Optimization of Thinned Arrays using Stochastic Immunity Genetic Algorithm

Zoubir M. Hamici [1], and Taisir H. Ismail [2]

[1] Amman University, Computer Engineering Department, Al Seru area, al salt road, 19328, Jordan.

[2] Amman University, Communication Engineering Department, Al Seru area, al salt road, 19328, Jordan.

***Abstract*—In this paper we propose a novel genetic algorithm called Immunity Genetic Algorithm (IGA) based on stochastic crossover evolution to solve the synthesis problem of thinned arrays. Our crossover operator is a variant of the known GA operator. A new expression of the array factor for a specific number of elements N is expressed as a linear Discrete Cosine Transform (DCT). Using IGA to generate thousands of array bit patterns and the DCT to compute the fitness function will result in a very high speed computation compared to traditional computation techniques. This high performance allows us to find a good approximation of the absolute minimum SLL of synthesized thinned arrays. Simulation results of this novel array signal processing technique show the effectiveness for pattern synthesis with low SLL.**

Keywords*—Thinned arrays; pattern synthesis; Immunity genetic algorithm; array factor transform.*

I. Introduction

ARRAY signal processing is the corner stone in the synthesis of antennas for long distance communication and radar systems. The design and development of such antennas with high directivity, low side lobe level (SLL) has been investigated during the last three decades. Since large antenna size is needed to obtain narrow main beam width, it requires a cumbersome procedure to calculate the current excitations which will be very sensitive to the relative errors. Thinned arrays are an interesting approach to minimize the SLL while maintaining the BW unchanged [1-3]. In contrast to periodic antennas where all elements are equally spaced, one approach of producing thinned array is with unequal inter-element spacing of the array [3, 4]. However, to simplify the feeding network, the constituent antennas in a linear array having identical characteristics, are uniformly spaced with identical current excitations.

Furthermore, the hardware complexity can be avoided in a thinned array by removing some elements in a uniformly spaced array to produce low side lobes [5]. The resulting thinned array has fewer number of elements and the inter-element spacing is integer multiples of one-half wavelength. The problem of finding the best thinned array with minimum SLL involves checking a large number of possibilities. Therefore, the thinning optimization problem is non-deterministic which means that most gradient optimization methods are not appropriate.

Recent researches for designing thinned arrays are oriented towards new algorithms to produce the lowest relative SLL, i.e. genetic algorithms (GA) [5-8], simulated annealing (SA) [9,10], neural networks [11], evolutionary programming [12-15]. Also, recently, a pattern search (PS) algorithm was proposed as a tool for array thinning [14]. Moreover, differential evolution algorithm (DE) and a binary-coded GA are jointly applied to the minimization of SLL problem in planar arrays [15]. The most prominent advantage of DE is its low computation time compared to that of GA, particularly in large antenna arrays.

In this paper a novel genetic algorithm called Immunity Genetic Algorithm (IGA) based on stochastic crossover evolution is introduced to solve thinned array optimization problem by removing some elements of a uniformly spaced array in order to produce low side lobes. Our crossover operator is a variant of the known GA crossover operator. It is based on the swapping of two-points selected on the parent unduplicated chromosome also known as Deoxyribonucleic acid (DNA). Furthermore a new expression of the array factor is presented based on Discrete Cosine Transform with precomputed DCT matrix. The linear transform is used to achieve a high speed computation of the array factor which allows us to push the search process to new borders in stochastic search for optimum solutions. Determining the side lobe level accurately is based on finding the nulls of the first lobe; the thinning process of an array causes the nulls to move, to resolve this problem a technique for dynamically finding the exact position of the antenna main beam is presented. With 40% thinning our algorithm achieves a side lobe level less than -20dB and with 20% thinning we achieve about -23dB SLL for 200 elements arrays.

II. Thinning Using Immunity Genetic Algorithm

Consider a Centro-symmetric linear array of $2N$ equispaced isotropic elements placed along the x-axis. A thinned antenna array and the corresponding binary sequence are shown in Figure 1. Practically, the feeding network of a large array can be simplified by exciting the antenna elements uniformly. Furthermore, thinned array is implemented by turning off (removing) some elements to produce low side lobes [5]. Elements at the center of the array play an important role on the SLL, and hence they are kept turned on (active) as shown in Figure 1 (indicated as array core). The Centro-symmetric thinned array factor is given by

$$AF(\theta) = \sum_{m=-N}^{-1} a_m \, e^{jmdk\,\sin(\theta)} + \sum_{n=1}^{N} a_n \, e^{jndk\sin(\theta)} \quad (1)$$

The first term is the array factor of left-hand half of the array and the second term is the array factor of right-hand half of the array.

Let $m=-n$ and $a_{-n} = a_n$ then

$$AF(\theta) = \sum_{n=1}^{N} a_n e^{-jndk\sin(\theta)} + \sum_{n=1}^{N} a_n \, e^{jndk\sin(\theta)} = \mathbf{F}(-\theta) + \mathbf{F}(\theta)$$

where, a_n represent the n*th* current amplitude of array elements with (a_n = 0 if the element is removed (thinned) and a_n = 1 if the element is on or active), d represents the interelement spacing , and θ represents angle from broadside. The Centro-symmetric array of size 2N is then reduced to

$$AF(\theta) = 2\sum_{n=1}^{N} a_n \cos[n\,k\,d\,\sin(\theta)] \quad (2)$$

For one part of the array, the arrival angle is θ while for the other part the arrival angle is -θ. The excitation amplitudes (a_n) are taken in a symmetrical manner relative to the center of the array. Note that there is no element at the center.

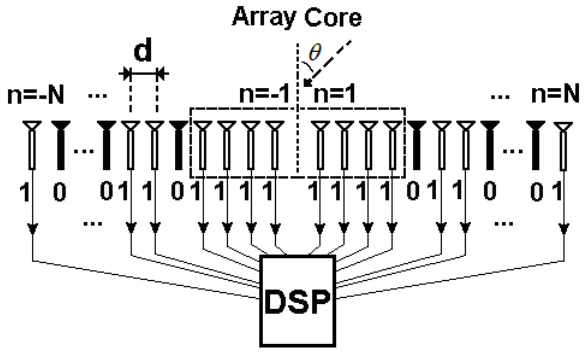

Fig. 1. Centro-symmetric thinned antenna array and the corresponding binary sequence (array bit pattern) representing active and thinned elements.

Figure 1 shows that the elements with white color are turned on (active elements) and are represented by the bit 1, and black elements are removed or thinned elements and are represented by the bit 0. The binary sequence of the array is considered to be the chromosome of the genetic algorithm.

In genetic algorithms (GA), crossover is a genetic operator used to vary the programming of a chromosome or chromosomes from one generation to the next. It is analogous to reproduction and biological crossover, upon which genetic algorithms are based. Many crossover techniques exist for organisms which use different data structures to store themselves. Our crossover algorithm is a variant of these algorithms, based on the swapping of two-points which are selected on the parent unduplicated chromosome; these two genes (bits) are swapped to produce a child unduplicated chromosome (evolution). It is similar to a DNA immunization process through generations against chronic viruses, hence the name Immunity Genetic Algorithm (IGA). The swapping is based on the generation of two random variables.

In this work, array elements binary sequence called array bit pattern is similar to a chromosome, where the immunity to diseases evolves from parent to child by crossover evolution. In order to find how this chromosome will evolve from generation to generation by changing a part of its genes (bits), we tried to generate large number of the chromosomes and then search for the best one in term of acquired immunity (array SLL). The core of the chromosome (Array Core) which contains the main genetic properties is not affected by the crossover through generations, and it is estimated to be 40% of the number of genes (bits), the rest of the chromosome genes are changed by evolution except the last element. This evolution from parent to child is performed by swapping individual bits in the chromosome (array bit pattern). The selection of bits to be swapped (crossed over) is based on a stochastic process controlled by two variables. The immunity of the chromosome to viruses, which represents the SLL of the array pattern, is measured and the best chromosome with high immunity is kept (array factor with lowest SLL). Consequently, our crossover genetic evolution algorithm called IGA is given by the following steps:

Algorithm

1- Initialization; the array size (*2N*), the number of generated chromosomes (*K*) of length *N*, the evolutions index (*I*), the thinning range [L_{min}*+1*, *N-1*] (L_{min} represents the index of the last element of the array core), the thinning efficiency (Th_{eff}), and initial DNA.
2- Generate two random indexes N_1, N_2 within thinning area.
3- Crossover the two bits (genes) of the last array bit pattern indexed by N_1 and N_2:
   $a_n(N_1) \leftarrow a_n(N_2)$ and $a_n(N_2) \leftarrow a_n(N_1)$.
4- Add this new array bit pattern to the set of generated chromosomes.
5- Increment index *I*.
6- If $I < K$ GOTO step 2.
7- Save array chromosomes population.

The *K* evolutions, generated stochastically, are spread all over the total number of possible evolutions given by

$$C_J^{N-Lmin-1} = \frac{(N - L_{min} - 1)!}{(N - J - L_{min} - 1)!.J!} \quad (3)$$

where J is the number of active elements in the thinning area given by

$$J = N(1 - Th_{eff}) - L_{min} - 1 \quad (4)$$

When the number of active elements, J, is variable, then the total number of all possible bit patterns will be $2^{(N-Lmin-1)}$. For

N=50 (actual array size is 2N) and L_{min}=19 this means that for a thinning efficiency (Th_{eff}) of 40% there will be 30,045,015 possible evolutions. The K evolutions generated by the stochastic crossover although are very limited compared to the whole evolutions space; they are spread on all possible evolution values and not concentrated in a limited area. This is a type of spread spectrum evolution where the search for lowest side lobe level is performed with fixed thinning efficiency.

Each evolution by stochastic crossover is stored for later analysis. The core of the chromosome is not affected by the evolutions. In genetic organism, this means that the global characteristics inheritations are conserved from parent to child through generations. The evolution from parent to child is illustrated by figure 2. The evolutions of genes in the biological chromosome are similar to the evolutions of bits in the thinned array bit pattern. The bit is 1 (white) when the array element is active and is 0 (black) when the element is removed.

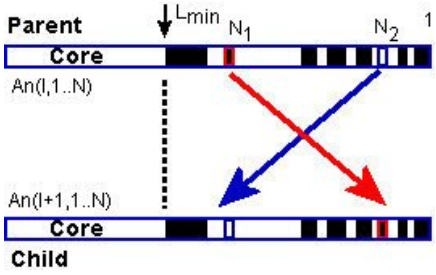

Fig. 2. DNA Immunity Evolution from parent to child.

After generating a set of thousands chromosomes (candidate solutions), a sample of evolutions is shown in figure 3, where the array core representing the unchanged genes (bits) is in the middle.

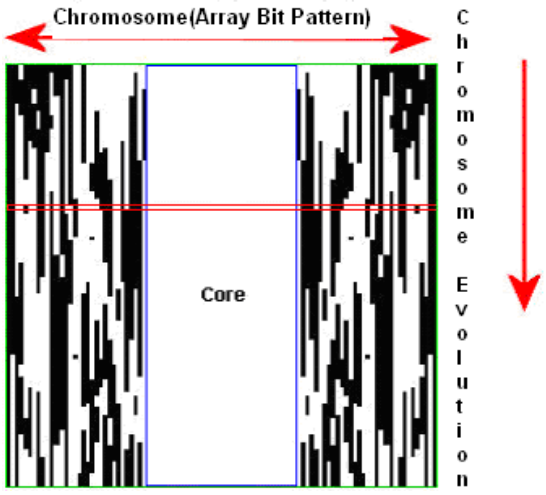

Fig. 3. Sample of the symmetric (duplicated) chromosome evolution generated by stochastic IGA.

Each row represents a possible solution with white pixels denoting active elements and black pixels denote thinned elements. The sample of evolutions is represented by an image which is the top view of the thinned array bit patterns matrix. The process of generating array bit patterns and measuring their SLL in order to increase their rejection to interferences is similar to a DNA immunization process against chronic viruses. The chromosome evolution from parent to child is performed by a 2 bits crossover until a desired side lobe level is obtained. All identical chromosomes are removed (Hamming distance d=0). Experiments have shown that about 50% of the chromosomes are duplicated. As the population of possible solutions is large, a fast computation technique is needed to evaluate the SLL of all array bit patterns.

III. Direct Array Factor Transform

The array factor expression given, by equation (2) is similar to the Discrete Cosine Transform (DCT), therefore, it can be written as linear transformation using matrix representation. With *M* is the number of evaluation points of the array factor and *N* is half the array elements. The array factor expression for the Centro-symmetric array of equation (2) is rewritten as

$$\begin{bmatrix} AF(\theta_1) \\ AF(\theta_2) \\ \vdots \\ AF(\theta_M) \end{bmatrix} = 2 \begin{bmatrix} W(1,\theta_1) & W(2,\theta_1) & \cdots & W(N,\theta_1) \\ W(1,\theta_2) & W(2,\theta_2) & \cdots & W(N,\theta_2) \\ \vdots & \vdots & \ddots & \vdots \\ W(1,\theta_M) & W(2,\theta_M) & \cdots & W(N,\theta_M) \end{bmatrix} \cdot \begin{bmatrix} a_1 \\ a_2 \\ \vdots \\ a_N \end{bmatrix} \quad (5)$$

where $W(n,\theta) = \cos\left[k\, d\, n \sin(\theta)\right]$. The above expression is called Direct Array Factor Transform (DAFT) and can be rewritten as

$$\mathbf{AF}_M = \mathbf{W}_{(n,\theta)} . \mathbf{a}_N \quad (6)$$

where, $\mathbf{a}_N$ is the amplitude weight vector of the *N* elements and $\mathbf{W}_{(n,\theta)}$ is the array factor transform (AFT) matrix of size (*M*,*N*) expressed by a non square matrix. Since the AFT matrix does not depend on the array coefficients, it can be precomputed to speed up the Transform computation process. This will push the computation process to new limits in the stochastic process where we search for a best solution among an astronomic number of combination to satisfy the design requirements such as thinning efficiency and SLL. The DAFT is applied to a set of thinned array bit pattern generated by the IGA algorithm. After a set of thousands of array bit patterns is generated, a search is performed within the corresponding chromosomes array factors to satisfy a given SLL.

IV. Side Lobe Level Estimation

When the Array Factor Transform is computed for all array bit patterns generated by the IGA, a calculation of the side

lobe level is performed on each array factor. This step is very critical because it is the key feature of an array bit pattern selection. The Array Factor contains the main beam or first lobe and contains other lobes. In order to exactly measure the SLL an accurate estimation of the first lobe is needed. The first lobe nulls cannot be estimated correctly from analytical solution since these nulls moves depending on the thinning process. Hence a good estimation is based on a good numerical method. To achieve this goal we considered the edges of the first lobe as an image processing edge detection problem and used the zero-crossing method to estimate their positions. The first derivative of the Array Factor normalized magnitude is computed. Figure 4, shows the array factor (Fig. 4a) and its derivative (Fig. 4b). The exact position of the first lobe is determined by the location of the zero crossing of its derivative.

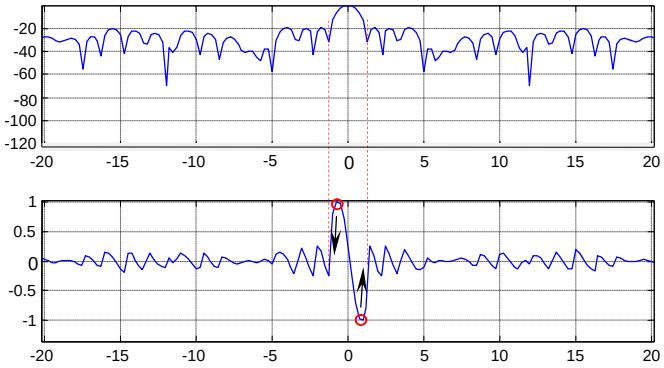

Fig. 4. Main beam or first lobe edges detection using zero-crossing method. Dashed lines show the main beam is located exactly at the zero crossing of the array factor derivative.

V. Results And Discussion

The new thinned linear antenna array design method that utilizes a stochastic Immunity GA is demonstrated by many computer simulation examples in order to minimize the SLL. The simulation is performed in two steps: 1) generate thousands of chromosomes (array bit patterns) for a fixed thinning percentage. 2) Compute the SLL for all chromosomes population and select the best chromosome which gives the lowest SLL. The computing was achieved on a dual core2 computer running at 1.8GHz equipped with 2GB of RAM. When generating a population of 300 thousands stochastic chromosomes only 50% of them are unduplicated (DNA), this is due to the fact that the probability to generate two numbers A and B is quasi the same as generating B and A in a large trials. The swapping of genes (bits) in the two cases will produce the same unduplicated chromosome. This also agrees with the fact that genetic chromosomes are also duplicated. After generating a given chromosomes population a process of filtering duplicated chromosomes (equal bit patterns) is done and only the filtered population is stored for later analysis using DAFT.

To verify the validity of the proposed thinning technique, simulations were conducted on 100 and 200 equispaced Centro-symmetric linear array elements of a half wave interelement spacing for different thinning percentages. In the following examples, three different cases with different thinning percentages are presented. Figure 5 shows the SLL values of an unduplicated population of more than 131 thousands array bit patterns obtained by the IGA algorithm for 100 elements (N=50) and thinning efficiency 40%. From the Figure, the lowest SLL is -18.4dB and the corresponding array bit pattern is given in row one of Table 1. Figure 6 shows the array factor of the array bit pattern given in row 1 of Table 1 with lowest SLL. Figure 7 shows the SLL of a population of about 100 thousands array bit patterns for a 30% thinning. The lowest SLL is -19 dB as shown in Figure 8. The corresponding best array bit pattern is given in row two of Table 1. Figure 8 shows the array factor with lowest SLL of the corresponding array bit pattern (array bit pattern given in row 2 Table 1). An SLL of 20.24 dB of the best array bit pattern (row three of Table 1) is achieved for a population of about 90 thousands array bit pattern for 20% thinning as shown in Figure 9.

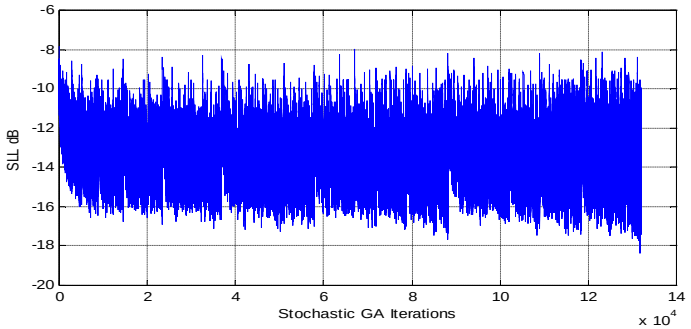

Fig. 5. SLL versus stochastic IGA iterations for an array of 100 elements (N=50), thinning efficiency 40% , and about 131 thousands population size.

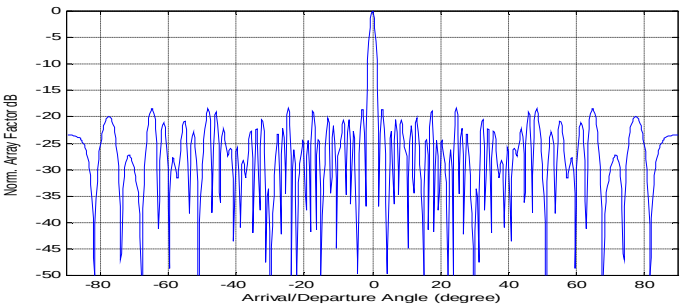

Fig. 6. Array factor for the best array bit pattern given in row one of Table I, 2N=100, only 60 elements are active. The SLL is -18.4 dB.

Figure 10 shows the array factor with the lowest SLL of the best array bit pattern shown in Figure 9 (array bit pattern given in row 3 Table 1). In all previous examples, the computation is achieved in less than 46 seconds since using DAFT, one array factor is computed in 0.35 ms, for *N* equal to 50 and a resolution of 720 evaluation points (*M*) equivalent

to an evaluation point each 0.25 degree of the spactial spectra.

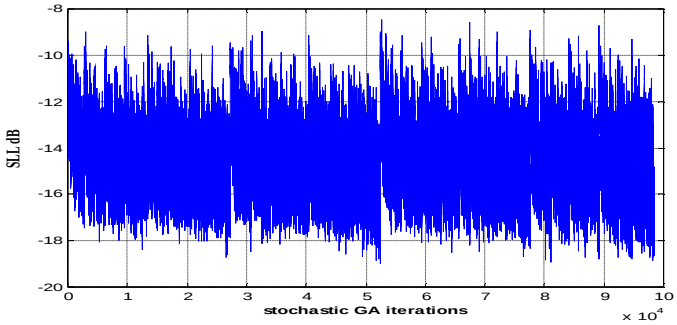

Fig. 7. SLL versus stochastic IGA iterations for an array of 100 elements (N=50), thinning efficiency 30% , and about 100 thousands population size.

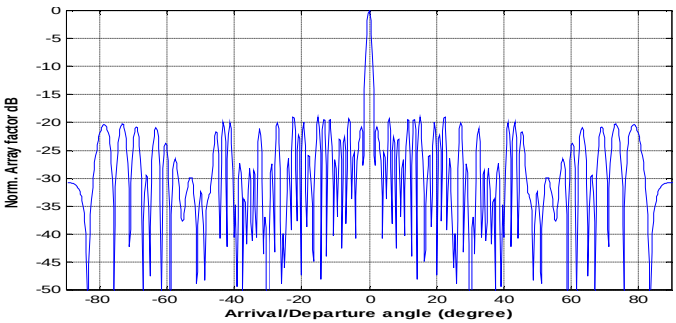

Fig. 8. Array factor for the array bit pattern given in row two of Table I, 2N=100, only 70 elements are active. The SLL is -19 dB.

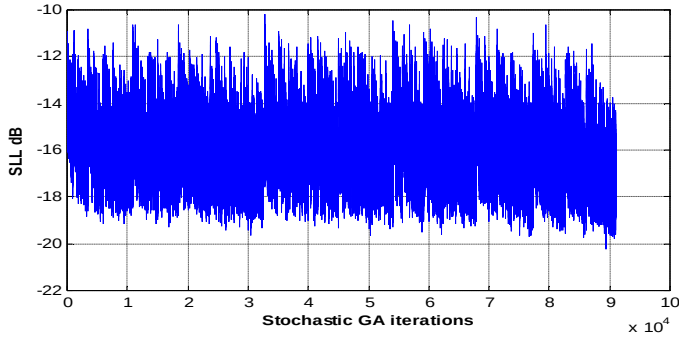

Fig. 9. SLL versus stochastic IGA iterations for a an array of 100 elements (N=50), thinning efficiency 20% , and about 91 thousands population size.

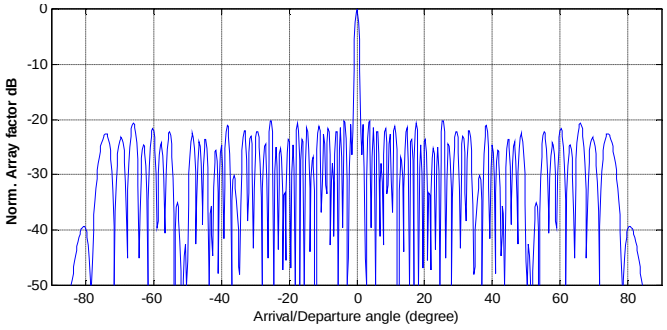

Fig. 10. Array factor for the array bit pattern given in row three of Table I, 2N=100, only 80 elements are active. The SLL is -20.24 dB.

Table 1. Array bit patterns and the corresponding SLL for differenT thinning percentageS, Th_{eff}, with 100 elements.

Fig	Array Bit Pattern with the lowest SLL	Th_{eff} %	Pop. $\times 10^3$	SLL (dB)
6	**1111111111111111111111101001011100010000000000000001**	40	131	-18.4
8	**1111111111111111111111011111110011100000000010100001**	30	100	-19.0
10	**1111111111111111111111111111101100111010000011101011**	20	90	-20.24

Since the optimization problem is highly non-linear and the search for the best array bit pattern is stochastic, the array bit pattern with the lowest SLL is not unique, as we may obtain different array bit patterns with similar SLL in different trials and some times in the same population. Table 2 gives different array bit patterns for different trials with almost the same SLL with 20% thinning percentage. The generated chromosomes of the thinned array given in Table 2 part (a) was 200 thousands chromosome population, which produced about 91 thousands unduplicated chromosomes (different array bit patterns), in 13 second and we computes all the corresponding DAFTs and searches all SLL in 35 seconds. Through thousands of generated array bit patterns, we have observed that the core of the array represents about 40% of the total number of elements which is not affected by the thinning process, since thinning the core will affect severly the SLL. Consequently, the design of the thinned array is accomplished by thinning the last 60% elements of the antenna array except the last element which controls the total length of the array.

Table 2. Different Array bit patterns that yield almost the same SLL.
a) 2N= 100 elements, b) 2N= 200 elements.

Trial	Array Bit Pattern (20% thinning, 100 elements Array)	SLL(dB)
1	**11111111111111111111111111011001110100001101011**	-20.24
2	**11111111111111111111111111010110001110100011011**	-19.99
3	**11111111111111111111110111011110110100100001111**	-20.03

a.

Trial	Array Bit Pattern (20% thinning, 200 elements Array)	SLL(dB)
1	**11111111111111111111111111111111111111111111101111 11101111100110110100101001100010001011011110000011**	-22.2
2	**11111111111111111111111111111111111111110111111101 01001111110111100011010011100101000111011001101**	-22.4
3	**11111111111111111111111111111111111111101111011111 01111011011100101110110010101011111100000101100111**	-22.1

b.

To demonstrate the capability of this new technique, we consider the thinning of a large array of 200 elemtens with thinning percentage of 20%. Figure 11 show an array factor for 200 elemtens, an SLL of -22.4dB is obtained. The corresponding bit pattern is given in row two of Table 2 (b),

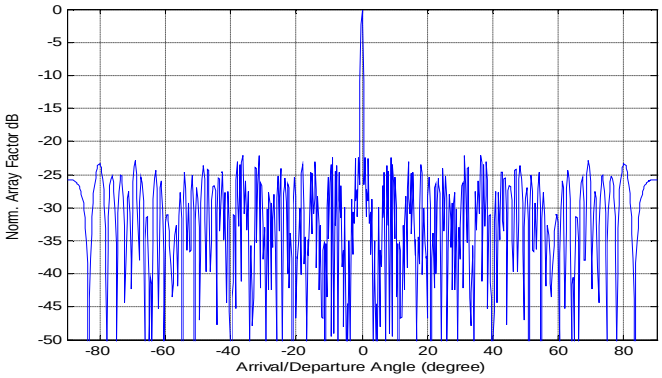

Fig. 11. Array factor for the array bit pattern given in row two of Table II (b), 2N=200, only 160 elements are active. The SLL is -22.4 dB.

In order to increase the efficiency, and due to symmetry of the array factor, the computation can be performed in the range from 0° to 90° instead of -90° to 90°; this will save 50% of the computation time. Using this feature we are able to compute 145 thousands array factors and extract the best array bit pattern for 200 elements in less than one minute.

VI. Conclusion

This paper presents a novel algorithm for designing a thinned linear antenna array with fixed percentage of thinning using a stochastic Immunity Genetic algorithm (IGA). Our IGA is a variant of the known crossover operator. It is based on the swapping of two-genes which are selected on the parent unduplicated chromosome (DNA). Several examples are presented in order to show the effectiveness of our proposed technique. Our main objective is to find the best SLL with fixed thinning percentage. Since the nulls of the array factor moves depending on thinning percentage we proposed a very fast method to determine the main beam width and hence compute the SLL with high accuracy. The method is based on the zero-crossing of the array factor derivative. We express the array factor as a Discrete Cosine Transform with precomputed DCT matrix. With this formulation, we were capable of finding the best array bit pattern for 200 elements by performing a search within a stochastic population of chromosomes (array bit patterns) of about 145 thousands extracted from 300 thousands of duplicated chromosomes of the IGA, all computations of the extracted population are done in less than one minute. Since the array factor is expressed as a linear transform based on a pre-computed DCT matrix, a huge reduction in computation time is obtained. This allows us to find a good approximation of the absolute minimum SLL of synthesized thinned arrays. Results for a thinned linear isotropic antenna array of different sizes have illustrated the performance of our proposed technique. Simulation results show the effectiveness of this novel algorithm for pattern synthesis with low SLL.

References

[1]. R. J. Mailloux and E. Cohen, "Statistically thinned arrays with quantized element weights", *IEEE Trans. Antennas Propagt.*, vol. 39, pp. 436-447, April 1991.

[2]. P. J. Bevelacqua and C. A. Balanis, "Minimum sidelobe levels for linear arrays", *IEEE Trans. Antennas Propagt.*, vol. 55, pp.3442-3449, Dec. 2007.

[3]. B. P. Kumar and G. R. Branner, "Design of unequally spaced arrays for performance improvement", *IEEE Trans. Antennas Propagt.*, vol. 47, pp.511-523, March 1999.

[4]. Lee,K. C. and J. Y. Jhang, "Application of particle swarm algorithm to the optimization of unequally spaced antenna arrays," Journal of Electromagnetic Waves and Applications, vol. 20,no. 14, pp.2001–2012, 2006.

[5]. R. L. Haupt, "Thinned arrays using genetic algorithms", *IEEE Trans. Antennas Propagt.*, vol. 42, pp. 993-999, July 1994.

[6]. M. Bray, D. Werner, D. Boeringer, and D. Machuga, "Optimization of thinned aperiodic linear phased arrays using genetic algorithms to reduce grating lobes during scanning", *IEEE Trans. Antennas Propagt.*, vol. 50, no.12, pp.1732-1742, Dec. 2002.

[7]. Mahanti,G. K.,N. Pathak,and P. Mahanti,"Synthesis of thinned linear antenna arrays with fixed sidelobe level using real-coded genetic algorithm," *Progress In Electromagnetics Research*, PIER 75, pp.319–328,2007.

[8]. Buckley M.J., "Linear array synthesis using a hybrid genetic algorithm", *Proc. IEEE Ant. Propagat. Soc. Int. Symp.*, Baltimore, MD, July 1996, pp. 584-587. 1996.

[9]. C. S. Ruf, "Numerical annealing low-redundancy linear arrays", *IEEE Trans. Antennas Propagt.*, vol. 41, pp. 85-90, Jan. 1993.

[10]. V. Murino, A. Trucco, & C. Regazzoni. "Synthesis of unequally spaced arrays by simulated annealing. *IEEE Trans. on Signal Processing"*, vol. 44,no. 1, pp.119–123. 1996.

[11]. Mulholland J.E., DiMeo F.N., Hoorfar A., Goverdhanam K. "The Optimization of Thinned Phased Arrays by the Use of Neural Networks," *10th Annual Benjamin Franklin Symposium*, Philadelphia, PA, May 2. 1992.

[12]. Kumar Chellapilla, SS Rao, and Ahmad Hoorfar, "Optimization of Thinned Phased Arrays using Evolutionary Programming,", *7th Intl. Conf. on Evolutionary Programming*, M. Valley Marriott, San Diego, CA, Mar 25-27, 1998

[13]. M. Rattan, M. S. Patterh, and B.S. Sohi. "Antenna Array Optimization using Evolutionary Approaches", *Apeiron*, vol. 15, no. 1, pp. 78. January 2008.

[14]. A. Razavi and K. Forooraghi, "Thinned arrays using pattern search algorithms", *Progress in Electromagnetics Research*, PIER 78, pp.61-71, 2008.

[15]. C. Rocha-Alicanoa, D. Covarrubias-Rosalesa, C. Brizuela-Rodrigueza, and M. Panduro-Mendozab, "Differential evolution algorithm applied to sidelobe level reduction on a planar array", *Int. J. Electron. Commun. (AEÜ)* vol. 61, pp.286– 290. 2007.