Progress In Electromagnetics Research, Vol. 125, 439–457, 2012

A MODIFIED DIFFERENTIAL EVOLUTION ALGORITHM FOR SHAPED BEAM LINEAR ARRAY ANTENNA DESIGN

A. Mandal[1,*], H. Zafar[1], S. Das[2], and A. Vasilakos[3]

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

[2]Electronics and Communication Sciences Unit, Indian Statistical Institute, Kolkata 700108, India

[3]Department of Computer and Telecommunications Engineering, University of Western Macedonia, Greece

Abstract—With the advancement of technology, the need of antenna arrays with shaped power patterns increases day by day for the purpose of improvement of communication. In this article, we represent a new method for designing optimized linear array with shaped beam radiation pattern of desired specifications. The main objective is to obtain suitable current excitation amplitude and phase distribution for the linear array elements so that it can produce the desired custom shaped radiation pattern as the user demands. The design procedure utilizes an improved variant of a prominent and efficient metaheuristics of current interest, namely the Differential Evolution (DE). In our modified DE algorithm, denoted as DE_rBM_2SX, new mutation and crossover strategies are employed. These modifications help to overcome some drawbacks of classical DE. Two examples of linear array with shaped radiation pattern design problem are considered to illustrate the effectiveness of our algorithm. Our results are also compared with two state-of-the-art variants of DE and Particle Swarm Optimization (PSO) — namely JADE and CLPSO (Comprehensive Learning Particle Swarm Optimization). The comparison clearly reveals that our optimization algorithm is more efficient than JADE or CLPSO in finding optimum element excitation amplitude and phase distribution for the desired shaped pattern.

Received 24 November 2011, Accepted 23 January 2012, Scheduled 6 March 2012

* Corresponding author: Ankush Mandal (ankushmandal19@gmail.com).

1. INTRODUCTION

The performance of a single antenna element is limited due to lack of high directivity, desired side lobe level is not always achieved, narrow beam width and uncontrollable null positions being the other problems. These problems can be addressed using suitable configuration of an array of antenna elements. Antenna arrays play an important role in detecting and processing signals arriving from different directions. By controlling various radiation characteristics antenna array can improve the performance of broadcasting system. Wide usage of antenna arrays in mobile, wireless, satellite and radar communications systems for the improvement of signal quality, increment of system coverage, maximization of spectral efficiency, capacity and link quality requires efficient design of the geometry of the antenna array. The objective in antenna array geometry synthesis is to determine the physical layout of the array to produce a radiation pattern as similar as possible to the desired pattern. Depending on the application the shape of the desired pattern can vary widely. Many antenna array synthesis techniques for the synthesis of shaped patterns can be found in the literature [1–6]. Many synthesis methods focus on suppressing the Side Lobe Level (SLL) while preserving the gain of the main beam [7]. Radiation patterns having minimal beam width and side lobe level not only fulfill the high gain demand of the wireless communication systems but also ensure the reduction of the inter-channel interferences. Some other methods take the help of null control to subdue the effects of undesired interference and jamming. Array pattern nulling techniques play a crucial role for the prevention of degradation in signal-to-noise ratio performance. In the design of linear antenna array it is possible to curb side lobe level while keeping the gain of the main beam unchanged. Array pattern can be controlled using optimizing configuration of current excitation amplitude of the array elements. Other methods include the designing of spacing between the elements keeping a uniform excitation over the array aperture, phased arrays [8].

The classical derivative-based optimization techniques require a starting point that is reasonably close to the final solution, or they are prone to be stuck in a local minimum whenever the initial values lie in a region of minimum. For complex optimization problems the gradient based methods tend to not reach the global optimum. The inherent shortcomings of the classical numerical methods have compelled the researchers all over the world to depend on metaheuristic algorithms that simulate some natural phenomena. These evolutionary metaheuristic algorithms have been used to solve antenna problems accurately using an objective function that suitably takes care of side lobe level suppression and desired main beam pattern required to produce the custom shaped pattern. Genetic algorithms (GA), a class of global optimization technique have been successfully used in the design of antenna arrays [9–12]. [13] uses memetic and tabu search algorithms along with GA for the design of linear antenna array. Simulated Annealing has been used in [14] for the antenna array design. Another efficient stochastic search process, particle swarm optimization (PSO) has also been used for the design of linear array [8]. Gies and Rahmat-Samii [15] used PSO in theta domain for the design of a phase-differentiated reconfigurable array. An adaptive Boolean PSO is developed by Zaharis and Yioultsis [16] for beam-forming on linear antenna array. Liu et al. in [17] suggested a new method for the synthesis of linear array with shaped power pattern. Guney and Basbug used Bacterial Foraging algorithm (BFA) for interference suppression of linear array [18]. Clonal selection algorithm and Bees algorithm have been used in [19] and [20] respectively for the linear array design. [21] and [22] use Ant colony optimization for array design by varying amplitude and phase. Some other antenna array design uses Invasive weed optimization [23], Taguchi's method [24], Firefly and artificial bees colony algorithm [25], FDTD method [26], plant growth simulation algorithm [27], etc. Recently a prominent and efficient metaheuristic Differential Evolution (DE) [28] has been applied for the design of effective antenna array [29, 30]. There are also different approaches besides these optimization based approaches which are of great relevance, such as [31, 32].

Differential Evolution (DE) has emerged as one of the most powerful tools for solving the real world optimization problems. Since its inception in 1996 it has been used to solve diverse class of optimization problems. Different improved variants of original DE have been suggested that yield better result in different situations. In this article we propose an improved variant of DE, named DE_rBM_2SX, for designing optimized linear array with shaped beam radiation pattern of desired specifications. The main objective is to obtain suitable current excitation amplitude and phase distribution for the linear array elements so that it can produce the desired custom shaped radiation pattern as the user demands. We also compared our results with the results obtained using two state-of-the-art EAs, namely JADE [33] and CLPSO [34]. The results clearly reveal that our method of designing the linear array is far better than the other two methods.

The paper is organized in the following way. Section 2 provides a brief overview of classical DE. Section 3 introduces the proposed DE_rBM_2SX algorithm and also describes the modifications used over classical DE to improve the efficiency. A formulation of the array pattern synthesis as an optimization task along with the objective function has been discussed in Section 4. Section 5 gives the parameter settings and values used in the experiment. Linear array synthesis examples and results have been discussed and presented in Section 6. Section 7 finally concludes the paper and unfolds a few important future research issues.

2. CLASSICAL DIFFERENTIAL EVOLUTION ALGORITHM

An iteration of the classical DE algorithm consists of the four basic steps — initialization of a population of search variable vectors, mutation, crossover or recombination, and finally selection. The last three steps are repeated generation after generation until a stopping criterion is satisfied.

2.1. Initialization of the Population

If DE searches for the global optima within the continuous search space of dimensionality D then it begins with an initial population of target vectors $\vec{X} = \{x_i^1, x_i^2, \ldots, x_i^D\}$ where $i = 1, 2, 3 \ldots NP$ (NP is the population size). The individuals of the initial population are randomly generated from a uniform distribution within the search space which has maximum & minimum bounds as follows: $\vec{X}_{\max} = \{x_{\max}^1, x_{\max}^2, \ldots, x_{\max}^D\}$ and $\vec{X}_{\min} = \{x_{\min}^1, x_{\min}^2, \ldots, x_{\min}^D\}$. The jth component of the ith individual is initialized as follows:

$$x_{i,0}^j = x_{\min}^j + \mathrm{rand}_i^j(0,1)\left(x_{\max}^j - x_{\min}^j\right); \quad j \in [1, D] \tag{1}$$

Here $\mathrm{rand}_i^j\,(0, 1)$ is a uniformly distributed random number in (0, 1) and it is instantiated independently for the j-th component of the i-th individual.

2.2. Mutation Operation

After the initialization, DE evolves the population by three operations: mutation, crossover & selection. This process is usually labeled as DE/x/y/z, where x denotes the method of selection of base vectors for mutation, y denotes the number of differential vectors used to construct the mutant vector, and z denotes the crossover type (bin: for binomial, exp: for exponential).

In each generation DE creates a *mutant vector* (also known as *donor vector*) corresponding to each individual or target vector of the current population. Three very common mutation strategies are described as follows:

a) DE/rand/1: $$\vec{V}_{i,G} = \vec{X}_{r1,G} + F \cdot \left(\vec{X}_{r1,G} - \vec{X}_{r3,G}\right) \tag{2}$$

b) DE/best/1: $$\vec{V}_{i,G} = \vec{X}_{\text{best},G} + F \cdot \left(\vec{X}_{r1,G} - \vec{X}_{r2,G}\right) \tag{3}$$

c) DE/current-to-best/1:
$$\vec{V}_{i,G} = \vec{X}_{i,G} + F_{\text{best}} \cdot \left(\vec{X}_{\text{best},G} - \vec{X}_{i,G}\right) + F \cdot \left(\vec{X}_{r1,G} - \vec{X}_{r2,G}\right) \tag{4}$$

The indices $r1$, $r2$ and $r3$ are mutually exclusive random integers in the range $[1, NP]$, they are also different from i. These indices are generated once for each mutant vector. $\vec{X}_{\text{best},G}$ is the target vector which has the best fitness value in the population at generation G. The scaling factor F and F_{best} control the amplification of the corresponding differentiation vector.

2.3. Crossover Operation

To enhance the potential diversity of the population, a crossover operation comes into play after generating the donor vector through mutation. The donor vector mixes its components with the target vector $\vec{X}_{i,G}$ under this operation to form the trial vector $\vec{U}_{i,G} = \{u^1_{i,G}, u^2_{i,G}, \ldots, u^D_{i,G}\}$. The DE family of algorithms uses mainly two kinds of crossover methods — *exponential* (or two-point modulo) and *binomial* (or uniform). Here we shall briefly outline the binomial crossover scheme that has been used in the proposed algorithm. Under this scheme the trial vector is created as follows:

$$u^j_{i,G} = \begin{cases} v^j_{i,G} & \text{if} \quad \text{rand}(0,1) \le CR \quad \text{or} \quad j = j_{\text{rand}} \\ x^j_{i,G} & \text{otherwise} \end{cases} \tag{5}$$

where CR is a user-specified parameter (*Crossover Rate*) in the range [0, 1) and $j_{\text{rand}} \in [1, 2, \ldots, D]$ is a randomly chosen index which ensures that the trial vector $\vec{U}_{i,G}$ will differ from its corresponding target vector $\vec{X}_{i,G}$ by at least one component.

2.4. Selection Operation

To keep the population size constant over subsequent generations, the next step of the algorithm calls for *selection* to determine whether the target or the trial vector survives to the next generation, i.e., at $G = G + 1$. For maximization problem, if the objective function value of the trial vector is not less than that of the corresponding target vector, then the trial vector is selected for the next generation; and if it is not so, then the target vector is selected for the next generation. Obviously, for minimization problem the condition for selection is just the opposite. The selection operation works as follows (for maximization problem):

$$\vec{X}_{i,G} = \begin{cases} \vec{U}_{i,G} & \text{if } f\left(\vec{U}_{i,G}\right) \geq f\left(\vec{X}_{i,G}\right) \\ \vec{X}_{i,G} & \text{otherwise} \end{cases} \tag{6}$$

where $f(\cdot)$ is the objective function to be optimized.

3. THE DE_RBM_2SX ALGORITHM

In our algorithm, denoted by DE_rBM_2SX, we have done two modifications over the classical DE — a random best mutation strategy and a new crossover strategy involving two parent vectors of similar fitness values. The comparison of the results represented later will show that these modifications significantly improve the performance of DE.

3.1. Random Best Mutation Strategy

Among several mutation strategies in DE, strategies like "DE/current-to-best" and "DE/best" are greedy in nature, i.e., they converge very quickly by guiding the algorithm to the best position so far discovered. But, as a result of such exploitative tendency, in many cases, the population may lose its diversity and global exploration abilities within a relatively small number of generations, thereafter getting trapped to some locally optimal point in the search space. Taking into consideration these facts and to overcome the limitations of fast but less reliable convergence performance of DE/current-to-best/1 scheme, in this article, we propose a less greedy and more explorative variant of the DE/current-to-best/1 mutation strategy by using randomly selected individual from M top ranking individuals for each target vector. The new scheme can be expressed as:

$$\vec{V}_{i,G} = \vec{X}_{i,G} + F_{\text{best}} \cdot \left(\vec{X}_{\text{Mrand,i,G}} - \vec{X}_{i,G}\right) + F \cdot \left(\vec{X}_{r1,G} - \vec{X}_{r2,G}\right) \tag{7}$$

where $\vec{X}_{\text{Mrand},i,G}$ is the randomly selected individual from the M top ranking individuals for the i-th individual. Other notations bear the same meaning as mentioned previously in Section 2.2. Under this scheme, the target solutions are not always attracted towards the same best position found so far by the entire population and this feature is helpful in avoiding premature convergence at local optima.

3.2. New Crossover Scheme

Crossover operation is very important for any Evolutionary Algorithm because by this procedure, information gathered by the current generation is transferred to the next generation. In the classic crossover operation, the mutant vector exchanges its components with the current individual to produce the trial vector. In our new crossover scheme, the mutant vector exchanges its component with two parent vectors to produce the trial vector. One of the parent vectors is the current vector and the other one is the vector whose fitness value is the closest one to the current vector. This process allows different components transfer of different vectors of same fitness value to the next generation vector. As a consequence the exploration efficiency of the algorithm increases. As there are two parent vectors, there are also two crossover probabilities (CR). The pseudocode of the proposed crossover scheme is given below.

The flowchart of DE_rBM_2SX algorithm is given in Figure 1.

4. FORMULATION OF THE DESIGN PROBLEM

An antenna array is a configuration of individual radiating elements that are arranged in space and can be used to produce a directional radiation pattern. In our design problem we consider a linear antenna array. Optimal configuration of the array elements can achieve the desired radiation patterns given by user defined functions. For a linear

Let X_i be the current individual and X_{close} is the vector whose fitness value is the closest one to X_i. The mutant vector is V_{mut} and the trial vector is U_i.
Let $CR1$ and $CR2$ be two crossover probabilities set by user.
Usually $CR1$ is in the range (0, 0.5] and $CR2$ is in the range (0.5, 1).
$Randomval = \text{rand}(0,1)$
if $Randomval <= \text{CR1}\ \&\ j \neq j_{\text{rand}}$ ***then***
$U_{i,j} = X_{i,j}$
else if $Randomval >= CR2\ \&\ j \neq j_{\text{rand}}$ ***then***
$U_{i,j} = X_{\text{close},j}$
else
$U_{i,j} = V_{\text{mut},j}$
end
Here subscript j means the jth component of the corresponding vector. All other notations bear the same meaning as mentioned in the classical crossover operation (Section 3.3).

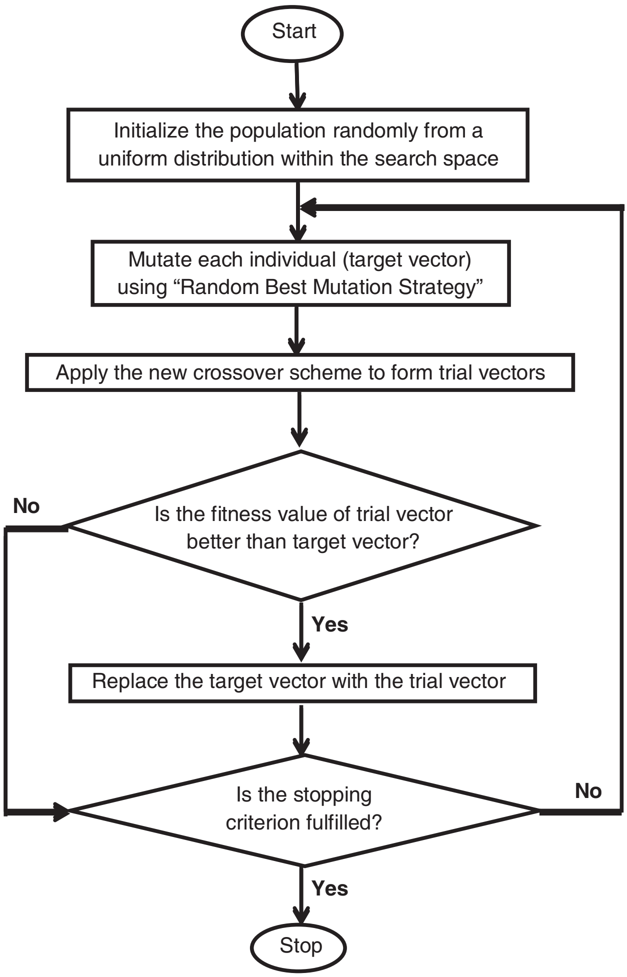

Figure 1. Flowchart of DE_rBM_2SX algorithm.

antenna array with N elements, separated by a uniform distance d, the array factor is given by,

$$AF(\theta) = \sum_{m=1}^{N} I_m e^{j2\pi md \sin\theta/\lambda}. \tag{8}$$

In the above equation I_m represents the current amplitude of mth element. The angle from the normal to the array axis is denoted by θ.

λ is the wavelength. The normalized array factor can be obtained as:

$$AF(\theta) = \frac{1}{AF_{\max}} \sum_{m=1}^{N} I_m e^{j2\pi md \sin\theta/\lambda}, \tag{9}$$

$AF_{\max}$ being the maximum value of the magnitude of the array factor. We assume $d = \lambda/2$ in this design context and the desired radiation pattern is achieved by optimizing the current amplitude and phase coefficients of the array elements. Here we have used a simple but very effective way to evaluate a particular element excitation amplitude distribution. For the radiation pattern in the main beam area, we have calculated the mean square deviation from the desired pattern and add this value to the fitness function. For the $SLLs$, we have to calculate how much greater is the SLL at a point than the desired level. Then we sum up these values for all the $SLLs$ and add it to the fitness function. Actually, in case of $SLLs$, we set a "do not exceed" scenario. The fitness function can be represented as:

$$\begin{aligned}\text{Fitness} &= \text{Mean Square Deviation in the Main Beam}\\ &+\sum(SLL_O - LL_D)\cdot H(SLL_O - SLL_D) + \sum(FNBW_0 - FNBW_D)^2\end{aligned} \tag{10}$$

$$\text{Mean Square Deviation in the Main Beam} = \frac{\sum_{i=1}^{M}(NRP_D - NRP_O)^2|_{\theta=\theta_i}}{M}, \tag{11}$$

where $FNBW$ denotes the first null beam width, NRP denotes the Normal Radiation Pattern, subscript O denotes the obtained value and D denotes the desired value. M is the number of sampling points of $\theta\,(-90^\circ \leq \theta \leq 90^\circ)$, we set it to 360. However, if we consider a main beam to be a pencil beam then the "*Mean Square Deviation in the Main Beam*" term is excluded because it is a natural shape for the main beam, not a customized shape. $H(\ )$ denotes the Heaviside step function. Heaviside step function $H(T)$ can be expressed as follows:

$$H(T) = \begin{cases} 0 & \text{if} \quad T < 0 \\ 1 & \text{if} \quad T \geq 0 \end{cases} \tag{12}$$

Our objective is to minimize the fitness function.

5. PARAMETER SETTINGS

In this section, we have represented the parameter values used for the experiment. The parameter values and settings are given below:

1) NP was kept fixed at 60 throughout the search process.
2) For the mutation process in DE_rBM_2SX, M was set to 15.
3) In this algorithm, scaling factor (F) for each dimension of the difference vector is generated randomly depending on the value of the difference vector along the corresponding dimension. F is generated independently for each dimension of the difference vector. Scaling factor generation can be expressed as follows:

$$F_j^d = \text{rand}(0,1) \cdot e^{-|x_j^d|/|x_R^d} \tag{13}$$

where $d \in [1, D\,(\text{Dimension of the search space})]$. We are generating scaling factor for the dth dimension of the jth individual. x_j^d is the value of the difference vector along dth dimension, x_R^d is the search range along that dimension.
4) $CR1$ was set to 0.1 and $CR2$ was set to 0.9.

Parameter settings for JADE were as follows: $p = 0.05$ and $c = 0.1$ and $NP = 60$. Details about the parameters can be found in [33].

Parameter settings for CLPSO were as follows: population size = 60. w and Pc_i were set as given in [34].

6. LINEAR ARRAY SYNTHESIS EXAMPLES

For all the examples, the array element excitation amplitude is in the range of $0 \le I \le 1$ and the phase is in the range of $-90° \le \theta \le 90°$.

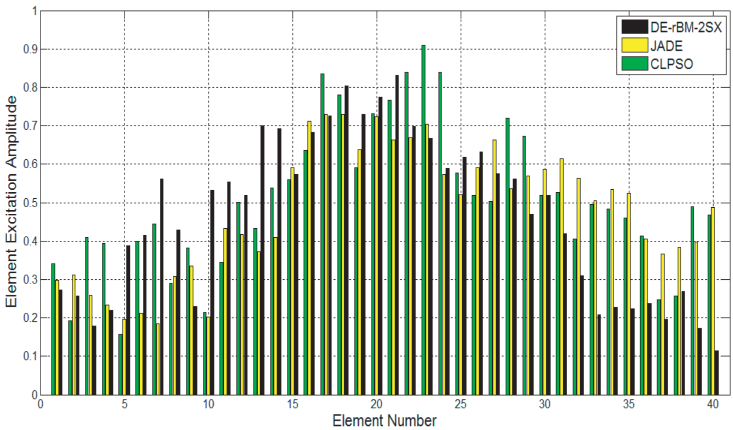

Figure 2. Element excitation amplitude distributions for first problem.

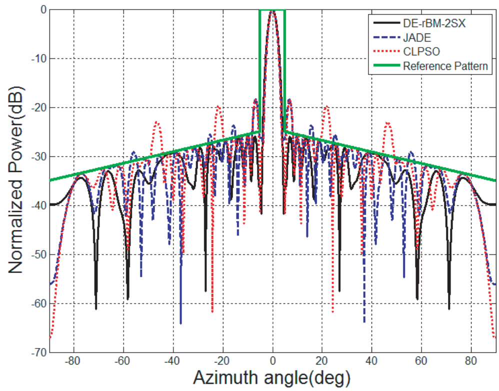

Figure 3. Normalized power patterns for first problem.

The accuracy in the value is measured up to 4 decimal places. For the first example, we have considered a linear antenna array of 40 elements. As mentioned earlier, the elements of the array of consideration are uniformly spaced. To obtain the desired radiation pattern in this example, only the excitation current amplitudes are optimized by our proposed DE_rBM_2SX algorithm. Here the desired radiation pattern has the following characteristics:

1) The $FNBW$ is 10°.
2) The $SLLs$ should have values less than the tapered sidelobe masks on both side of main beam that decrease linearly from -25 dB to -35 dB. The masks start from the first null points as -25 dB and ends on $\theta = -90°$ or $90°$ as -35 dB.

Clearly, here our objective is to obtain a pattern whose $SLLs$ are less than the tapered sidelobe masks and the $FNBW$ is almost equal to 10°. In Table 1, we have represented the element excitation amplitude values obtained for the 40 elements by our design method.

Actually, in Table 1 we have represented the element excitation amplitude values for the best radiation pattern we have obtained over 20 runs. For compactness of the representation, we have not explicitly represented the element excitation amplitude values for the best radiation pattern obtained by applying JADE or CLPSO. But we have represented the element excitation amplitude distribution graph for the best radiation pattern obtained by our method (DE_rBM_2SX) in comparison with the ones obtained by applying JADE and CLPSO in Figure 2. All the algorithms are executed for 20 independent runs for finding the best radiation pattern.

Table 1. Element excitation amplitude values obtained by our design method for first problem.

1 (0.2718)	2 (0.2567)	3 (0.1784)	4 (0.2207)	5 (0.3870)
9 (0.2292)	10 (0.5328)	11 (0.5546)	12 (0.5180)	13 (0.7003)
17 (0.7265)	18 (0.8040)	19 (0.7299)	20 (0.7748)	21 (0.8318)
25 (0.6191)	26 (0.6323)	27 (0.5758)	28 (0.5607)	29 (0.4689)
33 (0.2072)	34 (0.2278)	35 (0.2228)	36 (0.2376)	37 (0.1967)

6 (0.4156)	7 (0.5607)	8 (0.4293)
14 (0.6918)	15 (0.5735)	16 (0.6821)
22 (0.6985)	23 (0.6670)	24 (0.5891)
30 (0.5185)	31 (0.4185)	32 (0.3088)
38 (0.2695)	39 (0.1724)	40 (0.1152)

Next, in Figure 3, we have represented the radiation patterns obtained from the element excitation amplitude distributions represented above.

For the second example, we have considered linear array of 50 elements. In this design problem both excitation amplitude and phase distribution of the array elements are optimized to obtain the desired radiation pattern. Here, the desired radiation pattern has the following characteristics:

1) $FNBW$ is $20°$.
2) The main lobe is tapered 0 to -10 dB.
3) The $SLLs$ should have values less than the sidelobe mask on both side of main beam. The sidelobe mask is -20 dB in the region where $\theta < 0$ and tapered from -15 dB to -20 dB where $\theta > 0$.

Clearly, here our objective is to obtain sidelobe levels less than the mask in the sidelobe region and main beam equal to the mask in the main beam region. In Tables 2, 3, we have represented respectively the element excitation amplitude and phase values obtained for the 50 elements by our design method.

As done previously in the first problem, in Tables 2 and 3 we have respectively represented the element excitation amplitude and phase values for the best radiation pattern we have obtained over 20 runs. For compactness of the representation, we have not explicitly represented the element excitation amplitude and phase values for

Table 2. Element excitation amplitude values obtained by our design method for second problem.

1 (0.0083)	2 (0.0810)	3 (0.0201)	4 (0.1087)	5 (0.0239)
9 (0.0271)	10 (0.1316)	11 (0.1706)	12 (0.0088)	13 (0.4615)
17 (0.8832)	18 (0.7250)	19 (0.9709)	20 (0.4222)	21 (0.9574)
25 (0.9963)	26 (0.9954)	27 (0.6086)	28 (0.3436)	29 (0.0011)
33 (0.5592)	34 (0.7719)	35 (0.8874)	36 (0.3300)	37 (0.6508)
41 (0.8312)	42 (0.3226)	43 (0.5186)	44 (0.0609)	45 (0.2793)
49 (0.1154)	50 (0.0487)			

6 (0.0132)	7 (0.0225)	8 (0.0545)
14 (0.5674)	15 (0.2313)	16 (0.4712)
22 (0.9971)	23 (0.9941)	24 (0.9999)
30 (0.0780)	31 (0.0086)	32 (0.7630)
38 (0.1665)	39 (0.8293)	40 (0.4575)
46 (0.1112)	47 (0.4524)	48 (0.4159)

Table 3. Element excitation phase values obtained by our design method for second problem.

1 (−38.06)	2 (45.12)	3 (−81.45)	4 (67.78)
9 (−71.76)	10 (−57.21)	11 (−54.79)	12 (63.76)
17 (−57.38)	18 (−16.19)	19 (−27.57)	20 (40.32)
25 (58.06)	26 (84.77)	27 (33.34)	28 (81.39)
33 (12.30)	34 (27.93)	35 (59.07)	36 (−0.87)
41 (23.40)	42 (1.46)	43 (78.49)	44 (52.13)
49 (86.33)	50 (54.84)		

5 (−56.89)	6 (54.78)	7 (80.81)	8 (10.59)
13 (−84.50)	14 (−61.02)	15 (35.73)	16 (−75.46)
21 (−0.91)	22 (18.67)	23 (41.94)	24 (55.04)
29 (−87.10)	30 (−89.91)	31 (88.27)	32 (−40.11)
37 (89.94)	38 (−89.56)	39 (−79.54)	40 (−32.51)
45 (−88.16)	46 (86.99)	47 (−68.32)	48 (47.85)

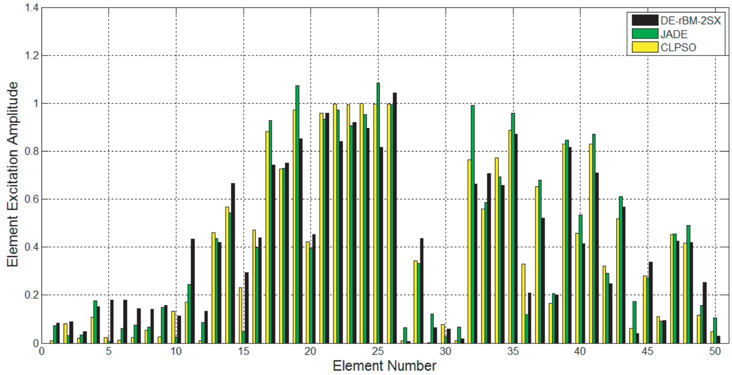

Figure 4. Element excitation amplitude distributions for second problem.

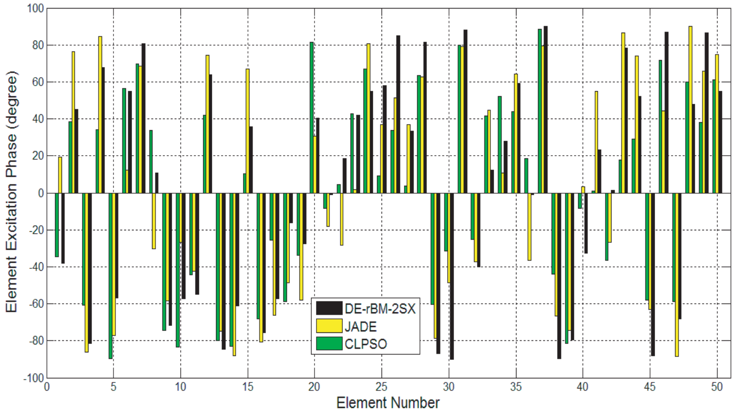

Figure 5. Element excitation phase distributions for second problem.

the best radiation pattern obtained by applying JADE or CLPSO. But we have represented the element excitation amplitude and phase distribution graph for the best radiation pattern obtained by our method (DE_rBM_2SX) in comparison with the ones obtained by applying JADE and CLPSO respectively in Figures 4 and 5. All the algorithms are executed for 20 independent runs for finding the best radiation pattern.

In Figure 6, we have represented the radiation patterns obtained from the element excitation amplitude and phase distributions obtained for the 50 element linear array by applying DE_rBM_SX, JADE and CLPSO.

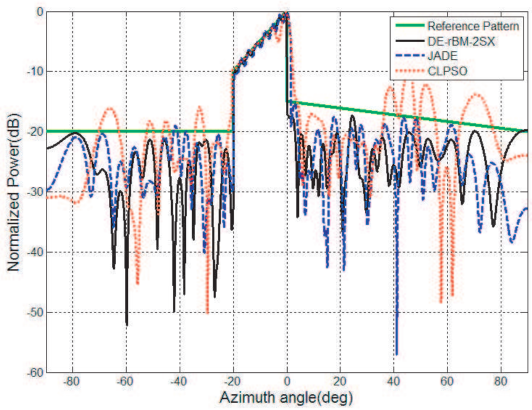

Figure 6. Normalized power patterns for second problem.

Table 4. Comparison of average CPU time required per run.

Problem	**Average CPU Time Required Per Run (in Seconds)**		
	DE_rBM_2SX	**JADE**	**CLPSO**
First Example	**2.65**	3.57	3.82
Second Example	**3.11**	4.28	4.91

To give an idea of runtime of the simulation process, we have represented the comparison of average CPU time required per run in DE_rBM_2SX, JADE and CLPSO based design methods in Table 4. We performed the simulation in the following experimental environment:

- CPU: 2.4 GHz Intel®Core™2
- RAM: 2 GB DDR2
- Language: MATLAB 7

7. CONCLUSION

Design of linear antenna array with shaped power pattern has become very important in the present context of growing need in the field electromagnetics. In this paper we introduce an improved variant of a well-known metaheuristic algorithm, Differential Evolution (DE) for designing linear antenna array with custom shaped power pattern. The new variant called DE_rBM_2SX has shown promising results regarding the design problems and to illustrate its superiority two instances of the linear antenna array design problem have been considered. The design problem was formulated as an optimization task on the basis of a cost function that takes care of the average side lobe levels and main beam pattern. The experimental results clearly indicate that the proposed DE_rBM_2SX algorithm performs much better than the other state-of-the-art algorithms, namely JADE and CLPSO.

Future research will focus on exploring the design of other array geometries using DE_rBM_2SX and improvement of the present algorithm. In future linear array design problem can be viewed as a multi-objective optimization problem but some problem-specific expert's knowledge may have to be incorporated then for pointing out the best solution from the Pareto-optimal set produced by a multi-objective optimizer.

REFERENCES

1. Milne, K., "Synthesis of power radiation patterns for linear array antennas," *IEEE Proceedings H*, Vol. 134, No. 3, 285–296, 1987.
2. Zhang, W. K. and Y.-M. Bo, "Pattern synthesis for linear equal-spaced antenna array using an iterative eigenmodes method," *IEEE Proceedings H*, Vol. 135, No. 3, 167–170, 1988.
3. Isernia, T., O. M. Bucci, and N. Fiorentino, "Shaped beam antenna synthesis problems: Feasibility criteria and new strategies," *Journal of Electromagnetic Waves and Applications*, Vol. 12, No. 1, 103–138, 1998.
4. Cid, J. M., J. A. Rodriguez, and F. Ares, "Shaped power patterns produced by equispaced linear arrays: Optimized synthesis using orthogonal $\sin(Nx)/\sin(x)$ beams," *Journal of Electromagnetic Waves and Applications*, Vol. 13, No. 7, 985–992, 1999
5. Azevedo, J. A. R., "Shaped beam pattern synthesis with non-uniform sample phases," *Progress In Electromagnetics Research B*, Vol. 5, 77–90, 2008.
6. Akdagli, A. and K. Guney, "Shaped-beam pattern synthesis of equally and unequally spaced linear antenna arrays using a modified tabu search algorithm," *Microwave Opt. Technol. Lett.*, Vol. 36, No. 1, 16–20, 2003.
7. Khodier, M. M. and C. G. Christodoulou, "Linear array geometry synthesis with minimum sidelobe level and null control using particle swarm optimization," *IEEE Transactions on Antennas and Propagation*, Vol. 53, No. 8, Aug. 2005.

8. Godara, L. C., Ed., *Handbook of Antennas in Wireless Communications*, CRC, Boca Raton, FL, 2002.
9. 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.
10. 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.
11. 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.
12. Lommi, A., A. Massa, E. Storti, and A. Trucco, "Side lobe reduction in sparse linear arrays by genetic algorithms," *Microwave Optical Techn. Lett.*, Vol. 32, 194–196, 2002.
13. 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.
14. Murino, V., A. Trucco, and C. S. Regazzoni, "Synthesis of equally spaced arrays by simulated annealing," *IEEE Trans. on Signal Process.*, Vol. 44, No. 1, 119–122, Jan. 1996.
15. Gies, D. and Y. Rahmat-Samii, "Particle swarm optimization for reconfigurable phase-differentiated array design," *Microwave and Opt. Technology Lett.*, Vol. 38, 168–175, 2003.
16. Zaharis, Z. D. and T. V. Yioultsis, "A novel adaptive beamforming technique applied on linear antenna arrays using adaptive mutated boolean PSO," *Progress In Electromagnetics Research*, Vol. 117, 165–179, 2011.
17. Liu, Y., Z.-P. Nie, and Q. H. Liu, "A new method for the synthesis of non-uniform linear arrays with shaped power patterns," *Progress In Electromagnetics Research*, Vol. 107, 349–363, 2010.
18. Guney, K. and S. Basbug, "Interference suppression of linear antenna arrays by amplitude-only control using a bacterial foraging algorithm," *Progress In Electromagnetics Research*, Vol. 79, 475–497, 2008.
19. Babayigit, B., A. Akdagli, and K. Guney, "A clonal selection algorithm for null synthesizing of linear antenna arrays by amplitude control," *Journal of Electromagnetic Waves and*

Applications, Vol. 20, No. 8, 1007–1020, 2006.
20. Guney, K. and M. Onay, "Amplitude-only pattern nulling of linear antenna arrays with the use of bees algorithm," *Progress In Electromagnetics Research*, Vol. 70, 21–36, 2007.
21. Karaboga, D., K. Guney, and A. Akdagli, "Antenna array pattern nulling by controlling both the amplitude and the phase using modified touring ant colony optimisation algorithm," *Int. J. Electronics*, Vol. 91, 241–251, 2004.
22. Hosseini S. A. and Z. Atlasbaf, "Optimization of side lobe level and fixing quasi-nulls in both of the sum and difference patterns by using continuous ant colony optimization (ACO) method," *Progress In Electromagnetics Research*, Vol. 79, 321–337, 2008.
23. Mallahzadeh, A. R., H. Oraizi, and Z. Davoodi-Rad, "Application of the invasive weed optimization technique for antenna configurations," *Progress In Electromagnetics Research*, Vol. 79, 137–150, 2008.
24. Sheng, N., C. Liao, W. Lin, L. Chang, Q. Zhang, and H. Zhou, "A hybrid optimized algorithm based on EGO and Taguchi's method for solving expensive evaluation problems of antenna design," *Progress In Electromagnetics Research C*, Vol. 17, 181–192, 2010.
25. Basu, B. and G. K. Mahanti, "Fire fly and artificial bees colony algorithm for synthesis of scanned and broad-side linear array antenna," *Progress In Electromagnetics Research B*, Vol. 32, 169–190, 2011.
26. Yang, S., Y. Chen, and Z.-P. Nie, "Simulation of time modulated linear antenna arrays using the FDTD method," *Progress In Electromagnetics Research*, Vol. 98, 175–190, 2009.
27. Guney, K., A. Durmus, and S. Basbug, "A plant growth simulation algorithm for pattern nulling of linear antenna arrays by amplitude control," *Progress In Electromagnetics Research B*, Vol. 17, 69–84, 2009.
28. Storn, R. and K. Price, "Differential evolution — A simple and efficient heuristic for global optimization over continuous spaces," *Journal of Global Optimization*, Vol. 11, No. 4, 341–359, 1997.
29. Dib, N. I., S. K. Goudos, and H. Muhsen, "Application of Taguchi's optimization method and self-adaptive differential evolution to the synthesis of linear antenna arrays," *Progress In Electromagnetics Research*, Vol. 102, 159–180, 2010.
30. Chowdhury, A., A. Ghosh, R. Giri, and S. Das, "Optimization of antenna configuration with a fitness-adaptive differential evolution algorithm," *Progress In Electromagnetics Research B*, Vol. 26,

291–319, 2010.

31. Bucci, O., G. D'Elia, G. Mazzarella, and G. Panariello, "Antenna pattern synthesis: A new general approach," *IEEE Proc.*, Vol. 82, 358–371, Mar. 1994.
32. Oliveri, G. and A. Massa, "Bayesian compressive sampling for pattern synthesis with maximally sparse non-uniform linear arrays," *IEEE Transactions on Antennas and Propagation*, Vol. 59, 467–481, Feb. 2011.
33. Zhang, J. and A. C. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. on Evol. Comput.*, Vol. 13, No. 5, 945–958, Oct. 2009.
34. Liang, J. J., A. K. Qin, P. N. Suganthan, and S. Baskar, "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Trans. on Evol. Compt.*, Vol. 10, 281–295, 2006.