Advances in Science and Technology Research Journal

*Advances in Science and Technology Research Journal*, 2026, 20(5), 386–399
https://doi.org/10.12913/22998624/216660
ISSN 2299-8624, License CC-BY 4.0

Received: 2025.11.30
Accepted: 2026.03.15
Published: 2026.04.01

# Interference-resilient design of aperiodic phased arrays for 5G applications using modified Laplacian invasive weed optimization

Ravi Kiran Balasa[1*], Malleswara Rao Veera[2], Tirumala Krishna Battula[3]

[1] Department of EECE, GITAM University, Visakhapatnam, India
[2] Department of E.C.E, AU College of Engineering, Andhra University, Visakhapatnam, India
[3] Department of E.C.E, University College of Engineering Kakinada, JNTUK, Kakinada, India
* Corresponding author's e-mail: rbalasa@gitam.edu

## ABSTRACT

Antenna arrays are essential components in present 5G wireless communication systems. They play a significant role in beamforming and interference suppression in desired directions. In present wireless communication scenarios, placing nulls in the interference directions and controlling the sidelobe power is essential due to an increase in EM pollution. To achieve this, antenna arrays with suitable beamforming algorithms need to be developed to generate the array patterns with low peak sidelobe level (PSLL) and desired nulls in the sidelobe angular region. Many traditional and evolutionary algorithms have been successfully applied to linear array synthesis. However, most of the methods are stuck at local optima and lead to local solutions with low accuracy. To overcome this, a new improved invasive weed optimization with Laplace distribution (MLIWO) method was introduced in this paper for the synthesis of linear antenna arrays. Additionally, this work focused on aperiodic linear antenna arrays, which provide better control over the radiation pattern without the need for non-uniform amplitude and phase excitations. The primary objective was to enhance the basic IWO algorithm performance by introducing a Laplace-based mutation in the weed position update equation. The proposed algorithm was applied to optimize the element positions of the linear array to suppress the PSLL and place the nulls in the desired directions. The proposed MLIWO method was used to synthesize the 28 and 32-element linear antenna arrays. The numerically synthesized results were compared with existing state-of-the-art genetic algorithms, particle swarm optimization, ant colony optimization, etc., and array designs reported in the literature. Simulation results indicate that the MLIWO method outperforms other methods in terms of accuracy and convergence speed. Finally, the proposed MLIWO method offers an effective method for designing efficient antenna arrays for defence and communication applications.

**Keywords:** 5G communications, aperiodic linear array, side lobe suppression, null placement, nature-inspired optimization, modified invasive weed optimization, Laplace mutation, antenna synthesis.

## INTRODUCTION

Antenna arrays [1] are used in almost all modern 5G wireless systems. They are part of wireless, satellite, mobile, and radar communication. They help improve directivity, signal quality, coverage, and overall system performance. For a communication system to function effectively, the antenna array must be carefully designed and optimized. In an antenna array, multiple antenna elements are assembled to act as a single system, distinct from the application of a single antenna. This is useful in regulating the transmission and reception of signals. A good antenna array can concentrate energy towards a desired direction. This enhances signal strength, reach, and general quality of communication. Meanwhile, improper design may result in undesired radiation in other directions, which can cause interference with other systems using the same frequency band. The reduction of the peak side lobe level (PSLL) is one of the primary design challenges that must be minimized to its lowest level. High side lobes consume power and cause interference. The other critical need is the capability to form deep nulls in some directions. These are nulls that are used to reject unwanted signals, noise, or sources of interference. This is particularly critical today, because the population of wireless [2] devices is growing faster, increasing the level of electromagnetic pollution. As such, the design of antenna arrays is a sensitive trade-off. The array must minimize side lobes, maintain a reasonable beam width, and accurately locate nulls as required. The combination of all these goals ensures effective communication, free from interruptions.

In a linear array, side lobes and nulls can be controlled in different ways. One approach is to adjust the spacing between elements while maintaining uniform excitation. The other way is to use non-uniform excitation with periodic element positions. However, in practice, it is not easy to use with those. Therefore, aperiodic element spacing is preferred. It provides more flexibility in shaping the array pattern while maintaining the simple feeding network. These arrays are called aperiodic arrays and can be formed either by changing the spacing of elements or by switching off selected elements. Unequally spaced arrays are created by shifting the element positions of a regular array. The number of elements remains the same, but the total array length may change. This non-uniform spacing helps avoid grating lobes and is helpful in wide-band scanning applications.

Previous works have shown that the interaction between radiofrequency (RF) fields and the human body is particularly significant in the presence of implantable medical devices. In 2016, Psenakova et al.[3] analyzed the RF exposure from a 2.4 GHz PIFA antenna and evaluated its effect on a cardiac pacemaker. Also, in 2016, Psenakova et al. [4] examined the impact of RF fields on a multilayer human skin model with a pacemaker structure. The study highlighted that different tissue layers absorb RF energy differently. These works emphasize the importance of controlling radiation patterns and minimizing unwanted RF exposure, particularly in modern wireless systems operating near the human body.

Aperiodic arrays [5–24] have been studied for many years, because they are simple to feed and offer reasonable pattern control. Their design involves nonlinear and nonconvex optimization. Manually finding the optimal parameter values for antenna array design is challenging, because the problem is complex and has numerous possible solutions. Some deterministic methods have been employed in the past, but they have often yielded unsatisfactory results. These methods typically require a good initial guess, and they may become unstable or become stuck in local optima instead of reaching the global optimal solution. Due to these limitations, researchers have begun using meta-heuristic algorithms, which are better suited for global optimization. These algorithms can search a wide solution space and avoid many of the problems seen in deterministic approaches. Several evolutionary algorithms have been employed for this purpose, including GA, DE, PSO, and ACO. These methods are effective in finding global solutions and have been successful in electromagnetic design problems.

Genetic algorithms (GAs) have continued to gain attention in antenna array pattern synthesis since the early studies. In 1997, Yan [6] showed further improvements using GA for array design. In 2004, Boeringer and Werner [10] used both GA and PSO for linear array synthesis. They observed that PSO was easier to implement and offered good performance, making it an attractive choice for many array design problems. To improve the performance of aperiodic arrays, researchers also explored hybrid methods that mix analytical techniques with stochastic search methods. In 2004, Salvatore Caorsi [11] and his team combined GA with the difference sets method to design thinned arrays. In 2012, Zhang [12] and his team explored the use of the orthogonal genetic algorithm (OGA) for designing thinned planar arrays. They created a crossover operator based on an orthogonal array, which helped reduce PSLL and also cut down the number of function evaluations needed. This made the design process more efficient compared to other methods. Several researchers have also used GA for large linear arrays.

In 2005, Khodier [18] and his team presented a new synthesis method for linear arrays using PSO. Their goal was to reduce the peak sidelobe level (PSLL) and control nulls simultaneously. They introduced new objective functions that allowed PSO to handle both targets together. Their study focused on position-only optimization for 28 and 32-element linear arrays. In 2007, Jin and Rahmat-Samii [19] demonstrated the flexibility of the PSO algorithm. They demonstrated its use for all types of optimization problems, including single-objective, multi-objective, real-world, and binary problems. They modeled swarm behavior using a randomized Newtonian mechanics approach and applied PSO to design unequally spaced and thinned antenna arrays.

Goudos and colleagues [20] then introduced a variant of the differential evolution (SADE) method for EM wave problems. They applied SADE to three different problems, including sidelobe suppression in a 32-element linear array by optimizing element spacing. SADE performed better than GA, PSO, and standard DE [21], and it required tuning only two parameters, making it easier to use.

Ant Colony Optimization (ACO) is one of the important algorithms. It is motivated by the searching behavior of ants and was first introduced in antenna arrays by Akdagli in 2002. Their work focused on optimizing complex weights for pattern nulling. Later, in 2007 [22], Eva and colleagues extended ACO to design thinned and unequally spaced arrays, showing that ACO could handle different types of array synthesis tasks. Pappula et al. [23] discussed the various synthesis problems, including 28 and 32-element linear array problems. Further, DE [24] with the SPS framework and nested optimization [25] have been used for linear array applications to suppress sidelobe levels.

To reduce computational complexity and improve exploration ability, Karimkashi and others introduced IWO for electromagnetic design problems. IWO, created by Mehrabian and Lucas in 2006 [26], is based on how weeds grow, spread, and adapt in nature. IWO [27, 28] was applied to both unequally spaced and thinned array designs. Comparisons with GA, PSO, and DE showed that IWO had faster convergence, lower PSLL, and needed fewer elements.

In 2010 [28], Roy and his team proposed the new version of IWO for the synthesis of unequally spaced circular antenna arrays. Since then, IWO has become widely used in electromagnetic optimization tasks. They tested the method on examples similar to those in earlier studies and reported lower PSLL than competing techniques.

IWO was also used by Karimkashi [29] and colleagues in 2010 to design a 20 × 10 planar array. Their design achieved a PSLL of –21.2 dB, showing the strong capability of the method for larger array structures. In 2017, Pappula [30] proposed the Cauchy mutated cat swarm optimization for the array synthesis. The Cauchy mutation enables the cats to move in good directions and achieves good solution accuracy. Erhan [31] proposed the seagull optimization algorithm (SOA) for PSLL suppression in linear arrays in 2021. SOA shows better performance in obtaining the desired performance metrics. Naruei [32] proposed a new meta-heuristic algorithm called Wild Horse Optimization for the EM optimization problem in 2022. Wang [33] proposed a variant of DE for sparse arrays in 2023 and suggested a strategic approach to optimizing them.

Recent literature shows a clear shift toward nature-inspired algorithms. Methods such as GA, PSO, DE, ACO, CSO, IWO, improved IWO [34], fruit-fly optimization [35], hybrid SSWOA algorithm [36], Hybrid synthesis [37], artificial Hummingbird optimization [38], and TLBO algorithm [39] have gained attention. Among them, IWO and its improved versions have shown promising results for low SLL and controlled null placement. Psenakova et al. [40] in 2022 proposed a computational approach for analyzing RF exposure effects using a multi-layered skin and cardiac pacemaker model. The authors introduced a neural network–based algorithm to predict electromagnetic field characteristics in the biological tissues efficiently

The IWO algorithm was launched in 2006 [26]. It is founded on the natural growth behavior of the weeds in an open field. Weeds develop their seeds on a case-by-case basis, appear fast, and compete among themselves in terms of space, light, and food. The best areas are occupied by the strongest weeds. This is a fundamental natural mechanism that underlies the concept of the IWO algorithm. This prevents the algorithm from early biases on a specific solution. The process repeats itself with the removal of weaker solutions, leaving better solutions to become even larger and spread further. This manner will enable global exploration and local refinement to be undertaken by IWO. However, there are limitations of the basic version of IWO. It converges prematurely in most instances or too gradually. It can also languish in small solution areas and lack an improved solution in the immediate vicinity. The primary cause of this issue is the way the positions of the weeds are updated during the optimization process. To address this issue, a Laplace mutation strategy [41] is incorporated into the IWO framework. The Laplace distribution offers the opportunity to make adjustments in the position of weeds controlled, while remaining flexible. This helps the algorithm avoid local optima and search more effectively. This results in a faster rate of convergence and an improvement in solution accuracy. In this paper, modified invasive weed optimization with Laplace distribution (MLIWO) was applied to optimize the distance between components in a linear array of antennas. The overall goal was to minimize the peak side lobe level and place deep nulls in the given directions of interference. The method offers enhanced pattern control while still maintaining realistic array constraints.

The main objectives of the carried work are:

- the new version of the classical IWO was proposed. It converges faster and avoids local optima. It can handle single-objective and multi-objective problems.
- the introduced MLIWO is operated on aperiodic synthesis for SLL reduction and null control.
- a detailed comparison with other popular algorithms was presented.

## MODIFIED INVASIVE WEED OPTIMIZATION WITH LAPLACE DISTRIBUTION

Invasive weed optimization is a nature-inspired algorithm, motivated by the growth and spread of weeds in nature. Weeds invade open land, reproduce, and compete for resources until the fittest plants survive. This biological behavior is utilized to resolve engineering problems, in which the aim is to estimate optimal parameters for a given objective function. IWO is especially useful for problems that are highly nonlinear, multidimensional, or difficult to solve with classical methods. The algorithm is simple to implement, flexible, and capable of achieving both exploration and exploitation.

The classical IWO has four steps to achieve the optimum solutions: initialization, reproduction, spatial dispersal, and competitive exclusion. Usually, random numbers for dispersal are drawn from a Gaussian (normal) distribution. In this work, the Gaussian distribution was replaced with a Laplace distribution to improve search ability and exploration. The modified version is referred to as the modified IWO with Laplace distribution (MLIWO), and the process in MLIWO is illustrated in Figure 1.

### Initialization

The first step in IWO is to initialize a population of seeds (candidate solutions) randomly across the search space. Each seed denotes a potential solution to the chosen problem. The initial positions of seeds are generated using uniform random numbers within the defined boundaries of the problem. If the search space is bounded by lower limit ($x_{min}$) and upper limit ($x_{max}$), the initialization is done as:

$$x_i = x_{min} + rand(0.1) \times (x_{max} - x_{min}) \quad (1)$$

where: $rand(0.1)$ generates a random number between 0 and 1, and $i = 1,2,\ldots,n$ represents each seed in the initial population.

Proper initialization is important, because it determines the diversity of the initial search and helps avoid premature convergence.

### Reproduction

In nature, the weeds that grow stronger and adapt better produce more seeds. IWO models this behavior by allotting a few seeds to every weed based on its strength (objective value). Better solutions generate more offspring, while weaker solutions produce fewer. For each i$^{th}$ plant, the seed calculation can be formulated as:

$$n_i = n_{min} + \frac{f_i - f_{worst}}{f_{best} - f_{worst}} (n_{max} - n_{min}) \quad (2)$$

Here, $f_i$ is the fitness of the i$^{th}$ plant, $f_{best}$ and $f_{worst}$ are the best and worst fitness values in the current population, and $n_{min}$ and $n_{max}$ are the minimum and maximum number of seeds that a plant can generate. This mechanism ensures that good solutions are explored more thoroughly while still maintaining diversity. The seed reproduction procedure is shown in Figure 2.

### Spatial dispersal

The seeds are dispersed around the weed plant after reproduction. This step simulates how seeds spread naturally in the environment. The spread is controlled by a parameter σ (standard deviation), which decreases over time, allowing a transition from global exploration to local exploitation. In the standard IWO [26, 30], seeds are scattered using a Gaussian distribution as:

$$x_{new} = x_{parent} + \sigma \times randn(0.1) \quad (3)$$

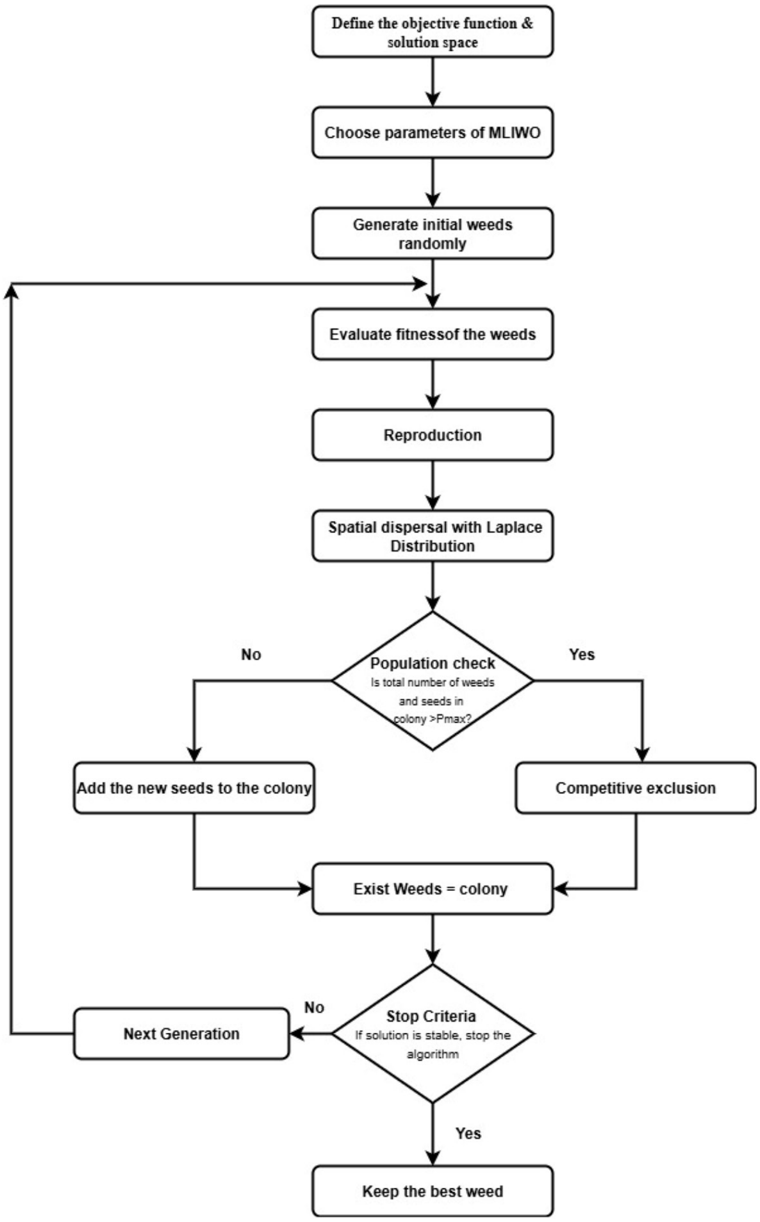


**Figure 1.** The flow chart of MLIWO

Here, randn(0.1) is a random number with normal distribution properties having an average of 0 and a standard deviation of 1. The value of σ decreases nonlinearly over iterations as [26, 30]:

$$\sigma = \left(\frac{T_{max}-t}{T_{max}}\right)^{n_l} \times \left(\sigma_{initial} - \sigma_{final}\right) + \sigma_{final} \quad (4)$$

where: present iteration is "t", the total number of generations is $T_{max}$. Initially, σ is large to encourage exploration, and it gradually reduces to refine solutions near the optimum. The term $n_l$ represents the nonlinear modulation index. It has been suggested that choosing $n_l = 3$ gives good results.

In the modified IWO, the Gaussian distribution is replaced with the Laplace distribution [41], which has heavier tails and offers improved search capabilities. The new dispersal equation becomes:

$$x_{new} = x_{parent} + \sigma \times L\,(0.1) \quad (5)$$

Here, $L$ (0,1) represents the properties of a Laplacian distribution random number with mean zero and scale 1. A Laplace variable can be generated using a uniform random number u ~ U(-0.5,0.5) as [41]:

$$L = -b\,sign(u)\,\ln(1 - 2|u|)\,, u \sim U(-0.5, 0.5) \quad (6)$$

where: $b$ is the scale parameter ($b$ = 1 for unit scale).

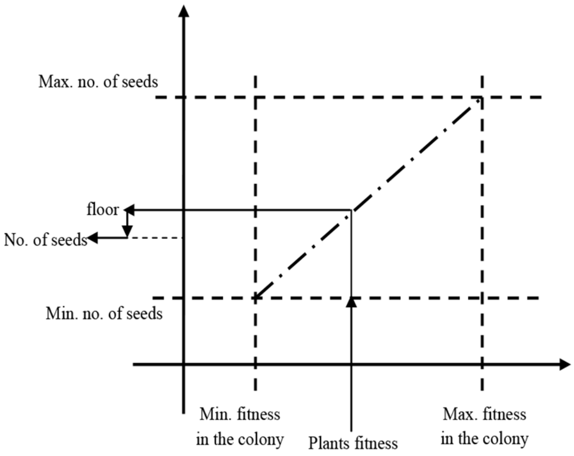


**Figure 2.** Procedure of seed reproduction [20]

The prolonged tracks of the Laplace distribution allow seeds to explore distant regions more effectively at the beginning, while still focusing locally in later iterations.

### Competitive exclusion

As the algorithm progresses, the total number of plants can grow beyond a manageable size. To maintain a stable population, a competitive exclusion mechanism is used. When the number of plants exceeds a predefined maximum limit N_max, only the fittest N_max plants survive to the next generation. This process eliminates weak solutions, allowing stronger ones to dominate the search space. It helps the algorithm focus on promising areas while keeping diversity among top-performing candidates. The IWO algorithm imitates the natural spreading and competition of weeds to solve complex optimization problems. By replacing the Gaussian-based seed dispersal with the Laplace distribution, the algorithm achieves a better balance between global search and solution accuracy. The heavier tails of Laplace distribution increase global search ability in early stages, while its central peak helps refine solutions later for MLIWO. This modification is especially useful in engineering problems, such as antenna array design, where controlling side lobe levels and null positions is crucial for enhancing the performance of communication systems.

### Computational complexity

Let $N$ be the number of antenna elements, $P$ be the population size, and $I$ be the maximum number of iterations. The computational cost is estimated by evaluating the fitness function for all population-based optimization algorithms. It is the O(N) per candidate solution. The GA algorithm involves selection, crossover, mutation, and fitness evaluation for each individual in the population. The computational complexity per iteration can be expressed as O (I(PN + P log P)). In PSO, the position and velocity updates for all particles are followed by fitness evaluations, resulting in a computational complexity of O(IPN). In ACO, the computational complexity per iteration is higher and can be approximated as O $(I(PN + P^2))$. Both IWO and the proposed MLIWO perform reproduction, spatial dispersal, and competitive elimination, as well as fitness evaluations. The Laplacian-based dispersal in MLIWO does not introduce any additional cost. Therefore, the computational complexity can be approximated as O(IPN).

## LINEAR ANTENNA ARRAY CHARACTERISTICS

The configuration under consideration is an equally illuminated linear array consisting of $2N$ identical elements. These are positioned alongside the x-axis. The structure, illustrating the spatial placement and symmetry of the elements with respect to the origin, is depicted in Figure 3.

The array factor (AF) of the linear array antenna [18, 23] is given as

$$AF(\theta) = 2\sum_{n=1}^{N} I_n \cos[kx_n \cos(\theta) + \varphi_n] \quad (7)$$

where: $k = 2\pi/\lambda$ denotes the wave number, θ is the azimuth angle and $I_n$, $\varphi_n$ and $x_n$ correspond to the excitation amplitude, phase, and position of the $n^{th}$ array element respectively.

## PROBLEM FORMULATION

The aim is to decide the suitable element positions of the chosen array antenna for the purpose of suppressing PSLL and placing nulls in the desired directions. The objective function is required to formulate to achieve the goals. To achieve this, the objective function is modeled as [18, 23]:

$$Fitness = \sum_i \frac{1}{\Delta\theta_i} \int_{\theta_{li}}^{\theta_{ui}} |AF(\theta)|^2 \, d\theta + \sum_k |AF(\theta_k)|^2 \quad (8)$$

where: $\theta_{li}$ and $\theta_{ui}$ define the angular regions where PSLL is to be suppressed; $\Delta\theta_i = \theta_{ui} - \theta_{li}$,

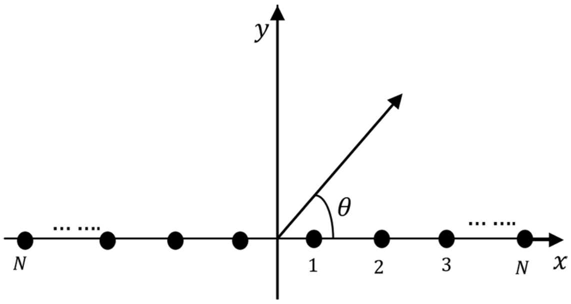


**Figure 3.** Illustration of a linear antenna array

angular width of each side lobe control region; $\theta_k$ represents the directions where nulls must be placed.

Equation 8 is divided into two parts. The first part represents the measure of the average power of the array pattern $|AF(\theta)|^2$ in sidelobe region. The second part forces the array factor to be very small at the specified null directions. The distance between the antenna elements must be maintained at a minimum of 0.25λ to avoid mutual coupling effects.

## NUMERICAL ILLUSTRATIONS: DESIGN EXAMPLES

To achieve the minimum PSLL and deep nulls in the required angles, the MLIWO algorithm described in Section 2 is utilized to optimize the objective functions. Two design examples are considered for simulation, where the array elements are modeled as isotropic radiators. For comparison with earlier studies, the best element positions of the array are also calculated using the MLIWO and the standard algorithms, IWO, ACO, PSO, and GA. The main parameters of the proposed MLIWO are listed in Table 1.

**Table 1.** MLIWO parameters

| S. No. | MLIWO | |
|---|---|---|
| | Parameter | Value |
| 1 | $n_{max}$ | 5 |
| 2 | $n_{min}$ | 0 |
| 3 | $\sigma_{initial}$ | 0.1 |
| 4 | $\sigma_{final}$ | 0.00015 |
| 5 | $P_{max}$ | 20 |
| 6 | $n_l$ | 3 |
| 7 | Initial population size | 10 |

These values were selected after testing various options and following recommendations from the literature [28]. Each algorithm is run ten times, with 1000 generations in each run. All optimization algorithms were simulated using a fixed number of 25,000 function evaluations to ensure a fair and consistent comparison of performance across methods. From these ten runs, the median result is selected to represent the performance. All simulations are carried out in MATLAB 2024b on a desktop equipped with an Intel(R) Core(TM) i5 processor and 8 GB of RAM.

### Design example: A 28-element linear array with controlling PSLL and placing nulls at 120°, 122.5°, and 125°

This design case discussed the 28-element array optimization aimed at reducing the PSLL in the side lobe region while placing nulls at 120°, 122.5°, and 125°. The radiation pattern achieved by MLIWO is presented in Figure 4, together with the pattern of a uniformly excited array and the IWO radiation pattern for comparison. The convergence curve of the fitness value for the MLIWO algorithm is shown in Figure 5. Table 2 presents the optimized position values obtained.

The proposed method achieves a peak side lobe level (PSLL) of –22.73 dB, as shown in Figure 4, which significantly improves over the conventional array and IWO array, where the side lobes remain higher. In addition to reducing PSLL, the algorithm successfully places deep nulls in the desired interference directions. The null depths obtained are –70.10 dB at $\theta$ = 120°, –72.14 dB at 122.5°, and -71.11 dB at 125° showing strong suppression of unwanted signals. The design example was synthesized using various state-of-the-art algorithms, and the best results are listed in Table 3 for comparison, along with the MLIWO results. The statistical performance of the 28-element linear array over 10 runs is given in Table 3. The results in Table 3 show that the proposed MLIWO method achieves the lowest mean SLL with a slight standard deviation. It demonstrates the robustness and effectiveness of precise PSLL and null control.

Figure 5 illustrates the convergence properties. The curve drops quickly in the early iterations and then gradually levels off as the algorithm approaches the optimal solution. This behavior shows that the Laplace-based modification allows

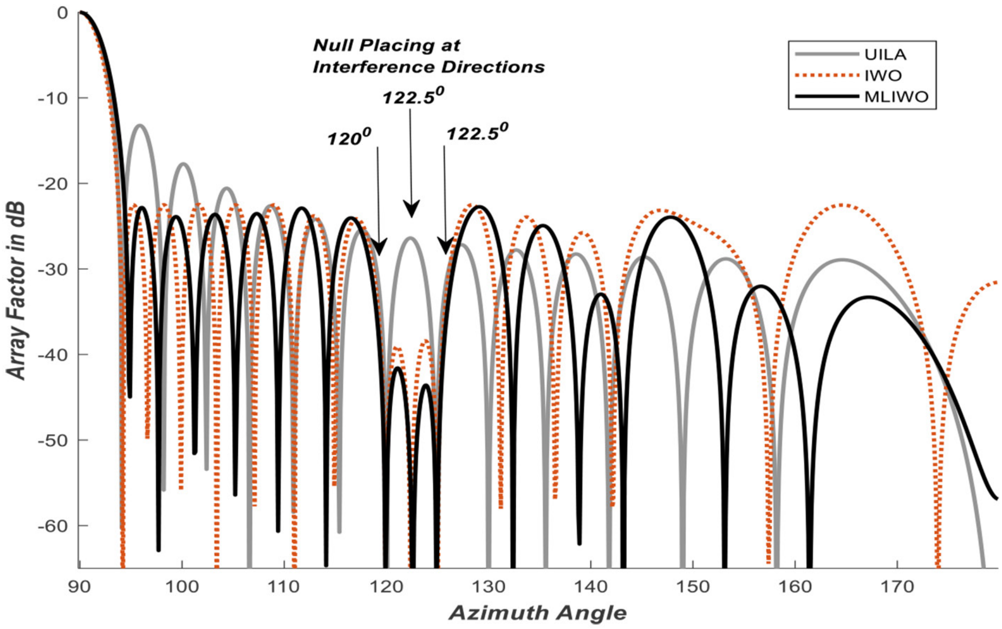


**Figure 4.** Array patterns of the MLIWO, PSO, and UILA 28-element array with nulls at 120°, 122.5°, and 125°

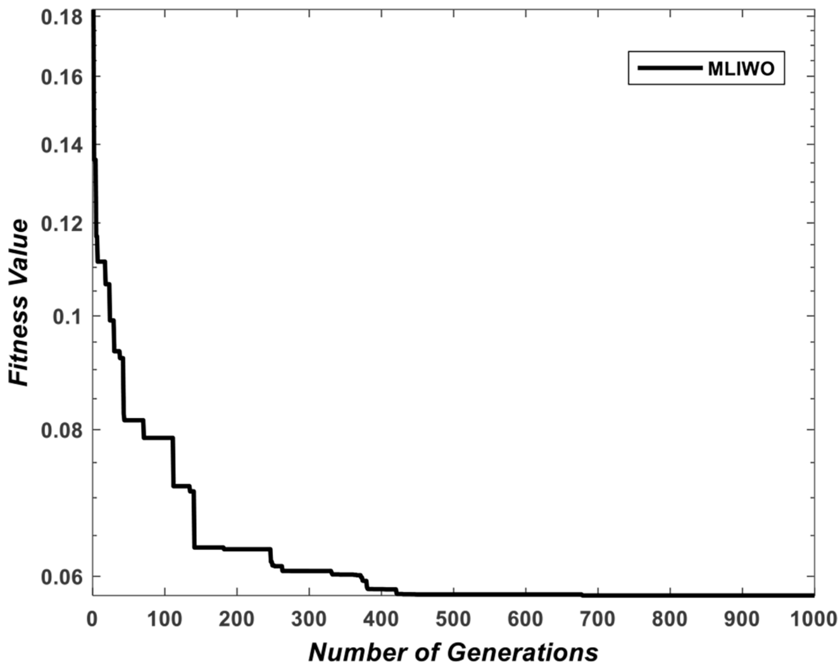


**Figure 5.** MLIWO convergence behavior for 28-element array synthesis

**Table 2.** Optimized positions* using MLIWO for 28-element antenna array synthesis

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.1250 | 0.6481 | 0.9465 | 1.4125 | 1.7949 | 2.2679 | 2.6165 | 3.1399 | 3.6898 | 4.1445 | 4.7183 | 5.5402 | 6.2946 | 7.0481 |

**Note:** *As the array is symmetric, only 14 element positions are provided in Table 2.

the algorithm to explore widely at the beginning and then fine-tune the solution in the later stages.

The numerical results in Table 3 demonstrate a significant improvement in performance when employing the proposed MLIWO algorithm for the 28-element unequally spaced array. Traditional methods, such as GA, PSO, and ACO, produce moderate sidelobe levels, with SLL values ranging from –13 dB to –15 dB. IWO performs better, reaching an SLL of –20.98 dB. However,

**Table 3.** Statistical performance of the 28-element linear array over 10 runs

| S. No. | Method for 28-element Array | SLL (Mean ± SD) (dB) | Null (Mean ± SD) (dB) | | |
|---|---|---|---|---|---|
| | | | 120° | 122.5° | 125° |
| 1 | GA [10] | −13.92 ± 0.18 | −56.90 ± 0.61 | −50.95 ± 0.72 | −54.80 ± 0.44 |
| 2 | PSO [18] | −13.31 ± 0.07 | −51.40 ± 0.58 | −50.10 ± 0.69 | −60.55 ± 0.47 |
| 3 | ACO [22] | −14.54 ± 0.15 | −56.10 ± 0.63 | −57.65 ± 0.74 | −59.45 ± 0.42 |
| 4 | IWO [26] | −20.63 ± 0.19 | −58.95 ± 0.66 | −59.90 ± 0.81 | −61.90 ± 0.39 |
| 5 | MLIWO | −22.39 ± 0.16 | −68.75 ± 0.71 | −69.90 ± 0.88 | −70.25 ± 0.46 |

MLIWO achieves the lowest SLL of –22.73 dB, showing stronger sidelobe suppression. A similar trend appears in null performance. While GA, PSO, and ACO achieve null depths between –52 dB and –60 dB, and IWO improves this to around –62 dB, the proposed MLIWO significantly deepens all three nulls, reaching values of –70 dB or better. These results confirm that MLIWO offers superior control over both sidelobe reduction and null placement, making it more effective than existing standard algorithms for complex synthesis (Table 4).

### Design example 2: A 32-element linear array with a null at 99°

This example illustrates the design of a 32-element linear antenna array featuring a wide null in the sidelobe region centered at approximately 99°. The optimized element positions are shown in Table 5. The array is optimized using the MLIWO algorithm. Figure 6 shows the radiation pattern produced by the proposed method, along with the patterns from a conventional uniformly excited array and the basic IWO for comparison. The MLIWO method achieves a low PSLL of –23.31 dB. It also produces a deep null of about –80.21 dB at 99°, showing strong and consistent suppression of unwanted signals over that angle. The statistical performance of the 32-element linear array over 10 runs is given in Table 6. Table 6 shows that MLIWO gives the lowest average SLL and the slightest variation, indicating better sidelobe reduction and stable performance across multiple runs.

The convergence curve for this case is shown in Figure 7. The curve drops sharply during the first few iterations and then settles smoothly as it approaches the optimum value. This behavior indicates that the Laplace-based modification helps the algorithm explore broadly at the start and then refine the solution stably.

Table 7 presents the best results from different state-of-the-art algorithms and compares them with the performance of MLIWO.

Table 7 shows that the MLIWO algorithm provides the best overall performance for the 32-element unequally spaced array. Traditional approaches, such as GA, PSO, and ACO, yield sidelobe levels between –17 dB and –19 dB, with null depths ranging from –59 dB to –62 dB. IWO improves these values, reaching an SLL of –20.42 dB and a null depth of –66.54 dB. However, MLIWO achieves a much lower SLL of –23.31 dB, showing stronger sidelobe suppression. It also produces a very deep null of –80.21 dB at 90°, which is far better than the other methods. These results clearly show that MLIWO offers superior control over both sidelobe reduction and null depth, making it the most effective method for this array configuration.

Overall, this example shows that the MLIWO algorithm can efficiently handle large antenna arrays. It achieves low sidelobe levels

**Table 4.** Comparison of synthesis results for a 28-element aperiodic antenna array

| S. No. | 28-element Array | SLL in dB | Null in dB | | |
|---|---|---|---|---|---|
| | | | 120° | 122.5° | 125° |
| 1 | GA [10] | -14.25 | -58.32 | -52.24 | -55.81 |
| 2 | PSO [18] | -13.23 | -52.74 | -51.66 | -61.46 |
| 3 | ACO [22] | -14.88 | -57.42 | -59.20 | -60.46 |
| 4 | IWO[ 26] | -20.98 | -60.2 | -61.85 | -62.85 |
| 5 | MLIWO | -22.73 | -70.10 | -72.14 | -71.11 |

**Table 5.** Optimized positions* using MLIWO for 32-element antenna array synthesis

| | | | | | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 0.1625 | 0.6615 | 0.9744 | 1.4781 | 1.7485 | 2.2222 | 2.6790 | 3.2085 | 3.6423 | 4.2666 | 4.6911 | 5.3432 | 6.2218 | 7.0281 |
| 8.0411 | 8.8425 | | | | | | | | | | | | |

**Note:** *As the array is symmetric, only 14 element positions are provided in Table 5.

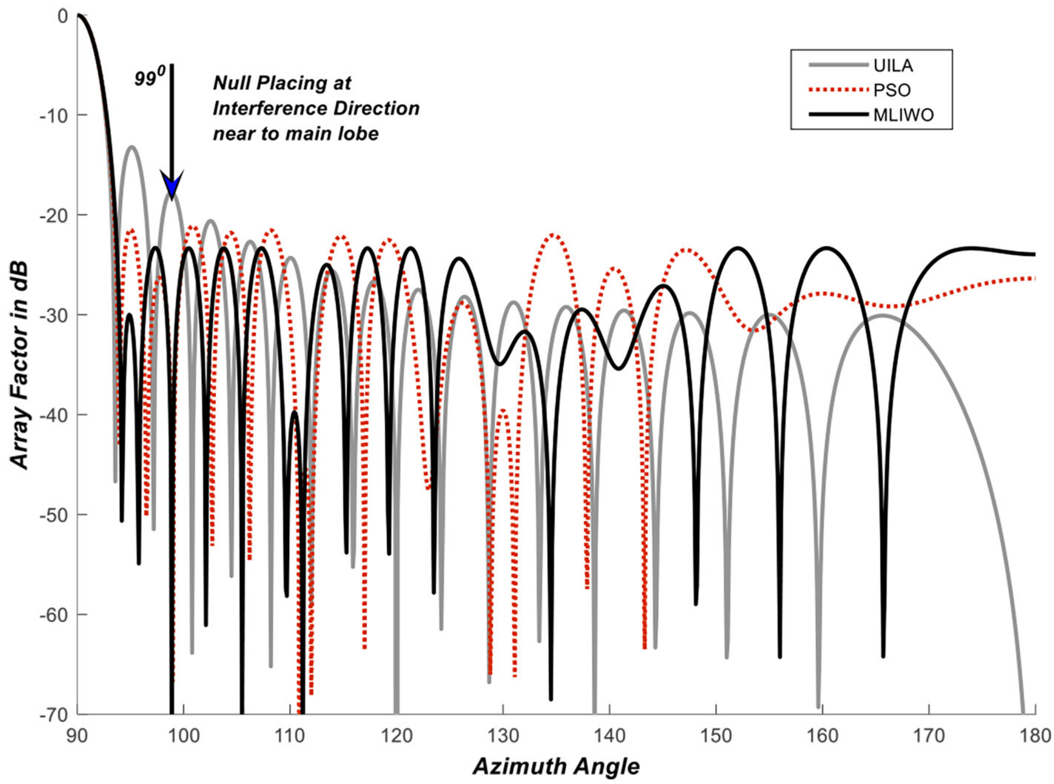


**Figure 6.** Array patterns of the MLIWO, PSO, and UILA 32-element array with nulls at 99°

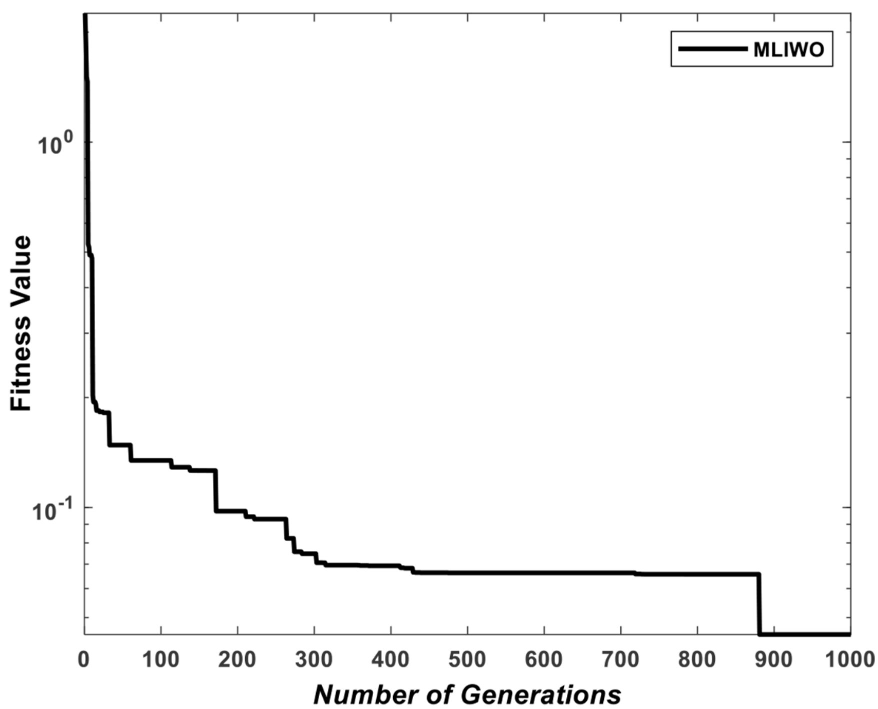


**Figure 7.** MLIWO convergence behavior for 32-element array synthesis

**Table 6.** Statistical performance of the 32-element linear array over 10 runs

| S. No. | Method | SLL (Mean ± SD) (dB) | Null at 90° (Mean ± SD) (dB) |
|---|---|---|---|
| 1 | GA [10] | −18.42 ± 0.52 | −57.20 ± 0.61 |
| 2 | PSO [18] | −18.05 ± 0.49 | −60.65 ± 0.58 |
| 3 | ACO [22] | −16.78 ± 0.28 | −59.20 ± 0.31 |
| 4 | IWO [26] | −19.85 ± 0.32 | −65.85 ± 0.35 |
| 5 | MLIWO | −22.95 ± 0.14 | −79.60 ± 0.21 |

**Table 7.** Comparison of synthesis results for a 32-element unequally spaced linear antenna array

| S. No. | 32-element Array | SLL in dB | Null in dB at 90° |
|---|---|---|---|
| 1 | GA [10] | -19.25 | -58.85 |
| 2 | PSO [18] | -18.73 | -62.12 |
| 3 | ACO [22] | -17.52 | -60.00 |
| 4 | IWO[ 26] | -20.42 | -66.54 |
| 5 | MLIWO | -23.31 | -80.21 |

**Table 8.** Parametric sensitivity analysis of MLIWO algorithm parameters

| S. No. | Parametric analysis of MLIWO | | | | |
|---|---|---|---|---|---|
| 1 | Range of $\sigma_{initial}$ | 0.05 | 0.1 | 0.2 | 0.3 |
| | Mean value of SLL and Null depth (in dB) | −21.98 & −78.12 | −22.95 & −79.60 | −22.41 & −78.80 | −22.01 & −78.02 |
| 2 | Range of $\sigma_{finitial}$ | 0.00015 | 0.0015 | 0.015 | 0.15 |
| | Mean value of SLL and Null depth (in dB) | −22.95 & −79.60 | −22.06 & −77.98 | −21.90 & −77.51 | −20.90 & −75.62 |
| 3 | Range of $P_{max}$ | 10 | 15 | 20 | 25 |
| | Mean value of SLL and Null depth (in dB) | −21.24 & −74.53 | −22.01 & −77.81 | −22.95 & −79.60 | −22.78 & −79.89 |
| 4 | Range of $n_l$ | 1 | 2 | 3 | 4 |
| | Mean value of SLL and Null depth (in dB) | −21.54 & −76.25 | −22.01 & −76.18 | −22.95 & −79.60 | −22.91 & −79.20 |

and wide sector nulls, making it suitable for applications where interference must be suppressed across a range of angles rather than at a single direction.

### Parametric analysis

A parametric analysis was carried out to study the influence of important MLIWO parameters, $\sigma_{initial}$, $\sigma_{final}$, $P_{max}$, and $n_l$. The results in Table 8 clearly demonstrate that these parameters have a significant impact on both sidelobe level (SLL) reduction and null depth performance. The best performance is achieved with $\sigma_{initial} = 0.1$, $\sigma_{final} = 0.00015$, $P_{max} = 20$, and $n_l = 3$, as this combination provides the lowest mean SLL and the deepest null depth with stable convergence.

## DISCUSSION

The results obtained from the 28 and 32-element array designs demonstrate the effectiveness of the proposed MLIWO algorithm. For the 28-element array design case, the MLIWO algorithm produces a low PSLL of –22.73 dB and null depths of -70.10 dB, -72.14 dB, and -71.11 dB in the null directions at 120°, 122.5°, and 125°, respectively. The results show that MLIWO simultaneously controls sidelobe levels and the placement of nulls in the desired directions. A -70 dB deep nulls are achieved in the desired directions. For a 32-element linear array, MLIWO reduces the PSLL to –23.31 dB and produces a very deep and wide null of –80.21 dB along the desired direction. The other metric is the convergence characteristics. The convergence curves indicate a fast decrease in the fitness value during the initial iterations. It is followed by a smooth and stable convergence.

The proposed Laplace-based mutation inclusion enables good searching of the search space initially. Later, it gradually shifts toward fine-tuning of the solution. The balance between exploration and exploitation helps the proposed MLIWO algorithm to avoid premature convergence and enhance solution accuracy. The computational cost of all algorithms increases along with the number of antenna elements. However, due to reduced control parameters and faster convergence, MLIWO requires fewer effective iterations to reach stable solutions. Hence, the proposed method exhibits improved scalability for large-scale antenna arrays compared to GA, PSO, and ACO, making it suitable for high-element array synthesis in advanced radar and 5G/6G systems.

There are a few limitations to the proposed approach, despite the promising results for linear antenna array synthesis. The study proposed linear arrays with fixed excitation amplitudes. Mutual coupling effects and fabrication tolerances have not been taken into account. Additionally, computational cost may increase along with the number of elements. Future work will address physical realization using real antenna elements, where mutual coupling effects will be fully incorporated through full-wave electromagnetic simulations and experimental validation.

## CONCLUSIONS

Aperiodic arrays are the most popular choice for the present generation of 5G wireless communication applications. An aperiodic array offers greater flexibility for controlling sidelobe power with low-complexity feeding networks.In this paper, maintaining the sidelobe power is accomplished by altering the distance between the antenna elements of the linear array antenna. Most of the classical gradient-based and nature-inspired optimization techniques have been applied to control the sidelobe power. However, these suffer from premature convergence and become stuck in local optimum solutions. To overcome these inefficiencies, the Modified Invasive Weed Optimization (MLIWO) algorithm, which introduces a Laplace-based mutation strategy to enhance the performance of aperiodic array synthesis, was proposed. MLIWO was applied to the large antenna arrays. The 28 and 32-element linear antenna arrays were synthesized to minimize the PSLL and null control in interference directions. The two types of cases were considered. Additionally, the convergence properties of the MLIWO demonstrate significant convergence towards the optimal solution. The convergence curves show that the Laplace modification played a crucial role in enabling the algorithm to approach the global optimum. A detailed comparison with GA, PSO, ACO, and classical IWO demonstrated the success of the proposed algorithm.

Overall, this work demonstrates that the MLIWO algorithm yields better performance in controlling sidelobe power. MLIWO can handle large arrays, achieving both narrow and wide nulls, and producing low PSLL. Such performance is especially required in present-day real-world applications where clear transmission is essential.

## REFERENCES

1. Balanis C. A., Antenna Theory: Analysis and Design, 4th ed. Hoboken, NJ, USA: Wiley, 2016.
2. Lo Sciuto G., Air pollution effects on the intensity of received signal in 3G/4G mobile terminal, Int. J. Energy Environ. Eng., 2019; 10(2): 221–229.
3. Psenakova Z., Smondrk M., Barabas J., Sciuto G. L., and Benova M., Simulation and assessment of pacemaker RF exposure (2.4 GHz) by PIFA antenna, in Proc. 2016 ELEKTRO, May 2016; 569–573.
4. Psenakova Z., Smondrk M., Sciuto G. L., and Benova M., Evaluation of radiofrequency field (2.4 GHz) effects to multilayer structure of human skin and pacemaker, in Information Technologies in Medicine, ITIB 2016, vol. 2, Cham, Switzerland: Springer, 2016; 245–254.
5. Khzmalyan D., Kondrat'yev A. S. Phase-only synthesis of antenna array amplitude pattern. Int J Electron. 2010; 81: 585–589. https://doi.org/10.1080/002072196136490
6. Yan K. K., Lu Y. Sidelobe reduction in array pattern synthesis using genetic algorithm. IEEE Trans Antennas Propag. 1997; 45: 1117–1122. https://doi.org/10.1109/8.596902
7. Chen K. S., He Z. S., Han C. L. A modified real GA for the sparse linear array synthesis with multiple constraints. IEEE Trans Antennas Propag. 2006; 54: 2169–2173.
8. Cen L., Yu Z. L., Ser W. Linear aperiodic array synthesis using an improved genetic algorithm. IEEE Trans Antennas Propag. 2012; 60: 895–902. https://doi.org/10.1109/TAP.2011.2173111
9. Ares-Pena F. J., Gonzalez J. A., Lopez E., Rengarajan S. R. Genetic algorithms in the design and

optimization of antenna array patterns. IEEE Trans Antennas Propag. 1999; 47: 506–510. https://doi.org/10.1109/8.768786

10. Boeringer D. W., Werner D. H. Particle swarm optimization versus genetic algorithms for phased array synthesis. IEEE Trans Antennas Propag. 2004; 52: 771–779. https://doi.org/10.1109/TAP.2004.825102
11. Caorsi S., Lommi A., Massa A., Pastorino M., Ds R. P. S. L. Peak sidelobe level reduction with a hybrid approach based on GAs and difference sets. IEEE Trans Antennas Propag. 2004; 52(4): 1116–1121. https://doi.org/10.1109/TAP.2004.823969
12. Zhang Q., Leung Y. W. An orthogonal genetic algorithm for multimedia multicast routing. IEEE Trans Evol Comput. 1999; 3(1): 53–62. https://doi.org/10.1109/4235.752920
13. Kurup D. G., Himdi M., Rydberg A. Synthesis of uniform amplitude unequally spaced antenna arrays using the differential evolution algorithm. IEEE Trans Antennas Propag. 2003; 51: 2210–2217. https://doi.org/10.1109/TAP.2003.816361
14. Lin C., Qing A. Y., Feng Q. Y. Synthesis of unequally spaced antenna arrays by using differential evolution. IEEE Trans Antennas Propag. 2010; 58: 2553–2561. https://doi.org/10.1109/TAP.2010.2048864
15. Goudos S. K., Siakavara K., Samaras T. Sparse linear array synthesis with multiple constraints using differential evolution with strategy adaptation. IEEE Antennas Wireless Propag Lett. 2011; 10: 670–673. https://doi.org/10.1109/LAWP.2011.2161256
16. Goudos S. K., Siakavara K., Samaras T. Self-adaptive differential evolution applied to real-valued antenna and microwave design problems. IEEE Trans Antennas Propag. 2011; 59: 1286–1298. https://doi.org/10.1109/TAP.2011.2109678
17. Zhang F., Jia W., Yao M. Linear aperiodic array synthesis using differential evolution algorithm. IEEE Antennas Wireless Propag Lett. 2013; 12: 797–800. https://doi.org/10.1109/LAWP.2018.2790907
18. Khodier M. M., Christodoulou C. G. Linear array geometry synthesis with minimum side lobe level and null controlling using particle swarm optimization. IEEE Trans Antennas Propag. 2005; 53: 2674–2679.
19. Jin N., Rahmat-Samii Y. Advances in particle swarm optimization for antenna designs: real number, binary, single-objective and multiobjective implementations. IEEE Trans Antennas Propag. 2007; 55(3): 556–567. https://doi.org/10.1109/TAP.2007.891552
20. Goudos S. K., Siakavara K., Samaras T. Application of comprehensive learning particle swarm optimizer to unequally spaced linear array synthesis with sidelobe level suppression and null control. IEEE Antennas Wireless Propag Lett. 2010; 9: 125–129. https://doi.org/10.1109/LAWP.2010.2044552
21. Bhattacharya R., Bhattacharyya T. K., Garg R. Position mutated hierarchical particle swarm optimization and its application in synthesis of unequally spaced antenna arrays. IEEE Trans Antennas Propag. 2012; 60: 3174–3181. https://doi.org/10.1109/TAP.2012.2196917
22. Eva R. L., Oscar Q. T. Linear array synthesis using an ant colony optimization based algorithm. IEEE Trans Antennas Propag. 2007; 49: 70–79.
23. Pappula L., Ghosh D. Linear antenna array synthesis using cat swarm optimization. AEU Int J Electron Commun. 2014; 68: 540–549. https://doi.org/10.1016/j.aeue.2013.12.012
24. Zhang R., Zhang Y., Sun J., Li Q. Pattern synthesis of linear antenna array using improved differential evolution algorithm with SPS framework. Sensors. 2020; 20(17): 5158. https://doi.org/10.1007/s10825-021-01711-w
25. Rocca P., Poli L., Anselmi N,. Massa A. Nested optimization for the synthesis of asymmetric shaped beam patterns in sub arrayed linear antenna arrays. IEEE Trans Antennas Propag. 2022; 70(5): 3385–3397. https://doi.org/10.1109/TAP.2021.3137176
26. Mehrabian A. R., Lucas C. A novel numerical optimization algorithm inspired from weed colonization. Ecol Inform. 2006; 1(4): 355–366.
27. Angeletti P., Berretti L., Maddio S., Pelosi G., Selleri S., Toso G. Phase-only synthesis for large planar arrays via Zernike polynomials and invasive weed optimization. IEEE Trans Antennas Propag. 2022; 70(3): 1954–1964.
28. Roy G. G., Das S., Chakraborty P., Suganthan P. N. Design of non-uniform circular antenna arrays using a modified invasive weed optimization algorithm. IEEE Trans Antennas Propag. 2011; 59(1): 110–118. https://doi.org/10.1109/TAP.2010.2090477
29. Karimkashi S., Kishk A. A. Invasive weed optimization and its features in electromagnetics. IEEE Trans Antennas Propag. 2010; 58(4): 1269–1278. https://doi.org/10.1109/TAP.2010.2041163
30. Pappula L., Ghosh D. Synthesis of linear aperiodic array using Cauchy mutated cat swarm optimization. AEU Int J Electron Commun. 2017; 72: 52–64. https://doi.org/10.1016/j.aeue.2016.11.016
31. Kurt E. K. G., Basbug S. Linear antenna array synthesis by modified seagull optimization algorithm. Appl Comput Electromagn Soc J. 2021;36. https://doi.org/10.13052/2021.ACES.J.361206
32. Naruei I., Keynia F. Wild Horse Optimizer: A new meta-heuristic algorithm for solving engineering optimization problems. London: Springer; 2022. https://doi.org/10.1007/s00366-021-01438-z
33. Wang X., Yao M., Zhang F. Teaching quality evaluation-based differential evolution and its application on synthesis of linear sparse arrays. Soft Comput. 2023; 27: 14735–14758. https://doi.org/10.1007/

s00500-023-08509-x

34. Pappula L., Ghosh D. Constraint-based synthesis of linear antenna array using modified invasive weed optimization. Prog Electromagn Res M. 2014; 36: 9–22. https://doi.org/10.2528/PIERM14021703
35. Darvish A., Ebrahimzadeh A. Improved fruit-fly optimization algorithm and its applications in antenna arrays synthesis. IEEE Trans Antennas Propag. 2018; 66(4): 1756–1766. https://doi.org/10.1109/TAP.2018.2800695
36. Prabhakar D., Satyanarayana M. Side lobe pattern synthesis using hybrid SSWOA algorithm for conformal antenna array. Eng Sci Technol Int J. 2019; 22(6): 1169–1174. https://doi.org/10.1016/j.jestch.2019.06.009
37. Elkhawaga A. M, Aboualalaa M, Dawood HS, Abd Elnaby MM. Hybrid synthesis approach for enhanced sidelobe suppression in linear and planar antenna arrays. Digit Signal Process. 2025; 166: 105373. https://doi.org/10.1016/j.dsp.2025.105373
38. Kumar S., Singh H. Design and synthesis of circular antenna array for low side lobe level and high directivity using artificial hummingbird optimization algorithm. Recent Adv Electr Electron Eng. 2025; 18(3). https://doi.org/10.2174/2352096516666230816091004
39. Khalili H., Mohammadpour-Aghdam K., Alamdar S. Sidelobe level reduction in pattern synthesis of a non-uniform series-fed microstrip antenna array using the TLBO algorithm. AEU Int J Electron Commun. 2024; 176: 155143. https://doi.org/10.1016/j.aeue.2024.155143
40. Psenakova Z., Smondrk M., Barabas J., Benova M., Brociek R., Wajda A., et al., Computational analysis of a multi-layered skin and cardiac pacemaker model based on neural network approach, Sensors, 2022; 22(17): 6359.
41. Laplace distribution. ScienceDirect Topics [Internet]. 2025 [cited 2025 Oct 5]. Available from: https://www.sciencedirect.com/topics/computer-science/laplace-distribution