PATTERN NULLING IN LINEAR ARRAYS WITH FEWER ELEMENTS BY USING DIFFERENTIAL EVOLUTION ALGORITHM

Ertuğrul Aksoy Erkan Afacan

e-mail: ertugrulaksoy@gazi.edu.tr *e-mail: e.afacan@gazi.edu.tr*

Gazi University, Faculty of Engineering and Architecture, Department of Electrical & Electronics Engineering, 06570, Maltepe, Ankara, Turkey

Key words: Differential evolution algorithm; linear antenna arrays; array thinning

ABSTRACT

In this study, differential evolution algorithm is applied to the problem of radiation pattern nulling for a linear array. The number of array elements are reduced while keeping the same desired null positions. The linear array consists of isotropic elements and is placed symmetrically on *z*-axis with inter element spacing of half wavelength. The pattern nulls are controlled with excitation amplitudes of the array elements. Several simulations are realized to illustrate the performance of the method.

I. INTRODUCTION

The problem of pattern nulling is a much studied topic in literature. As well as there are analytical solutions for the problem [1], it is shown that the evolutionary algorithms such as genetic algorithms, modified touring ant colony optimization, differential evolution algorithm, bees algorithm can produce a solution for the problem [2-7].

In analytical solution the null positions and the main beam direction are expressed as steering vectors and simultaneous solution of these vectors gives the complex weight vector [1].

In evolutionary approach, variables of the array factor are selected as optimization parameters. Controlling these variables can produce a desired radiation pattern. The most common pattern nulling techniques using evolutionary algorithms in literature can be classified as follows; controlling the excitation amplitudes of the array elements, the excitation phases of the array elements, the positions of array elements in array plane (or axis) and the element height from the array plane (or axis). Besides, these variables can be optimized stand alone, combinations of these variables can also produce the desired pattern [2-7].

One common point of all these optimization processes is that they use a fixed size array. In other words all these processes start with an N-element array and stops with an array of same size.

In this study, a method for the problem of pattern nulling is suggested. In this method the differential algorithm is used for optimization and pattern nulls are controlled with the amplitude excitations. The algorithm starts with an N-element linear array and discards some of the array elements during the optimization process. When the algorithm stops desired criteria are obtained with fewer elements than standard process. A linear array consists of zero- phased isotropic sources and is symmetrically placed on *z*-axis with inter element spacing of half-wavelength. Several simulations are made to illustrate the performance of the process.

II. THE PROBLEM

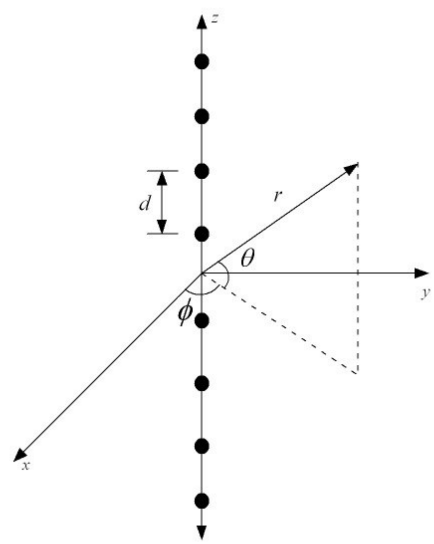

Figure 1. Geometry of the linear array

The geometry of the array is shown in Figure 1. The reference point is taken as the normal of the array. For a zero-phased symmetrically placed 2N elements along *z*-axis with equal inter element spacing of *d*, the array factor can be written as:

$$AF(\theta) = 2\sum_{n=1}^{N} a_n \cos\left[\frac{(2n-1)}{\lambda}\pi d \sin\theta\right] \qquad (1)$$

where a_n, d, λ are the excitation amplitudes of the array elements, the inter-element spacing and the operating wavelength respectively.

The problem is defined as the optimization of the excitation amplitudes with some zero magnitudes that reduce the number of array elements.

III. DIFFERENTIAL EVOLUTION ALGORITHM

Differential Evolution Algorithm (DE) is described in [8] as a direct parallel search method over continuous spaces. Like other evolutionary algorithms DE is an iterative search method and have some processes called mutation and crossover. Detailed information about DE can be found in [9], by the way the algorithm can be described as below:

- (i) Generate an initial population with uniformly distributed (unless otherwise stated) variables.
- (ii) Calculate the fitness of population using fitness function "*f*".
- (iii) Generate a mutant vector from parameter vectors by mutation.
- (iv) Generate a trial vector by crossover.
- (v) Calculate the fitness of trial vector using fitness function "*f*".
- (vi) Compare the trial vector's and *j*th parameter vector's fitness values and select a suitable vector for next generation.
- (vii) Go to step (iii) until *j* reachs the population size.
- (viii) Build new population and go to step (ii) until stop condition is fulfilled.

The fitness function used in optimization process is defined as:

$$f = \sqrt{\sum_{i=1}^{K} w_i \,|\, E_i - ND_i \,|^2 + w_m \left|SLL_{\max}\right|^2} \qquad (2)$$

Here K is the number of the interference source, E_i ($i=1,2,..,K$) is the electric field intensity at those interference source direction, ND_i is the desired null depth for the ith interference source, SLL_{max} is the maximum side-lobe level. w_i and w_m are the weight factors of each term and both of them are set to one initially. w_i weights are set to zero when the corresponding E_i is below that of ND_i and w_m is also set to zero when maximum side-lobe level of the array drops below a predefined value.

In the optimization procedure the a_n vector is selected as the parameter vector in the interval $[0,\infty)$. To maintain that a_n vector takes integer values the mutation factor is set to unity. Normalized element weights are obtained by a normalization process. Unnecessary array elements are discarded in this way.

The initial values of the a_n vector are selected as 0's or 1's with equal probability. Negative values of trial vector are set to zero and to avoid zero solution in case all of the trial vector parameters become zero the vector parameters are randomly selected as 0 or 1 with equal probability.

At first glance, it seems the search interval is not large enough so a trial-error approach may be more suitable than this method. For an unsigned 32 bit integer value each element can take a value between 0 and 2^{32}. For a 20 elements symmetrical array 10 elements give a result of 2^{320} different solution. This value is approximately equal to 2×10^{96}. So it seems that the trial-error approach is not suitable to search a result.

In the next section three examples are given to illustrate the performance of the algorithm. The population size is taken as 50, the crossover factor is taken as 0.9 and the mutation factor is set to unity, for all examples.

IV. NUMERICAL RESULTS AND DISCUSSION

In first scenario, the interference sources at 20^o, 40^o and 60^o are suppressed with a 20 elements linear array. The results for a normal optimization process executed in search interval [0,1] are given in Table 1 and radiation pattern with these excitations is given in Figure 2. This is a typical result and satisfies the desired conditions of null angles of 20^o, 40^o and 60^o, desired null depth of -90 dB and desired maximum side-lobe level of -30 dB.

When this array is optimized with new search interval of $[0,\infty)$ and new mutation factor of F=1 it is observed that some of the excitation amplitudes returned a value of zero and this means there is no element in that position. For the same scenario of null positions at 20^o, 40^o and 60^o with null depth of -90 dB and maximum side-lobe level of -30 dB two results of amplitude optimization process are given in Table 1 as Example 1 and Example 2. The results in Table 1 are normalized. Also the radiation pattern of examples are given in Figure 3 and Figure 4 respectively.

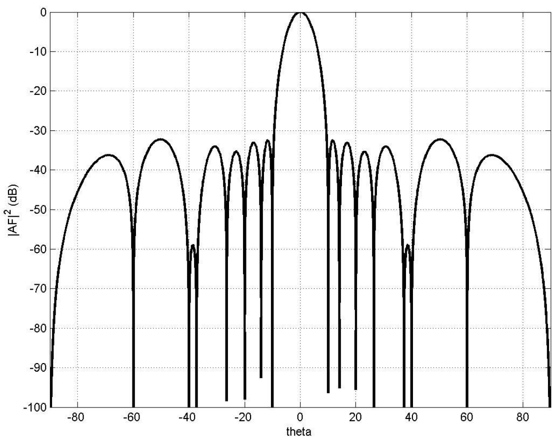

Figure 2. Radiation pattern of 20 elements linear array

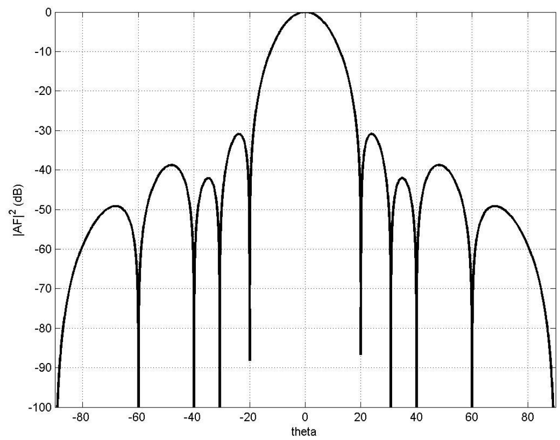

Figure 4. Radiation pattern of Example 2

It is seen from the figures that both of the thinned arrays satisfy the desired conditions. First example returns with a total element number of 12 and the second example returns with a total number of 10 elements. The 12 element antenna array has nulls at 20^o, 40^o and 60^o with depth of below -90 dB and maximum side-lobe levels below -30 dB. The 10 element antenna array have nulls at 40^o and 60^o with depth of below -90 dB and maximum side-lobe levels below -30dB; at 20^o the null depth is below -85 dB.

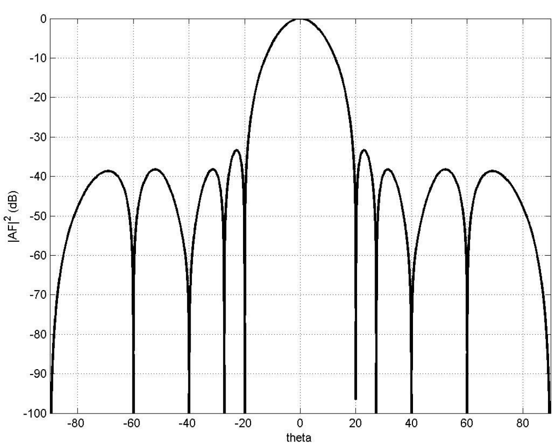

Figure 3. Radiation pattern of Example 1

In another scenario of suppressing the angles of 14^o, 23^o, 37^o, 66^o and 79^o, a maximum side-lobe level of -30 dB is required. The typical results of the optimization process are given in the second column of Table 2 and radiation pattern of the array is given in Figure 5. As expected it satisfies the desired conditions.

For second scenario, with new algorithm parameters, the algorithm returns with a total number of 16 elements and also the array satisfies the desired conditions. The normalized amplitudes of the array elements are given in third column of the Table 2 and radiation pattern is given in Figure 6.

As a result of using random functions the evolutionary algorithms never guarantee to find a solution in a time interval and also in an iteration interval. This weakness may prevent their usage in real time applications. Sticking in local maxima and convergence speed are also problems of evolutionary algorithms.

As it is seen from the results the method can handle the operation of thinning the array under predefined conditions. The sensitive parameters such as side-lobe level can be handled with the cost function which means the reduction in the array elements is not affect the side-lobe level. Also different parameters can be added to the cost function and the results will satisfy the desired parameter values if the method finds a solution.

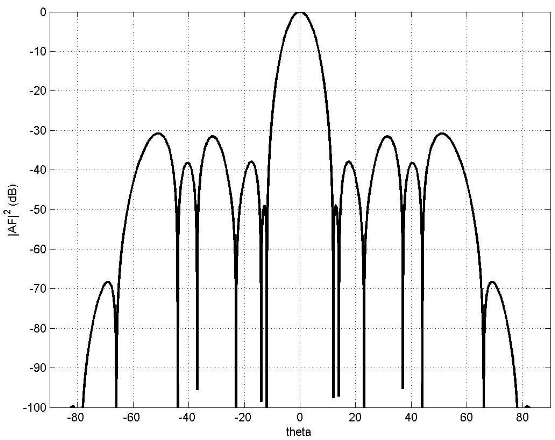

Figure 5. Radiation pattern of 20 elements array for the second scenario

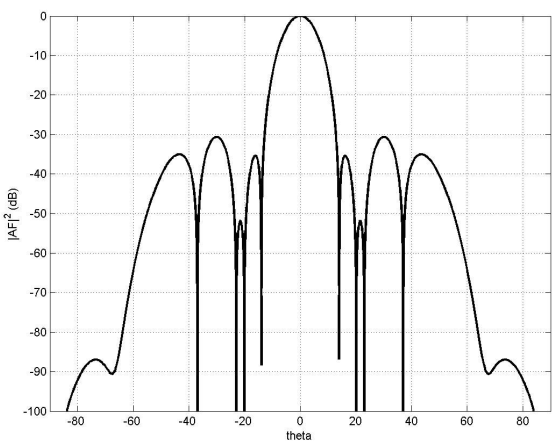

Figure 6. Radiation pattern of thinned array for the second scenario

Discarded elements are selected by the algorithm and this process is done on a random basis. From the random nature of the evolutionary algorithm it is possible that the method can still return with a full-filled array.

In the amplitude tables both in Table 1 and Table 2, the zero magnitudes mean that there is no element in that position.

Element number	Typical solution	Example-1	Example-2
±1	1	1	1
±2	0.8991	0.8537	0.8713
±3	0.8797	0.6010	0.6167
±4	0.7640	0.3479	0.3498
±5	0.6205	0.1338	0.1244
±6	0.5228	0	0
±7	0.3510	0	0
±8	0.3140	0	0
±9	0.1510	0.0182	0
±10	0.0460	0	0

Table-1 Amplitudes of the first scenario

Element number	Typical solution	Example
±1	0.9816	1
±2	1	0.9430
±3	0.8429	0.8468
±4	0.7325	0.6193
±5	0.5679	0.4176
±6	0.4665	0.3043
±7	0.3681	0.1486
±8	0.1660	0.0272
±9	0.1150	0
±10	0.0686	0

Table 2. Amplitudes for the second scenario

V. CONCLUSION

In this work, a method for the problem of array thinning is suggested. A linear array placed symmetrically on *z*-axis is considered for examples and it is shown that the method can be used for array thinning problem and it can be used in design procedure. The method can also be applied to more complex geometries such as a planar geometry. A program is written in C++ language to calculate the amplitudes. MATLAB program is used to show the results. By using this method the unnecessary elements can be discarded from the array under desired conditions.

REFERENCES

1. B. Allen, M. Ghavami, Adaptive Array Systems Fundamentals and Applications, John Wiley & Sons, pp.31-41, 2005.
2. A. Tennant., M. M. Dawoud, A. P. Anderson, "Array pattern nulling by element position perturbations using a genetic algorithm," Electronics Letters, Vol. 30, No. 3, pp. 174–176, 1994.
3. W. P. Liao, F. L. Chu, "Array pattern nulling by phase and position perturbations with the use of the genetic algorithm," Microwave Opt. Technology Lett., Vol. 15, No. 4, pp. 251–256, 1997.
4. D. Karaboğa, K. Güney, A. Akdağlı, "Antenna array pattern nulling by controlling both amplitude and phase using modified touring ant colony optimization algorithm," Int. J. Electronics, Vol. 91, No. 4, pp. 241–251, 2004.
5. S. Yang, Y. B. Gan, A. Qing, "Antenna-array pattern nulling using a differential evolution algorithm," Int. J RF and Microwave CAE, Vol. 14, No. 1, pp. 57–63, 2004.
6. D. G. Kurup, M. Himdi, A. Rydberg, "Synthesis of uniform amplitude unequally spaced antenna arrays using the differential evolution algorithm," IEEE Trans. on Antennas and Propagat., Vol. 51, No. 9, pp. 2210–2217, 2003.
7. K. Güney, M. Onay, "Amplitude-only pattern nulling of linear antenna arrays with use of bees algorithm," Progress in Electromagnetic Research, vol. 70, pp. 21-36, 2007.
8. R. Storn, K. Price, "Differential evolution-A simple and efficient heuristic for global optimization over continuous spaces," Journal of Global Optimization, Vol. 11, No. 4, pp. 341–359, 1997.
9. K. V. Price, R. M. Storn, J. A. Lampinen, Differential evolution-A practical Approach to global optimization, Springer, 2005.