*Progress In Electromagnetics Research B, Vol. 43, 333–354, 2012*

# DESIGN OF NON-UNIFORM CIRCULAR ANTENNA ARRAYS — AN EVOLUTIONARY ALGORITHM BASED APPROACH

**P. Ghosh[1, *], J. Banerjee[1], S. Das[2], and S. S. Chaudhury[1]**

[1]Department of Electronics and Communication Engineering, Jadavpur University, Kolkata 700032, India

[2]Electronics and Communication Sciences Unit, Indian Statistical Institute, 203 B T Road, Kolkata 700108, India

**Abstract**—Our main objective in this article is to achieve minimum side lobe levels for a specific first null beam-width and also a minimum size of the circumference by an optimization-based design method for non-uniform, planar, and circular antenna arrays. Our approach is based on a new variant of Particle swarm Optimization technique. This new technique is a hybrid of Local Neighborhood based PSO with Hierarchical PSO Algorithms termed as Hierarchical Dynamic Local Neighborhood Based PSO (HDLPSO) Algorithm. Three difficult instances of the circular array design problem have been presented to illustrate the effectiveness of the proposed HDLPSO algorithm. The design results obtained with HDLPSO have been shown to comfortably beat the results obtained with other state-of-the-art meta-heuristics like Genetic Algorithm (GA), Particle Swarm Optimization (PSO), Invasive Weed Optimization (IWO) and Differential Evolution (DE) in a statistically significant manner.

## 1. INTRODUCTION

Antennas having very directive radiation characteristics are required for the purpose of long distance communication. To meet this requirement a single antenna may be impotent. Antenna array that can be formed by combinations of many individual antenna elements in certain electrical and geometrical configurations is a solution to this problem. Antenna arrays have a plethora of applications including

*Received 18 May 2012, Accepted 6 September 2012, Scheduled 12 September 2012*

* Corresponding author: Pradipta Ghosh (iampradiptaghosh@gmail.com).

radar, sonar, radios, and third generation wireless communication systems [1–3]. In technologies that involve high power transmission, reduced power consumption and enhanced spectral efficiency, antenna arrays are found to be very practical. To obtain an excellent directive pattern it must be ensured that array elements add constructively in some preferred directions and add destructively and cancel each other in the remaining space.

Determination of the positions of array elements that jointly produce a radiation pattern which matches the preferred pattern as closely as possible is the primary design objective of antenna array geometry. In present literature designs of uniform and non-uniformly spaced linear arrays have already been reported. For the design of arrays nowadays researchers prefers different meta-heuristic algorithms. Many modern meta-heuristics were tried to accomplish optimized Side Lobe Level (SLL) and null control from the designed arrays [4–9] as because the classical derivative-based optimization techniques are prone to getting trapped in local optima and are strongly sensitive to initialization.

The popularity of circular arrays has also spread in mobile and wireless communications [10–12]. Panduro et al. [13] traces the first meta-heuristic approach towards the design of circular arrays. The article proposes the application of the real-coded Genetic Algorithm (GA) for designing circular arrays with maximal side lobe level reduction coupled with the constraint of a fixed beam width. Shihab et al. in [14] achieved better results as compared to those reported in [13] by applying Particle Swarm Optimization (PSO) algorithm that draws motivation from the smart collective behavior of a group of social creatures, to the same problem. Recently, Panduro et al. [15] worked on the design of scanned circular array by comparing three authoritative population-based optimization algorithms — PSO, GA, and Differential Evolution (DE). The three algorithms were compared on a single instantiation of the design problem with an objective of studying the behavior of array factor for the scanning range of $0^\circ$ to $360^\circ$ in angular steps of $30^\circ$ and the number of antenna elements were set to 12 and for a uniform separation of $d = 0.5\lambda$, optimizing excitation current amplitudes and phase perturbations. Chen et al. [16] presented a new way for finding an optimal solution of complex antenna array design problem using a Crossed Particle Swarm Optimization Algorithm. In [17], Khodier and Al-Aqeel provided a study of Antenna Array design using Particle Swarm Optimization. There are many other articles regarding circular antenna array design in various journals and scientific magazines [18–23]. Most of the articles [8–23] provide enough evidence of the fact that Swarm based optimization is far more effective that stochastic optimization method for antenna array design problems. In general, the superiority of such swarm based method for solving any kind of engineering optimization problems are proved in terms of benchmark problems in different articles [24–31].

As PSO is a stochastic search process hence it is not free from false or untimely convergence, in particular over multimodal fitness landscapes. To eradicate this problem PSO needs to be modified. Our main objective in this article is to use an improved variant of PSO named Hierarchical Dynamic Local Neighborhood Based PSO (HDLPSO), which is a hybrid of Hierarchical PSO and LPSO, for designing non-uniform circular arrays with optimized performance with respect to SLL, directivity, and null control in a scanning range of $[0^\circ, 360^\circ]$. The efficiency and effectiveness of our proposed method for this particular design problem is shown and discussed through various experimental results in the later part of this manuscript.

The rest of this paper is organized as follows: Section 2 discusses the design problem; Section 3 presents a brief discussion of classical PSO algorithm. Section 4 introduces our novel optimization algorithm along with the discussions of the ancestors. Section 5 presents the experimental data and Section 6 concludes this manuscript.

## 2. ARRAY FACTOR OF CIRCULAR ARRAY AND DESIGN PROBLEM

We consider a non-uniform and planar circular antenna array as shown in Figure 1. $N$ elements are spaced non-uniformly on a circle of radius $r$ in the $x$-$y$ plane. The elements of the array are considered to be isotropic sources, so that its array factor can represent the radiation pattern of the array.

Formulation of the array factor requires the following:

- Excitation current amplitude $I_n$,
- Phase $\beta_n$,
- Angular position of the $n$-th element $\varphi_n$,
- Circular arc separation between any two adjacent elements ($d_n$ — the distance between elements $n$ and $n-1$).

The expression for the array factor in the $x$-$y$ plane can be represented as:

$$AF(\varphi) = \sum_{n=1}^{N} I_n . e^{j.(kr.\cos(\varphi-\varphi_n)+\beta_n)} \tag{1}$$

In the above expression $kr$ and $\varphi_n$ can be given as

$$\left.\begin{aligned} kr &= \frac{2\pi r}{\lambda} = \sum_{i=1}^{N} d_i, \\ \varphi_n &= \frac{2\pi}{kr}\sum_{i=1}^{n} d_i, \end{aligned}\right\} \tag{2}$$

We can direct the peak of the main beam in the $\varphi_0$ direction by choosing excitation phase of the $n$th element as

$$\beta_n = -kr \cdot \cos(\varphi_0 - \varphi_n) \tag{3}$$

The simplified array factor of the non-uniform circular array is

$$AF(\varphi) = \sum_{n=1}^{N} I_n \exp\left(j \cdot kr \cdot \left(\cos(\varphi - \varphi_n) - \cos(\varphi_0 - \varphi_n)\right)\right) \tag{4}$$

Parameters of this expression are $I_n$ and $\varphi_n$ (i.e., $d_i$) values of the elements. The peak of the radiation pattern is directed along the $x$-axis i.e., $\varphi_0 = 0$.

For the evaluation of fitness (or cost) function different parameters such as gain, side lobe level, radiation pattern, and size can be used. Our goal is to design a circular antenna array with minimum side lobes levels for a specific first null beam-width (FNBW) to ensure maximum directivity of the antenna, and also to minimize the circumference of

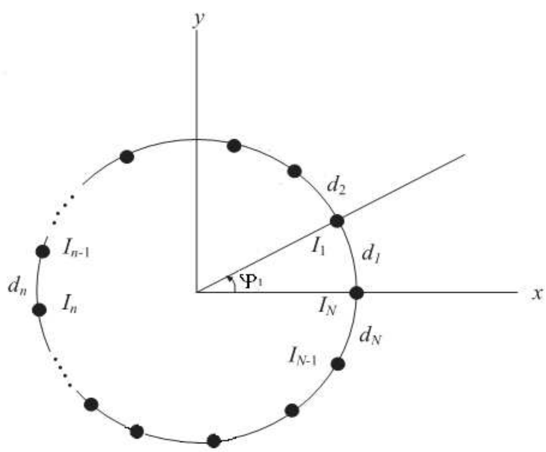


**Figure 1.** Geometry of a non-uniform circular antenna array with $N$ isotropic radiators.

the circular array which is of paramount importance in the modern world where the focus is increasingly on miniaturization.

Along with the average side lobe level, the maximum side lobe level has been incorporated in the fitness function and it serves the directivity purpose. The objective function satisfying the above requirements is given as follows

$$f_{NU} = |AF(\varphi_{NULL1})| + |AF(\varphi_{NULL2})| \tag{5}$$

$$f_{SLA} = \frac{1}{\pi + \varphi_{NULL1}} \int_{-\pi}^{\varphi_{NULL1}} |AF(\varphi)| d\varphi + \frac{1}{\pi - \varphi_{NULL2}} \int_{\varphi_{NULL2}}^{\pi} |AF(\varphi)| d\varphi \tag{6}$$

$$f_{MSL} = |AF(\varphi_{MSLL1}| + |AF(\varphi_{MSLL2}| \tag{7}$$

where $\varphi_{NULL1}$ and $\varphi_{NULL2}$ are the two angles at the null, and $\varphi_{MSLL1}$ is the angle where the maximum side lobe level is obtained in the lower band $[-\pi, \varphi_{NULL1}]$ and, $\varphi_{MSLL2}$ is the angle where the maximum side lobe level is obtained in the upper band $[\varphi_{NULL2}, \pi]$.

Another objective is the minimization of circumference. This reduces the dimension of the designed array. The mathematical expression is

$$f_D = \sum_{i=1}^{N} d_i \tag{8}$$

where, $d_i$'s have their usual meanings.

The final cost function is obtained by combining all the objectives which is as follows,

$$F = a_1 * f_{NU} + a_2 * f_{SLA} + a_3 * f_{MSL} + a_4 * f_D, \tag{9}$$

where $a_i$s represent the respective weights assigned to the sub-functions. We have to consider Eq. (9) as the function used for optimization.

## 3. CLASSICAL PSO ALGORITHM

The PSO algorithm is an evolutionary algorithm capable of solving difficult multidimensional optimization problems in various fields introduced in 1995 by Kennedy and Eberhart [24, 25]. Random initialization of a population of candidate solutions (particles) over the fitness landscape is the starting point of classical PSO. However, during the search PSO does not uses direct recombination of genetic material between individuals unlike other evolutionary computing techniques but instead works by depending on the social behavior of the particles in the swarm. Therefore, by merely adjusting the trajectory of each individual towards its own best position and toward the best particle of the entire swarm at each time-step (generation), it detects and finds the global best solution. In a search space of $D$-dimension, the position vector and velocity of the $i$-th particle are given by $\vec{X}_i = (x_i^1, x_i^2, \ldots, x_i^D)$ and $\vec{V}_i = (v_i^1, v_i^2, \ldots, v_i^D)$ respectively. Adjustment of position vector and velocity were made and at each time steps the objective function to be optimized $f(\vec{X}_i)$ is evaluated with the new coordinates. The representation for the $d$-th dimension of the $i$-th particle in the swarm of velocity and position is given below:

$$\begin{aligned} v_i^d &= \omega * v_i^d + c_1 * rand1_i^d * \left(pbest_i^d - x_i^d\right) + c_2 * rand2_i^d * \left(gbest_i^d - x_i^d\right), \\ x_i^d &= x_i^{d-1} + v_i^d \end{aligned} \tag{10}$$

where $c_1$ and $c_2$ are the acceleration constants with $c_1$ controlling the consequence of the personal best position and $c_2$ determining the effect of the best position found so far by any of the particles, $rand1_i^d$ and $rand2_i^d$ are two uniformly distributed random numbers in the range [0, 1], $\omega$ is the inertia weight that takes care of the influence of the previous velocity vector and balances between the global and local search abilities, $pbest_i = (pbest_i^1, pbest_i^2, \ldots, pbest_i^D)$ is the best earlier position giving the best fitness value $pbest_i$ for the $i$th particle and $gbest = (gbest^1, gbest^2, \ldots, gbest^D)$ is the best position discovered by the whole population.

## 4. HIERARCHICAL D-LPSO ALGORITHM

### 4.1. A Brief Overview of the Ancestor Algorithms

#### *4.1.1. Local Neighborhood Based PSO*

Global PSO and local PSO are the two main variants of PSO. The variants may limit the velocity of a particle by a maximal value $V_{\max}$, while some variant linearly varies $\omega$. In the local version of PSO, instead of learning from the personal best and the best position achieved so far by the whole population in the global version, the particle's personal best modifies its velocity and the best performance achieved so far within its neighborhood. Thus the velocity updating equation can be represented as:

$$v_i^d = \omega * v_i^d + c_1 * rand1_i^d * \left(pbest_i^d - x_i^d\right) + c_2 * rand2_i^d * \left(lbest_i^d - x_i^d\right), \tag{11}$$

where $lbest_i = (lbest_i^1, lbest_i^2, \ldots, lbest_i^D)$ is the best position achieved within its neighborhood.

Various mechanisms have been designed in order to increase the variety among the particles of a swarm. Different neighborhood topologies have been investigated for PSO as the topology of the neighborhood plays a substantial role in PSO. A ring topology is used to define the neighborhood through the particle's index in the *lbest* model of PSO. Different neighborhood structures are proposed and discussed for the enhancement of this *lbest* model of PSO. Multi-swarm [26] and subpopulation [26] are used by some variants. Sub-groups may be treated as special neighborhood structures. The swarms are predefined or dynamically attuned in order to the distance in the existing local versions of PSO with different neighborhood structures and the multi-swarm PSOs. A dynamic or randomly assigned topology is used by the dynamic multi-swarm optimizer.

### *4.1.2. Hierarchical PSO*

The particles are arranged in a hierarchy which defines the neighborhood structure in the hierarchical version of PSO. Each of the particles is neighbored both to itself and its parent in the hierarchy. The hierarchy is a regular tree-like structure. The hierarchy is defined by the *height*, *branching degree* [27], i.e., the maximum number of children of the inner nodes, and *total number of nodes* of the corresponding tree. In this hierarchy, all inner nodes, except the inner nodes on the deepest level which might have a smaller number of children, have the same number of children. Hence, the maximum difference is at most one between the numbers of children of inner nodes on the deepest level. The upward and downward movement of the particles in the hierarchy highly influences the best particle of the swarm. The evaluation of the objective function and velocity update in each iteration determines the new positions of the particles. The best solution obtained by the particles in the child nodes is compared to the best solution of $j$th particle in a node of the tree. This is done for the entire particle in that node. Particles $i$ and $j$ swap their places if best solution obtained by any particle (say $i$th particle) in the child node is better than that of $j$th particle. The starting point of this comparison is from the top of the hierarchy and then proceeds in a breadth-first manner down the tree. At each iteration, the particle which has the global best position of the hierarchy moves up one level of the hierarchy. A particle's best position so far and the best position of the individual that is directly on the top of the considered particles in the hierarchy moulds its velocity.

In case of H-PSO [27] depending on the fitness growth of the individuals, the neighborhood of a particle changes continuously. This changing arrangement of the particles can help to preserve the diversity in the search. Different influence for the particles at different positions is led because of the continuous change in neighborhood of a particle. When the particle with the current best found solution can (indirectly) manipulate all the other particles after it has reached the top of the hierarchy.

Optimization behavior of H-PSO is influenced by the structure of the hierarchy and the branching degree $d$. For example, the performance might be better initially if the branching degree is higher, on the other hand because of a smaller value of $d$ the performance may be worse in the beginning of optimization process in finding the best solution but in the end it might further improve the objective function value. There is a dynamic change in the branching degree for this reason. The hierarchy is traversed starting at the root node when the branching degree is decreased from $d$ to $d-1$. This is done so that if there is an excess of the number of children compared to new required branching degree always one of the direct sub trees below the considered node is removed. Based on the quality of the particles in the topmost nodes of all sub trees of the considered nodes, i.e., all children of the considered node, the sub tree is removed. For the entire tree this procedure is repeated. This removal of sub tree causes the remaining tree to have a branching degree $d-1$ with fewer nodes than before. At the bottom of the hierarchy the removed nodes are then evenly inserted. So that the number of children of all the nodes on the second last level differs at most by one the removed nodes are appended one by one so. A new level is added to the hierarchy and the procedure is sustained until all removed nodes are reinserted if all of these nodes have $d-1$ children. The branching degree reduction is done in every $f_{adapt}$th iteration, this $f_{adapt}$ is called decrease frequency. Branching degree is decreased by $k_{adapt}$ known as decrease step size. For $k_{adapt} > 1$ the reduction procedure is applied consecutively (i.e., the branching degree is always reduced in steps of 1) until the hierarchy has the required branching degree. Until a certain minimum is reached this is continued. For choosing the *sub-tree* which is to be removed two strategies are employed — removing the *sub-tree* with the worst root node or removing the *sub-tree* with the best root node.

## 4.2. Our Proposed Algorithms

### *4.2.1. D-LPSO*

The Dynamic Local Neighborhood based Particle Swarm Optimization (DLPSO) [28] is a variant of PSO constructed based on the local version of PSO employing a new neighborhood topology. In case of PSO, it has been found that satisfactory results can be obtained using smaller population size. PSO with smaller neighborhoods has better performance on complex problems also. In case of DLPSO smaller neighborhoods are used. As a result the convergence velocity of the population decreases, diversity increases and better solutions are achieved for multi-modal problems. The population is divided into small sized swarms. Size of each swarm varies randomly within a certain limit (max limit value = 5). So a sub-swarm may contain any number of particles between 1 and 5. This is done to get the maximum benefit of the sub-swarms of different sizes. Also we have included a check after a certain no of iterations to verify the convergence rate of the swarms of different size. According to the outcome the program is assigned a greater probability for the most converging size of sub-swarm while sub swarms of other sizes have a lesser probability. Each sub-swarm uses its own members to search for better area in the search space. Since the small sized swarms are searching using their own best historical information, they are easy to converge to a local optimum because of PSO's convergence property. In order to avoid it we must allow information exchange among the swarms. So every sub swarm needs to be aware of the position of the best particle in other sub-swarm. We have incorporated an information exchange schedule according to which in every iteration all the sub-swarm's best particle exchange information. We want to keep more information including the good ones and the not so good ones to add the varieties of the particles and achieve larger diversity. So a randomized regrouping schedule is introduced to make the particles have a dynamic changing neighborhood structures. That means we haven't included any parameter for this selection. Each sub-swarm containing at most five particles search for better location and they may converge to near a local optimum.

After regrouping, the particles previously belonging to a common sub-swarm now belong to different sub-swarms and get the opportunity to modify their velocity and position learning from the new swarm members. In every $k$ generation, the population is regrouped randomly and starts searching using a new configuration of small swarms. The value of $k$ is kept between 5 and 10. In this way, the information obtained by each swarm is exchanged among the swarms as a particle belongs to different swarms during the search process and it carries the information obtained in the previous swarm and uses this information to influence other particles' movement in the new swarm. With the randomly regrouping schedule, particles from different swarms are grouped in a new configuration so that each small swarm's search space is enlarged and better solutions are possible to be found by the new small swarms. The procedure is shown in Figure 2.

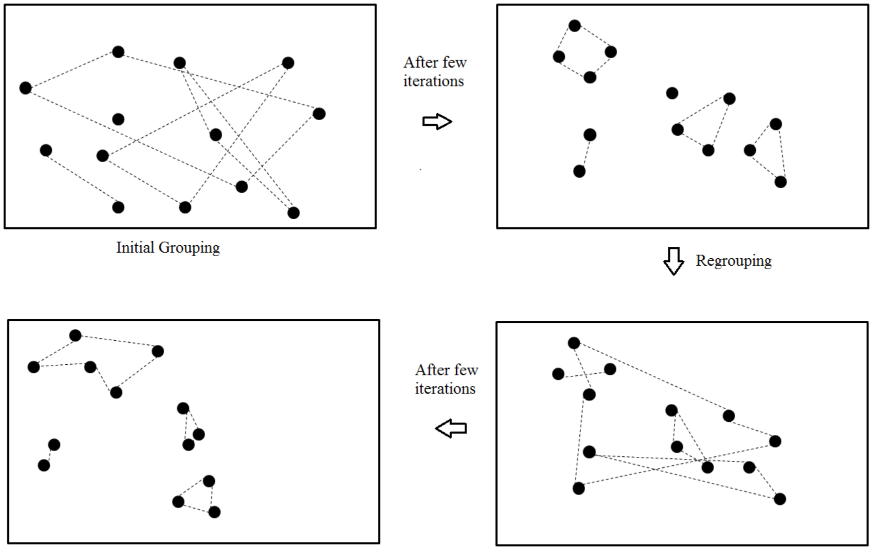


**Figure 2.** DLPSO's search process.

### *4.2.2. Hierarchical D-LPSO Algorithm*

In this algorithm Hierarchical PSO is combined with DLPSO. The steps of our proposed PSO algorithm is

**Step 1.** The Initialized population is divided into stages. First stage contains 1 particle; next stage contains maximum $n$ elements (Initially $n = 2$). $N$-th stage can contain maximum $n^N$ particles.

**Step 2.** Each particle is assigned randomly a parent from the previous stage except the particle on 1st stage.

**Step 3.** Evaluate each particles Fitness.

**Step 4.** Now DLPSO is applied along each stage except 1st stage.

**Step 5.** Now each particle compares its fitness with its assigned parent. If its position is better than its parent's position, they are swapped.

**Step 6.** Step 2 to Step 5 continues until next 1/15th of the total FEs is completed. Now $n = n + 1$, if $n < 5$ and Step 1 takes place again. If $n = 5$, $n$ is reinitialized to 1 i.e., $n = 1$. (N.B. $n$ maybe reinitialized to 2, but our idea is to provide each particle some time to search slowly and locally without modification of its *lbest* position by neighborhood particles. Also if we reinitialize $n = 1$ instead of $n = 2$, the result obtained is much better.) Repeating step 1 performs the regrouping of hierarchy.

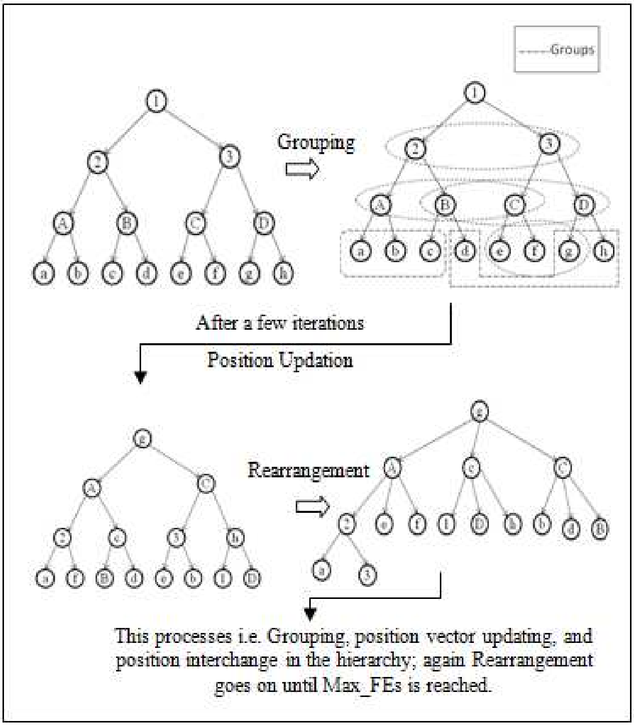


**Figure 3.** HDLPSO's search process.

**Step 7.** When maximum no of FEs is reached all the processes are stopped and the result is shown.

In the velocity update equation, we have used a constriction factor to avoid the unlimited growth of the particles' velocity. Also using this factor, a better result is obtained, which was proposed by Clerc and Kennedy [29]. Eq. (11) becomes

$$V_i^d = \chi * (\omega * V_i^d + c_1 * rand1_i^d * (pbest_i^d - X_i^d) + c_2 * rand2_i^d * (lbest_i^d - X_i^d)) \quad (12)$$

where $\chi$ is the constriction factor given by

$$\chi = 2/\left|2 - c - \sqrt{c^2 - 4c}\right| \quad (13)$$

where $c = \sum_i c_i$.

Another feature is added to this algorithm for better result. If the minimum fitness value obtained remains constant for 1000 FEs, we reinitialize some particle to its *pbest* position and other to a position in between their *pbest* and *lbest* position.

This procedure is graphically shown in Figure 3.

The pseudo codes of Step 1 to Step 5 are given below. Step 6 and Step 7 can be easily coded.

---

*Step 1*

```
N = total no of particles.
FEs = No of FEs covered;
Max_FEs = Max No of FEs
Count = Max no of particles in second stage.
Icount = Max no of particles in any stage Stage (1,1) = Any particle from the population.
Stagecount = 2;
Icount = Count;
While 1
        For i = 1: Icount
                Stage (Stagecount, i)  = any particle from the rest of the population
                If All the particles Covered
                        Break;
                End
                End
                If all the particles are covered
                        Break;
                End
                Icount = Icount*Count;
                Stagecount = Stagecount + 1;
End
```

---

*Step 2*

```
Parent (Stage (1,1)) = Stage (1,1);
Icount = Count;
For  j = 1: Stagecount
            For i = 1: Icount
                      Parent (Stage (Stagecount, i)) = any particle from the previous stage.
            End
                      Icount = Icount*Count;
End
```

---

*Step 3*

```
For i = 1 : N
           Evaluate Each particle's Fitness;
End
```

---

*Step 4*

```
For j = 1: Stagecount
           Apply DLPSO;
End
```

---

*Step 5*

```
For j = 1: Stagecount
    For i = 1: Icount
        If Fitness (Parent (Stage (Stagecount, i))) < Fitness (Stage (Stagecount, i))
            Swap the particles.
        End
    End
  Icount = Icount*Count;
End
```

In the above algorithm, we are repeating steps 2 to 5 for 1/15th of the total FEs, and after that we start the process again. It has been found empirically that good results are obtained by regrouping the hierarchy after 1/15th of the total FEs. Steps 2 to 5 yield sufficiently good result within this nos. of FEs and more nos. of FEs are not required. When applying DLPSO in step 4 we create sub-swarms containing at most 4 particles within each level of hierarchy, the nos. of sub-swarms in a level of hierarchy depends on the nos. of particles in that particular level. Details of this algorithm can also be found in [28].

## 5. EXPERIMENTAL DATA

In this paper, three instantiations of the circular antenna array design problem are solved by using the HDLPSO algorithm with four other state-of-the-art metaheuristics, namely, PSO [14], real coded GA [15], DE and IWO [30]. This DE variant is called DE/rand/1/bin and is the most widely used one in DE literature [31]. In [28], it is proved that the combination of HPSO and DLPSO, significantly improves the results over the results obtained using each of the algorithms separately. So, we have not included the results obtained using HPSO and DLPSO in case of this problem.

### 5.1. Problem Description

The three instantiations of the design problem are

**Table 1.** Design problems.

| Problem No. | Nos. of Array Elements | FNBW |
|---|---|---|
| 1 | 8 | 70.27 |
| 2 | 10 | 55.85 |
| 3 | 12 | 46.26 |

The FNBW is assumed to be a constant, corresponding to a uniform circular array with a uniform $0.5\lambda$ spacing between the elements. To meet the requirements of practical considerations, normalization is done for the current amplitudes, with maximum value of the amplitude being set equal to 1.

## 5.2. Parametric Setup

At first, we need to choose proper values of the $a_i$s for successful use of optimization algorithms. According to [14], it is preferable to select the weights $a_i$ for $i = 1, 2, 3$ the first three as 1, 1, and 1 respectively. $a_4$ was not considered in [14]. However, a higher weight to the component $f_{SLA}$ which deals with the minimization of the average side-lobe level may be advantageous for the overall design purpose. This should not affect the attainment of null at the null points. A uniform circular array with the same number of elements is shown to have poorer value of SLL in comparison in [13, 14]. However, using $F_4$ and taking the values of $a_i$s as 1.5, 3, 2 and 1 for $i$ = 1, 2, 3 and 4 respectively, we have achieved best results. Those values are chosen based on empirical observation and knowledge of radiation pattern. In this cost function, the independent parameters are the current amplitudes and the distances between the elements.

**Table 2.** Parametric setup for the contestant algorithms ($r_d$ is the difference between the maximum and minimum values of the $d$-th decision variable).

| HDLPSO | | Modified IWO | | DE | | PSO | | GA | |
|---|---|---|---|---|---|---|---|---|---|
| Param. | Val. | Param. | Val. | Param. | Val. | Param. | Val. | Param. | Val. |
| swarm size | 50 | swarm size | 50 | $Np$ | 10*$dim$ | swarm size | 50 | Pop_size | 150 |
| $C_1$ | 2.0 | $C_1$ | 2.0 | $CR$ | 0.90 | $C_1$ | 2.0 | Crossover Probability $P_c$ | 1.0 |
| $C_2$ | 2.0 | $C_2$ | 2.0 | $F$ | 0.50 | $C_2$ | 2.0 | Mutation Probability $P_m$ | 0.1 |
| Inertial Weight $w$ | 0.60 | Inertial Weight $w$ | 0.60 | | | Inertial Weight $w$ | 0.60 | | |
| $v_{d,\max}$ | 0.9*$r_d$ | $v_{d,\max}$ | 0.6*$r_d$ | | | $v_{d,\max}$ | 0.9*$r_d$ | | |
| | | Maximum standard deviation $sd_{\min}$ | 0.001 | | | | | | |
| | | $pow$ | 2 | | | | | | |

The control parameters for modified HDLPSO [28], IWO [30], PSO [14], real coded GA [15], and DE [31] were set after performing a series of hand tuning experiments. The parameters for PSO, and GA were set following the guidelines provided in [14, 15]. The parametric setups for all the algorithms are shown in Table 2.

### 5.3. Results

A comparison of the final values of optimizing function along with standard deviations for HDLPSO, IWO, PSO, GA and DE based approaches is presented in Table 3. Table 4 presents the optimized values of $d_i$ in terms of wavelength and normalized $I_i$.

It is a well known fact that a stochastic optimization algorithm does not yield same results over repeated runs on the same problem. So, we report the mean and the standard deviation of the best-of-run values for 50 independent runs of each of the five algorithms. A non-parametric statistical test called Wilcoxon's rank sum test for

**Table 3.** Final cost function values obtained with the five algorithms over three design instances.

| Number of Elements | Algorithm | Mean Cost Function Value | Standard Deviation |
|---|---|---|---|
| 8 | HDLPSO | **3.1810** | **0.0098** |
| | IWO | 3.3682 | 0.0167 |
| | DE | 3.2459 | 0.1181 |
| | PSO | 3.3571 | 0.6470 |
| | GA | 4.3922 | 0.8941 |
| 10 | HDLPSO | **3.5619** | **0.0121** |
| | IWO | 3.7698 | 0.0294 |
| | DE | 3.8511 | 0.2688 |
| | PSO | 3.8825 | 0.8990 |
| | GA | 5.7319 | 0.9633 |
| 12 | HDLPSO | **4.1109** | **0.0490** |
| | IWO | 4.3357 | 0.0925 |
| | DE | 4.5431 | 0.3758 |
| | PSO | 4.5590 | 0.8883 |
| | GA | 6.0938 | 0.9847 |

**Table 4.** Design variables obtained with modified IWO algorithm.

| Number of Elements | FNBW | $d_i$ in terms of wavelength | Normalized $I_n$ |
|---|---|---|---|
| 8 | 70.27 | 0.3248 0.6236<br>0.1967 0.7779<br>0.5759 0.8195<br>0.8066 0.3012 | 0.9184 0.2643<br>0.5304 0.9895<br>0.9895 0.5475<br>0.9739 0.2272 |
| 10 | 55.85 | 0.3654 0.7322<br>0.7595 0.6574<br>0.3192 0.4231<br>0.7499 0.7492<br>0.7666 0.3181 | 0.8215 0.9039<br>0.8394 0.6266<br>0.9626 0.9951<br>0.8803 0.9140<br>0.8554 0.8374 |
| 12 | 46.26 | 0.2913 0.8620<br>0.6753 0.5792<br>0.9391 0.2230<br>0.3665 0.8261<br>0.7115 0.5989<br>0.9201 0.2888 | 0.9257 0.6177<br>0.9675 0.5981<br>0.6487 0.9294<br>0.9197 0.5394<br>1.0181 0.6251<br>0.9154 0.8622 |

**Table 5.** $P$-values obtained with Wilcoxon's rank sum test comparing the best-performing algorithm with all other contestants on three design instances.

| Number of Elements | Algorithm | $P$-Value |
|---|---|---|
| 8 | HDLPSO/IWO | **1.7563e-005** |
| | HDLPSO/DE | **2.4196e-009** |
| | HDLPSO/PSO | **9.8523e-006** |
| | HDLPSO/GA | **9.8523e-008** |
| 10 | HDLPSO/IWO | **3.2431e-006** |
| | HDLPSO/DE | **8.4211e-007** |
| | HDLPSO/PSO | **1.1129e-012** |
| | HDLPSO/GA | **1.0971e-011** |
| 12 | HDLPSO/IWO | **4.4152e-004** |
| | HDLPSO/DE | **7.9821e-009** |
| | HDLPSO/PSO | **9.7536e-007** |
| | HDLPSO/GA | **1.8262e-006** |

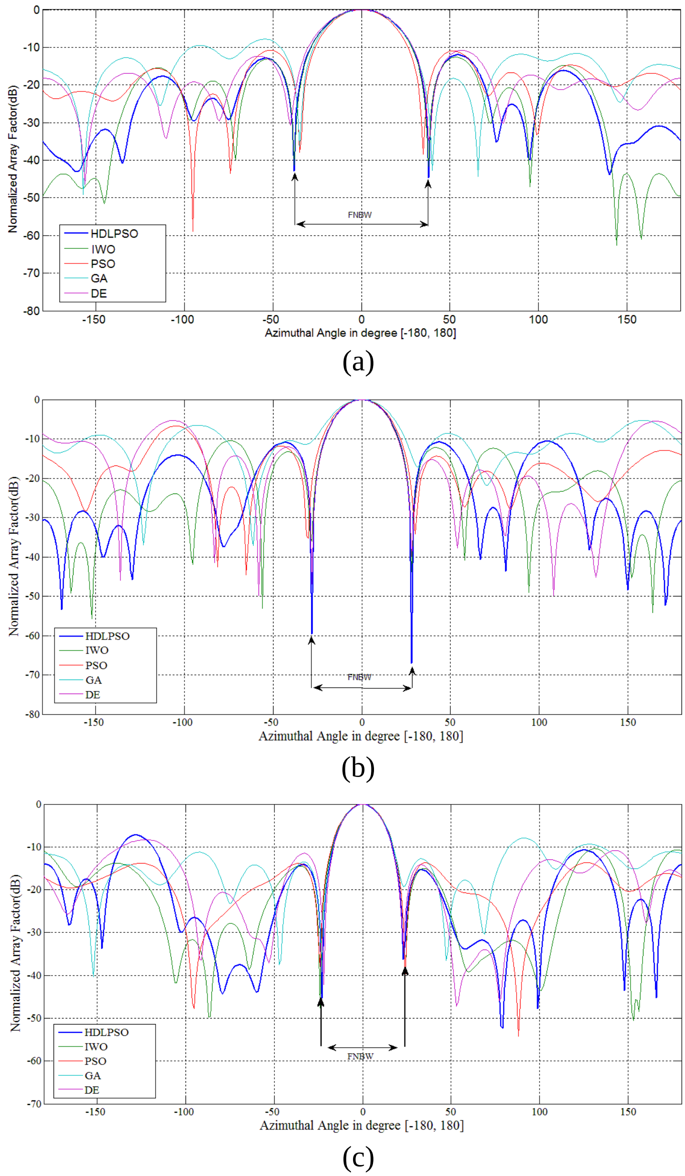


**Figure 4.** Normalized radiation patterns for circular arrays of different number of elements obtained using five different optimization techniques. (a) For number of elements $N = 8$. (b) For number of elements $N = 10$. (c) For number of elements $N = 12$.

independent samples [32, 33] is conducted at the 5% significance level in order to judge the results in a statistically significant way. $P$ values obtained through the rank sum test between the best algorithm and each of the other algorithms over the three design instances are presented in Table 5. In this table, N/A stands for *Not Applicable* and occurs for the best performing algorithm itself in each case.

In Table 6, the best results obtained (out of 50 independent runs) for the aforesaid three problem instances are judged in terms of — the average SLL (in decibels), the directivity (in decibels), and the circumference of the circular array (in terms of wavelength) for all the five algorithms based approaches. Figure 4 depicts the radiation patterns of the circular antenna arrays (corresponding to best of the 50 runs in each case) obtained with all five algorithms for 8, 10, and 12 element arrays.

From Tables 3, 4, 5 and 6, we can clearly state that HDLPSO is much better in a statistically significant way than the other four population-based meta-heuristics namely IWO, GA, PSO, and DE on this specific problem of circular array design. From Table 3, it can be clearly noted that for all three instances of design problem HDLPSO maintains smallest standard deviation of the results indicating its

**Table 6.** Design figures of merit obtained in the best (out of 50) run of the four algorithms on three design instances.

| Number of Elements | Algorithm | SLL in decibels | Directivity in decibels | Circumference (in terms of wavelength) |
|---|---|---|---|---|
| 8 | HDLPSO | **−20.313** | **9.915** | **4.4262** |
| | IWO | −18.860 | 9.761 | 4.4499 |
| | DE | −19.045 | 9.752 | 4.4509 |
| | PSO | −18.151 | 9.744 | 4.4931 |
| | GA | −14.754 | 8.299 | 4.5244 |
| 10 | HDLPSO | **−19.861** | **11.102** | **5.8406** |
| | IWO | −19.772 | 10.782 | 5.8951 |
| | DE | −19.732 | 10.755 | 5.8993 |
| | PSO | −19.582 | 10.744 | 5.9029 |
| | GA | −10.412 | 8.062 | 6.0886 |
| 12 | HDLPSO | **−19.813** | **12.212** | 7.2825 |
| | IWO | −19.776 | 11.308 | **7.1480** |
| | DE | −18.608 | 11.300 | 7.1495 |
| | PSO | −18.658 | 11.298 | 7.1501 |
| | GA | −11.628 | 9.223 | 7.7700 |

greatest effectiveness on the problem at hand.

In Figure 4, it can be clearly observed that the normalized array factor for values of the independent parameters obtained with HDLPSO has better side-lobe suppression than those of IWO, DE, PSO, and GA. Although the magnitude of the array factor at certain points of the azimuth angle range for PSO, DE or IWO is lower than the corresponding values for the case of HDLPSO as observed in Figure 4, the average side-lobe level is much lower for the latter as demonstrated in Table 6. A scrutiny of Table 6 indicates that the HDLPSO yields better values of three important factors — the SLL, directivity and the circumference in comparison to IWO, GA, PSO, and DE for all the cases.

## 6. CONCLUSION

Designing circular antenna arrays with minimum SLL, maximum directivity, and also minimum size of the circumference is one of the most challenging optimization problems in electromagnetism. In this article, we proposed an improved variant of a recently developed, ecologically inspired meta-heuristic algorithm called HDLPSO and the superiority of the proposed technique over four other state-of-the-art stochastic optimizers is demonstrated through simulation experiments in the context of three instances of the circular antenna array design problem. We formulated the design problem as an optimization task on the basis of a cost function that takes care of the average side lobe levels, the null control, and the circumference of the array. Our simulation experiments, shown in Table 3 to Table 6 and Figure 4, clearly indicated that the HDLPSO outperforms PSO, IWO, DE, and GA over 8, 10, and 12 element array design problems based on metrics such as average final accuracy, best obtained design figures of merit (like SLL, directivity, circumference size in terms of wavelength), convergence speed, and robustness, in a statistically significant manner. All these factors together have been considered for optimal results in our design problem and these accounts for the significance of this work.

Our future research will be focused upon exploration of the design problems of other array geometries and concentric circular arrays with PSO and its variants. Also, if we treat all four different components of the cost function given in (9) as separate objective functions, a multi-objective optimization, after incorporating some problem-specific expert's for pointing out the best solution from the Pareto-optimal set, may prove to be a significant avenue of future investigation.

## REFERENCES


1. Godara, L. C., Ed., *Handbook of Antennas in Wireless Communications*, CRC, Boca Raton, FL, 2002.
2. Chandran, S., Ed., *Adaptive Antenna Arrays: Trends and Applications*, Springer, 2004.
3. Tsoulos, G. V., Ed., *Adaptive Antennas for Wireless Communications*, IEEE Press, Piscataway, NJ, 2001.
4. Udina, A., N. M. Martin, and L. C. Jain, "Linear antenna array optimization by genetic means," *Third International Conference on Knowledge-Based Intelligent Information Engineering Systems Adelaide*, Australia, Sept. 1999.
5. Cengiz, Y. and H. Tokat, "Linear antenna array design with use of genetic, memetic and tabu search optimization algorithms," *Progress In Electromagnetics Research C*, Vol. 1, 63–72, 2008.
6. Weng, W.-C., F. Yang, and A. Z. Elsherbeni, "Linear antenna array synthesis using Taguchi's method: A novel optimization technique in electromagnetics," *IEEE Transactions on Antennas and Propagation*, Vol. 55, No. 3, 723–730, Mar. 2007.
7. Ares-Pena, F. J., A. Rodriguez-Gonzalez, E. Villanueva-Lopez, and S. R. Rengarajan, "Genetic algorithms in the design and optimization of antenna array patterns," *IEEE Transactions on Antennas and Propagation*, Vol. 47, 506–510, Mar. 1999.
8. Tian, Y. B. and J. Qian, "Improve the performance of a linear array by changing the spaces among array elements in terms of genetic algorithm," *IEEE Transactions on Antennas and Propagation*, Vol. 53, 2226–2230, Jul. 2005.
9. Khodier, M. M. and C. G. Christodoulou, "Linear array geometry synthesis with minimum side lobe level and null control using particle swarm optimization," *IEEE Transactions on Antennas and Propagation*, Vol. 53, No. 8, Aug. 2005.
10. Dessouky, M., A. Sharshar, and Y. A. Albagory, "Efficient sidelobe reduction technique for small-sized concentric circular arrays," *Progress In Electromagnetics Research*, Vol. 65, 187–200, 2006.
11. Gurel, L. and O. Ergul, "Design and simulation of circular arrays of trapezoidal-tooth log-periodic antennas via genetic optimization," *Progress In Electromagnetics Research*, Vol. 85, 243–260, 2008.
12. Dessouky, M., H. Sharshar, and Y. Albagory, "A novel tapered beamforming window for uniform concentric circular arrays," *Journal of Electromagnetic Waves and Applications*, Vol. 20, No. 14, 2077–2089, 2006.

13. Panduro, M., A. L. Mendez, R. Dominguez, and G. Romero, "Design of non-uniform circular antenna arrays for side lobe reduction using the method of genetic algorithms," *Int. J. Electron. Commun. (AEU)*, Vol. 60, 713–717, 2006.
14. Shihab, M., Y. Najjar, N. Dib, and M. Khodier, "Design of non-uniform circular antenna arrays using particle swarm optimization," *Journal of Electrical Engineering*, Vol. 59, No. 4, 216–220, 2008.
15. Panduro, M. A., C. A. Brizuela, L. I. Balderas, and D. A. Acosta, "A comparison of genetic algorithms, particle swarm optimization and the differential evolution method for the design of scannable circular antenna arrays," *Progress In Electromagnetics Research B*, Vol. 13, 171–186, 2009.
16. Chen, T. B., Y. L. Dong, Y. C. Jiao, and F. S. Zhang, "Synthesis of circular antenna array using crossed particle swarm optimization algorithm," *Journal of Electromagnetic Waves and Applications*, Vol. 20, No. 13, 1785–1795, 2006.
17. Khodier, M. M. and M. Al-Aqeel, "Linear and circular array optimization: A study using particle swarm intelligence," *Progress In Electromagnetics Research B*, Vol. 15, 347–373, 2009.
18. Du, K. L., "Pattern analysis of uniform circular array," *IEEE Transactions on Antennas and Propagation*, Vol. 52, No. 4, 1125–1129, 2004.
19. Mandal, D., A. Bhattacharjee, and S. Ghoshal, "Comparative optimal designs of non-uniformly excited concentric circular antenna array using evolutionary optimization techniques," *2009 2nd International Conference on Emerging Trends in Engineering and Technology (ICETET)*, 619–624, Dec. 2009.
20. Mandal, D., A. Bhattacharjee, and S. Ghoshal, "A novel particle swarm optimization based optimal design of threering concentric circular antenna array," *Advances in Computing, Control, Telecommunication Technologies*, 385–389, Dec. 2009.
21. Mandal, D., S. Ghoshal, and A. Bhattacharjee, "Improved swarm intelligence based optimal design of concentric circular antenna array," *Applied Electromagnetics Conference (AEMC),* 1–4, Dec. 2009.
22. Benedetti, M., R. Azaro, D. Franceschini, and A. Massa, "PSO-based realtime control of planar uniform circular arrays," *IEEE Transactions on Antennas and Propagation*, Vol. 5, 545–548, 2006.
23. Haupt, R., "Optimized element spacing for low sidelobe concentric ring arrays," *IEEE Transactions on Antennas and Propagation*, Vol. 56, No. 1, 266–268, Jan. 2008.

24. Kennedy, J. and R. C. Eberhart, "Particle swarm optimization," *Proc. IEEE Conf. Neural Networks IV*, Piscataway, NJ, 1995.
25. Eberhart, R. C. and Y. Shi, "Particle swarm optimization: Developments, applications and resources," *Proc. 2001 Congr. Evolutionary Computation*, Vol. 1, 2001.
26. Liang, J. J. and P. N. Suganthan, "Dynamic multi-swarm particle swarm optimizer," *Proc. Swarm Intell. Symp.*, 124–129, Jun. 2005.
27. Janson, S. and M. Middendorf, "A Hierarchical particle swarm optimizer and its adaptive variant," *IEEE Transactions on Systems, Man, and Cybernetics — Part B: Cybernetics*, Vol. 35, No. 6, Dec. 2005.
28. Ghosh, P., H. Zafar, S. Das, and A. Abraham, "Hierarchical dynamic neighborhood based particle swarm optimization for global optimization," *Proceedings of IEEE Congress on Evolutionary Computation (CEC)*, 757–764, New Orleans, Jun. 5–8, 2011.
29. Clerc, M. and J. Kennedy, "The particle swarm-explosion, stability, and convergence in a multidimensional complex space," *IEEE Trans. Evol. Comput.*, Vol. 6, No. 1, 58–73, Feb. 2002.
30. Mehrabian, R. and C. Lucas, "A novel numerical optimization algorithm inspired from weed colonization," *Ecological Informatics*, Vol. 1, 355–366, 2006.
31. Price, K., R. Storn, and J. Lampinen, *Differential Evolution — A Practical Approach to Global Optimization*, Springer, Berlin, 2005.
32. Wilcoxon, F., "Individual comparisons by ranking methods," *Biometrics*, Vol. 1, 80–83, 1945.
33. García, S., D. Molina, M. Lozano, and F. Herrera, "A study on the use of non-parametric tests for analyzing the evolutionary algorithms' behavior: A case study on the CEC'2005 special session on real parameter optimization," *Journal of Heuristics*, 2009, DOI: 10.1007/s10732-008-9080-4.