radiation pattern of design Example-IIb is shown in Fig. 11(b). The figure shows that CISLL suppression, a very demanding feature can be achieved by position optimization. Here, uniform array, the array optimized employing PSO [13], Taguchi [24], BBO [25],

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

Table 10. Peak SLL (dB), and CISLL (dB) for Example-II.

Design	Uniform array	ACO [10]	PSO [13]	Taguchi [24]	SMO [11]	SBA [12]	CSO [26]	BBO [25]	ALO [21]	Proposed MALO
Example-IIa (Peak SLL)	−12.97	−22.66	−20.72	−18.08	−20.25	−22.73	−22.89	−19.70	−23.29	**−24.69**
Example-IIb (Peak SLL)	−12.97	–	−18.31	−18.08	–	–	–	−18.14	−18.5	**−18.01**
Example-IIb (CISLL)	−12.97	–	−31.0	−30.44	–	–	–	−31.02	−30.24	**−33.13**

The values in bold indicate that they belong to our method.

Table 11. Optimal phases of elements for Example-III.

Optimization	Element phases (φ_n)
PSO [13]	0, 253, 134, 6.5, 237.9, 109.7, 343.3, 217.1, 91.3, 324.8, 197.8, 70.9, 304.9, 179.0, 51.8, 284.6, 156.1, 28.8, 268.8, 165.9
ALO [21]	0, 237.6558, 100.6945, 324.6030, 189.2521, 54.7329, 281.2010, 148.3234, 6.0491, 244.2962, 113.2517, 342.8036, 213.1424, 83.7345, 314.4018, 185.1796, 57.0217, 292.8150, 181.4052, 90.1626
Proposed MALO	**0, 233.1453, 101.4513, 331.6756, 205.0493, 80.2251, 312.6770, 188.2829, 61.5675, 292.4402, 161.1995, 30.1116, 263.7237, 137.6325, 11.7202, 247.9531, 118.6177, 348.3104, 221.6622, 93.3327**

The values in bold indicate that they belong to our method.

ALO [21], and MALO reveals the peak SLL of −12.97, −18.31, −18.08, −18.14, −18.5, and −18.01 dB, respectively. The corresponding CISLL value, utilizing MALO is −33.13 dB. This CISLL is 20.16, 2.13, 2.89, 2.69, and 2.11 dB lesser than that of uniform array, PSO [13], ALO [21], Taguchi [24], and BBO [25], respectively. The MALO provides the highest CISLL suppression among other algorithms. The FNBWs offered by the algorithms are nearly identical. ACO [13], SMO [14], SBA [12], and CSO [26] are applied only for the position optimization of Example-IIa (i.e. peak SLL suppression) and not for Example-IIb (i.e. CISLL suppression). Table 10 shows the summarized results of Examples-IIa and IIb.

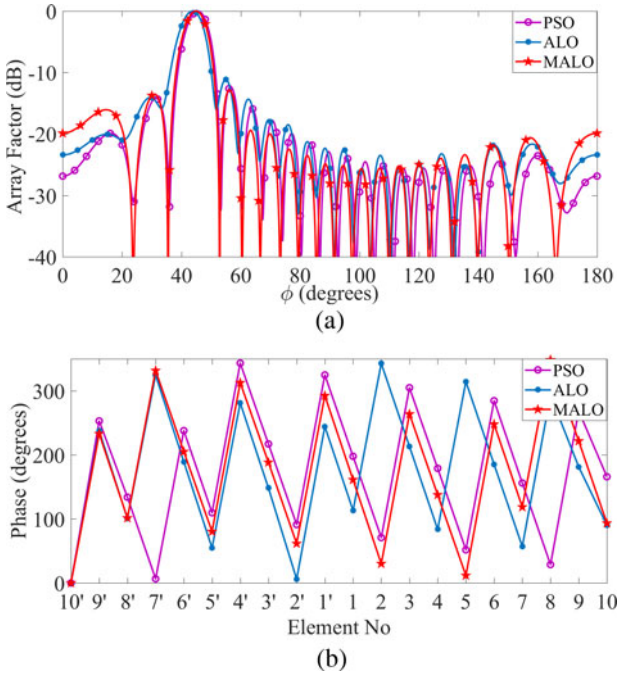

Fig. 12. (a) Radiation pattern of Example-III. (b) Phase distribution of Example-III.

Table 12. Peak SLL (dB) of Example-III.

Design	PSO [13]	ALO [21]	Proposed MALO
Example-III	−12.34	−11.14	**−12.84**

The values in bold indicate that they belong to our method.

Example-III illustrates primarily the steering of the main beam along with SLL suppression by the optimization of only the excitation phases of the antenna elements. The optimal phases for beam steering in design Example-III are presented in Table 11. Figure 12 displays the respective radiation pattern and the plots of phase distribution. The radiation pattern using MALO in Fig. 12(a) is nearly the same as that of PSO. But the null depths in case of MALO are more than that of PSO. The main beam steering by MALO towards the desired angle 45° is superior than ALO and PSO. This can be explicitly observed with the expansion of the radiation pattern. The peak SLL obtained here is −12.84 dB. Although the peak SLL suppression here is not much better, still it is 0.5 and 1.7 dB lower than PSO [13] and ALO [21], respectively. This analysis is summarized in Table 12.

In the literature, the authors have focused on either beam steering or null positioning applications. One of the most important features of simultaneous beam steering and null positioning for smart antenna application is proposed in Example-IV. The coinciding of beam steering and null positioning is achieved by optimizing the element amplitude weights and phase weights. Optimal normalized amplitude weights and phase weights of the elements are obtained by the application of MALO and are presented in Table 13. The normalized amplitude weight of an element is computed by dividing its amplitude weight by the maximum weight among all the weights of the elements. The corresponding radiation pattern for the angles $\phi = [-90^\circ, 90^\circ]$ is depicted in Fig. 13(a). This demonstrates that the main beam of

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

Table 13. Optimal normalized amplitude and phase weights of elements for Example-IV.

Optimized parameter	Elements
Amplitude	0.9376, 0.8741, 0.8504, 0.8497, 0.8872, 0.9257, 0.9048, 1.0000, 0.8681, 0.8011, 0.8618, 0.8665, 0.9383, 0.9603, 0.8713, 0.9103, 0.8283, 0.8697, 0.8906, 0.9571
Phase	4.5335, −122.9518, 105.4615, −24.3896, −151.0186, 84.5545, −43.4081, −167.4414, 66.4572, −63.7759, 168.1544, 37.2871, −89.6465, 144.6510, 18.7167, −105.1231, 125.8615, −4.8675, −131.1866, 100.0791

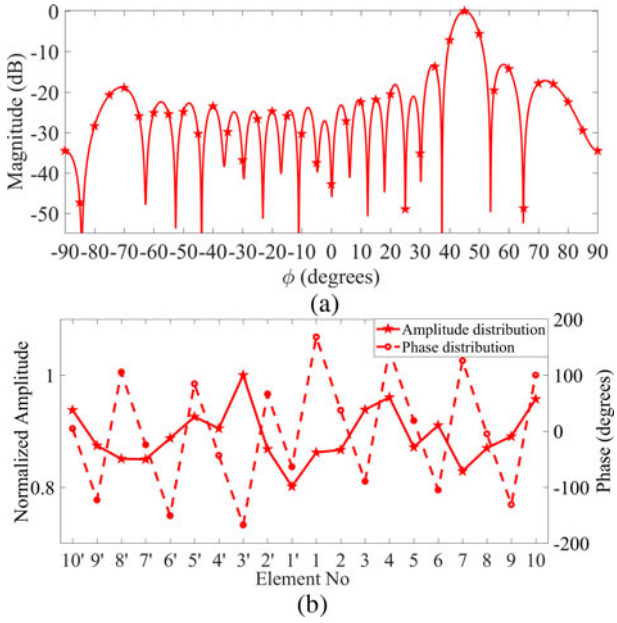

Fig. 13. (a) Radiation pattern of Example-IV. (b) Normalized amplitude and Phase distribution of design Example-IV.

the radiation pattern is directed towards the SOI angle $\phi_s = 45^\circ$ and nulls at SNOI angles $\phi_{nl1} = 25^\circ$ and $\phi_{nl2} = 65^\circ$. Null depths of −48.93 and −52.29 dB are achieved for the nulls at 25° and 65°, respectively. The peak SLL achieved for this example is −13.184 dB. Both the normalized amplitude and phase distribution plots are presented in Fig. 13(b). This example is a new approach in the field of antenna technology. Thus, no comparison with other algorithms could be established.

Conclusion

This article proposed the hybrid algorithm MALO successfully. MALO is validated by verifying its performance on five popular benchmark functions. The effectiveness of MALO is compared with other widely used optimization algorithms, such as ALO, PSO, CS, FPA, FA, BA, and GA. Considering the results, it is concluded that MALO being a potential optimization algorithm can be effectively used to solve many engineering optimization problems.

The amplitudes, positions, and phases of Chebyshev-based linear array are successfully optimized, employing MALO. The proposed algorithm is applied to optimize single or multiple design parameters at a time for obtaining optimal arrays. In all the examples, the MALO provides a considerable reduction in peak SLL. Higher peak SLL suppression is achieved in Example-I when the amplitude optimization is considered. Example-II illustrates the suppression of CISLL for specific applications by optimizing the positions between the elements. Beam steering meant for mobile and other wireless applications like environmental monitoring system are successfully presented in Example-III. The amplitude and phase excitations of the antenna elements are simultaneously optimized to achieve the beam steering in the desired direction. The design Example-IV presents a new direction of effort in the field of smart antenna technology. This design example discusses both beam steering and null positioning in contrast to only beam steering or null positioning by other researchers. The beam steering is carried out along the SOI angle $\phi_s = 45^\circ$ and nulls at SNOI angles $\phi_{nl1} = 25^\circ$ and $\phi_{nl2} = 65^\circ$. The results obtained by implementing MALO are compared with that of conventional uniform arrays and the arrays optimized using algorithms, such as ACO, PSO, Taguchi, SMO, SBA, BBO, CSO, SOS, ALO, and MFO. The comparisons depict that the MALO-based array designs provide better performance.

MALO is employed in the optimization of four different design examples to motivate the antenna design community. In addition, MALO may be used to optimize certain antenna geometries like microstrip, conformal, and fractal. Configurations like planar, circular, elliptical, or hexagonal arrays may also be considered for optimization using MALO. This is because, the provided results such as SLL minimization, beam steering and null positioning along specific directions are also valid for the above configurations. It may be a useful extended approach to evaluate the effectiveness of MALO by optimizing specifically the array structures generating shaped beam like flat-top, isoflux, or cosecant beam patterns.

Acknowledgements. We hereby acknowledge the TEQIP-III of VSSUT, Burla for its continuous encouragement to our research work.

References

1. **Balanis CA** (2016) *Antenna Theory: Analysis and Design*. 3rd Edn. New York: John Willy & Sons.
2. **Srar JA, Chung KS and Mansour A** (2010) Adaptive array beamforming using a combined LMS-LMS algorithm. *IEEE Transactions on Antennas and Propagation* **58**, 3545–3557.
3. **Recioui A and Azrar A** (2007) Use of genetic algorithms in linear and planar antenna array synthesis based on Schelkunoff method. *Microwave and Optical Technology Letters* **49**, 1619–1623.
4. **Panduro MA, Reyna A and Camacho J** (2009) Design of scannable linear arrays with amplitude and phase optimisation for maximum side lobe level reduction. *International Journal of Electronics* **96**, 323–329.
5. **Panduro MA, Brizuela CA, Garza J, Hinojosa S and Reyna A** (2013) A comparison of NSGA-II, DEMO, and EM-MOPSO for the multi-objective design of concentric rings antenna arrays. *Journal of Electromagnetic Waves and Applications* **27**, 1100–1113.
6. **Panduro MA** (2007) Design of coherently radiating structures in a linear array geometry using genetic algorithms. *AEU International Journal of Electronics and Communications* **61**, 515–520.

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

7. **Panduro MA, Mendez AL, Dominguez R and Romero G** (2006) Design of non-uniform circular antenna arrays for side lobe reduction using the method of genetic algorithms. *AEU International Journal of Electronics and Communications* **60**, 713–717.
8. **Panduro MA, Brizuela CA and Covarrubias DH** (2008) Design of electronically steerable linear arrays with evolutionary algorithms. *Applied Soft Computing* **8**, 46–54.
9. **Panduro MA, Covarrubias DH, Brizuela CA and Marante FR** (2005) A multi-objective approach in the linear antenna array design. *AEU International Journal of Electronics and Communications* **59**, 205–212.
10. **Rajo-Lglesias E and Quevedo-Teruel O** (2007) Linear array synthesis using an ant colony-optimization-based algorithm. *IEEE Transactions on Antennas and Propagation* **49**, 70–79.
11. **Al-Azza AA, Al-Jodah AA and Harack-iewicz FJ** (2019) Spider monkey optimization: a novel technique for antenna optimization. *IEEE Antennas and Wireless Propagation Letters* **15**, 1016–1019.
12. **Subhashini KR** (2019) Antenna array synthesis using a newly evolved optimization approach: strawberry algorithm. *Journal of Electrical Engineering* **70**, 317–322.
13. **Khodier MM and Al-Aqeel M** (2009) Linear and circular array optimization: a study using particle swarm intelligence. *Progress in Electromagnetics Research B* **15**, 347–373.
14. **Khodier M and Saleh G** (2010) Beamforming and power control for interference reduction in wireless communications using particle swarm optimization. *AEU International Journal of Electronics and Communications* **64**, 489–502.
15. **Goudos SK, Moysiadou V, Samaras T, Siakavara K and Sahalos JN** (2010) Application of a comprehensive learning particle swarm optimizer to unequally spaced linear array synthesis with sidelobe level suppression and null control. *IEEE Antennas and Wireless Propagation Letters* **9**, 125–129.
16. **Mikki SM and Kishk AA** (2006) Quantum particle swarm optimization for electromagnetics. *IEEE Transactions on Antennas and Propagation* **54**, 2764–2775.
17. **Muralidharan R, Vallavaraj A, Mahanti GK and Patidar H** (2017) QPSO For failure correction of linear array of mutually coupled parallel dipole antennas with desired sidelobe level and return loss. *Journal of King Saud University – Engineering Sciences* **29**, 112–117.
18. **Yue Y, Cao L, Hu J, Cai S, Hang B and Wu H** (2019) A novel hybrid location algorithm based on chaotic particle swarm optimization for Mobile position estimation. *IEEE Access* **7**, 58541–58552.
19. **Li X, Duan B, Zhou J, Song L and Zhang Y** (2017) Planar array synthesis for optimal microwave power transmission with multiple constraints. *IEEE Antennas and Wireless Propagation Letters* **16**, 70–73.
20. **Mirhadi S and Soleimani M** (2015) Topology design of dual-band antennas using binary particle swarm optimization and discrete Green's functions. *Electromagnetics* **35**, 393–403.
21. **Saxena P and Kothari A** (2016) Ant lion optimization algorithm to control side lobe level and null depths in linear antenna array. *AEU International Journal of Electronics and Communications* **70**, 1339–1349.
22. **Dib N, Amaireh A and Al-Zoubi A** (2019) On the optimal synthesis of elliptical antenna arrays. *International Journal of Electronics* **106**, 121–133.
23. **Das A, Mandal D and Kar R** (2021) An optimal circular antenna array design considering the mutual coupling employing ant lion optimization. *International Journal of Microwave and Wireless Technologies* **13**, 164–172.
24. **Dib N, Goudos S and Muhsen H** (2010) Application of Taguchi's optimization method and self-adaptive differential evolution to the synthesis of linear antenna arrays. *Progress in Electromagnetics Research* **102**, 159–180.
25. **Sharaqa A and Dib N** (2014) Design of linear and elliptical antenna arrays using biogeography based optimization. *Arabian Journal for Science and Engineering* **39**, 2929–2939.
26. **Pappula L and Ghosh D** (2014) Linear antenna array synthesis using cat swarm optimization. *AEU International Journal of Electronics and Communications* **68**, 540–549.
27. **Rattan M, Patterh MS and Sohi BS** (2009) Optimization of circular antenna arrays of isotropic radiators using simulated annealing. *International Journal of Microwave and Wireless Technologies* **1**, 441–446.
28. **Dib N** (2016) Design of linear antenna arrays with low side lobes level using symbiotic organisms search. *Progress in Electromagnetics Research B* **68**, 55–71.
29. **Singh U and Salgotra R** (2018) Synthesis of linear antenna array using flower pollination algorithm. *Neural Computing and Applications* **29**, 435–445.
30. **Zhang C, Fu X, Ligthart LP, Peng S and Xie M** (2018) Synthesis of broadside linear aperiodic arrays with sidelobe suppression and null steering using whale optimization algorithm. *IEEE Antennas and Wireless Propagation Letters* **17**, 347–350.
31. **Das A, Mandal D, Ghoshal SP and Kar R** (2018) Moth flame optimization based design of linear and circular antenna array for side lobe reduction. *International Journal of Numerical Modelling* **32**, 1–15.
32. **Jamunaa D, Mahanti GK and Hasoon FN** (2020) Optimized inter-element arc spacing and ring radius in the synthesis of phase-only reconfigurable concentric circular array antenna using various evolutionary algorithms. *Electromagnetics* **40**, 104–118.
33. **Sallam T and Attiya AM** (2020) Low sidelobe cosecant-squared pattern synthesis for large planar array using genetic algorithm. *Progress in Electromagnetics Research M* **93**, 23–34.
34. **Yoshimoto E and Heckler MVT** (2019) Optimization of planar antenna arrays using the firefly algorithm. *Journal of Microwaves Optoelectronics and Electromagnetic Applications* **18**, 126–140.
35. **Maldonado AR and Panduro MA** (2015) Synthesis of concentric ring antenna array for a wide isoflux pattern. *International Journal of Numerical Modelling* **28**, 433–441.
36. **Greda LA, Winterstein A, Lemes DL and Heckler MVT** (2019) Beamsteering and beamshaping using a linear antenna array based on particle swarm optimization. *IEEE Access* **7**, 1–12.
37. **El-Hassan MA, Awadalla KH and Hussein KF** (2020) Shaped-beam circularly polarized antenna array of linear elements for satellite and SAR applications. *Wireless Personal Communications* **110**, 605–619.
38. **Li WT, Shi XW, Hei YQ, Liu SF and Zhu J** (2010) A hybrid optimization algorithm and its application for conformal array pattern synthesis. *IEEE Transactions on Antennas and Propagation* **58**, 3401–3406.
39. **Chaker H, Abri M and Badaoui HA** (2016) Hybrid evolutionary algorithm genetical swarm optimization for 1D and 2D annular ring unequally spaced antennas arrays synthesis. *Electromagnetics* **36**, 485–503.
40. **Omar OK, Debbat F and Stambouli AB** (2012) Null steering beamformer using hybrid algorithm based on honey bees mating optimization and tabu search in adaptive antenna array. *Progress in Electromagnetics Research C* **32**, 65–80.
41. **Amaireh AA, Al-Zoubi AS and Dib NI** (2019) Sidelobe-level suppression for circular antenna array via new hybrid optimization algorithm based on ant lion and grasshopper optimization algorithms. *Progress in Electromagnetics Research C* **93**, 49–63.
42. **Prabhakar D and Satyanarayana M** (2019) Side lobe pattern synthesis using hybrid SSWOA algorithm for conformal antenna array. *Engineering Science and Technology, an International Journal* **22**, 1169–1174.
43. **Mirjalili S** (2015) The ant lion optimizer. *Advances in Engineering Software* **83**, 80–98.
44. **Eberhart R and Kennedy J** (1995) A New Optimizer Using Particle Swarm Theory. *Proceedings of the Sixth International Symposium on Micro Machine and Human Science*, 39–43.
45. **Shi Y and Eberhart R** (1998) A modified particle swarm optimizer. *IEEE International Conference on Evolutionary Computation Proceedings. IEEE World Congress on Computational Intelligence*, 69–73.
46. **Alatas B, Akin E and Ozer AB** (2009) Chaos embedded particle swarm optimization algorithms. *Chaos, Solitons and Fractals* **40**, 1715–1734.
47. **Castorina G, Donato LD, Morabito AF, Isernia T and Sorbello G** (2016) Analysis and design of a concrete embedded antenna for wireless monitoring applications. *IEEE Antennas and Propagation Magazine* **58**, 76–93.
48. **Mauro GS, Castorina G, Morabito AF, Donato LD and Sorbello G** (2016) Effects of lossy background and rebars on antennas embedded In concrete structures. *Microwave and Optical Technology Letters* **58**, 2653–2656.

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press

Hrudananda Pradhan received the B.E. degree in Electronics and Telecommunication Engineering from the Veer Surendra Sai University of Technology (VSSUT), Burla, India, in 2006. He received his M.Tech. degree in Communication System Engineering from National Institute of Technology Rourkela, Odisha, India in 2009. He is currently working as an Assistant Professor and also pursuing his Ph.D. degree in the Department of Electronics and Telecommunication Engineering at VSSUT, Burla. He is a member of Indian Society for Technical Education. His main research interests are the design and optimization of linear and planar antennas, and reconfigurable antennas.

Biswa Binayak Mangaraj received the B.E. degree in Electronics and Telecommunication Engineering from the Sambalpur University, Burla, Odisha, India, in 1994 and the M.E. and Ph.D. degrees in Microwave Engineering from the Jadavpur University, Kolkata, India, in 2003 and 2012, respectively. He is currently working as an Associate Professor at Veer Surendra Sai University of Technology, Burla, India. He is a member of Indian Society for Technical Education and Institution of Electronics and Telecommunication Engineers. He has published 30 international journal papers and several international conference papers. His current research interests include computational electromagnetics, analysis, design, and coding of different antenna structures and optimization of antenna parameters using optimization algorithms.

Santanu Kumar Behera (SM'15) received the B.Sc.(Engg.) degree from the Veer Surendra Sai University of Technology, Burla, India, in 1990, and the M.E. and Ph.D.(Engg.) degrees from the Jadavpur University, Kolkata, India, in 2001 and 2008, respectively. He is currently a Professor and Head of the Department of Electronics and Communication Engineering, National Institute of Technology Rourkela, Odisha, India. His current research interests include planar antennas, dielectric resonator antennas, reconfigurable planar antennas, bio-electromagnetics, and RFID tags & reader antennas.

https://doi.org/10.1017/S1759078721000295 Published online by Cambridge University Press