Progress In Electromagnetics Research, PIER 43, 287–304, 2003

GENETICALLY EVOLVED PHASE-AGGREGATION TECHNIQUE FOR LINEAR ARRAYS CONTROL

D. I. Abu-Al-Nadi and M. J. Mismar

Department of Electrical Engineering
Faculty of Engineering and Technology
University of Jordan
Amman, Jordan

T. H. Ismail

Department of Electrical Engineering
Al-Ahliyya Amman University
Amman, Jordan

Abstract—This paper presents a modified Genetic Algorithm (GA) technique in which the large phase perturbations are calculated by aggregating small phase increments. The proposed aggregation GA technique overcomes the major drawback of the large solution space required by the classical GA techniques. The proposed method adopts small ranges for increments of the parameters and the optimality is reached via aggregation of the best increments of phases. Consequently, the GA searches in a smaller solution space and finds the solution with reduced number of iterations. Simulation results show the achieved improvement of the proposed technique over the classical GA. The suppressed sectors using phase-only control are accomplished with and without element failures. Problems like imposing symmetrical nulls around the mainbeam and compensation for the failure of center element have been achieved.

1. INTRODUCTION

Array pattern synthesis has long been of interest to antenna designers where suppressed sectors are created at the direction of the interfering signals. In general, the pattern synthesis techniques require variable complex weights, variable phase shifters, or variable attenuators at the elements of the array. Array pattern control with variable complex weights is rather expensive compared with the phase-only control [1, 2] and amplitude-only [3] control. But, the phase-only control has the advantage of design simplicity as the required phase shifters are normally incorporated at the array elements. Generally, pattern synthesis by the phase-only control in the conventional configuration of a linear antenna involves nonlinear algorithms to obtain the phase adjustments [1, 4]. The nonlinear problem can be simplified by restricting the array phase perturbations to small values such that the problem can be linearized [2]. To avoid the limitation of the small phase restriction, large values for the phases are allowed using nonlinear algorithms. A linear algorithm for large phase values can be achieved using an iterative procedure with linear programming [5]. A general strategy for phase-only control have been described and is applicable to continuous apertures and to linear and planer arrays [6].

Genetic Algorithms (GA's) are optimization techniques that have been used to solve general problems with objective functions that do not possess continuity and differentiability properties. Recently, GA's have been used as an optimization tool to control many of the features of the array patterns. A GA was developed by Haupt [7] to adjust a quantized phase-only weights by small phase perturbations. Unlike binary coding and binary crossover genetic algorithms, an approach has been proposed where the excitation weighting vectors were presented as complex number chromosomes and uses decimal linear crossover without crossover site [8]. Also, the GA was used to establish the optimal solution for a uniformly null-filled array and for array thinning [9]. On the other hand, GA was employed to steer the pattern nulls when only the amplitudes of the array elements are perturbed [10].

In fact, the pattern synthesis with phase-only control is considered to be a highly nonlinear problem; therefore, GA's have been found to be well suited to this problem including the element failure case. In this work, the large phase perturbations of the antenna elements are calculated by aggregating small phase increments using the proposed aggregation genetic algorithm. The modified GA is used to find the best phase increments of the elements such that the performance index is optimum. This technique will overcome the major drawback of the classical approach; that is, the solution space of each parameter should be specified. For a specified large range of solution space, the convergence will be slow. On the other hand, for a specified small range of solution space the optimal solution is not guaranteed. Therefore, the proposed aggregation method adopts small ranges for the increments of the parameters and the optimality can be reached via the accumulation of the best increments of phases. This will cause the genetic algorithm to search in small solution space and find the solution with reduced number of iterations.

2. PROBLEM FORMULATION

The initial pattern of N equi-spaced isotropic elements has the following form

$$F_o(\theta) = \sum_{n=1}^{N} a_{0n} e^{j(d_n \kappa \sin\theta + \phi_n)} \tag{1}$$

where a_{0n} is the current excitation of the nth element, d_n is the position of the nth element, κ is the wave number, θ is the scanning angle from broadside, and ϕ_n is the initial phase shift of the nth element. Denoting the angular direction $u = \sin(\theta)$ and using matrix notation, Equation (1) can be formed as

$$F_o(u) = \mathbf{\Phi}^0 \boldsymbol{S}(u) \tag{2}$$

where

$$\mathbf{\Phi}^0 = \begin{bmatrix} e^{j\psi_1^0} & e^{j\psi_2^0} & e^{j\psi_3^0} & \cdots & e^{j\psi_N^0} \end{bmatrix} \tag{3}$$

$$\boldsymbol{S}(u) = \begin{bmatrix} a_{01}e^{jd_1ku} & a_{02}e^{jd_2ku} & a_{03}e^{jd_3ku} & \cdots & a_{0N}e^{jd_Nku} \end{bmatrix}^T \tag{4}$$

and $\psi_n^0 = \phi_n$.

The method of null steering by controlling the phase shifters is obtained by determining a set of new phase values such that the resultant pattern has nulls formed in the interference directions while maintaining the main beam pointing towards the desired signal. Consequently, the specifications are set by a desired function, template, that is defined as the initial antenna array pattern in the main beam region, the specified side lobe level, α_s, and the specified null levels, α_j, $j = 1, 2, \ldots, J$, where J is the number of interfering signals. Hence, the template, $D(u)$, is defined as

$$D(u) = \begin{cases} F_o(u) & u \in R_o \\ \alpha_s & u \in R_s \\ \alpha_j & u \in R_j,\ j = 1, 2, \ldots, J \end{cases} \tag{5}$$

where R_o, R_s, and R_j represent the angular regions of the main beam, the side lobes and the null regions, respectively. Realization of the above pattern can be done using phase-only control method with the following function

$$F(u) = \mathbf{\Phi} \boldsymbol{S}(u) \tag{6}$$

where

$$\mathbf{\Phi} = \begin{bmatrix} e^{j\psi_1} & e^{j\psi_2} & e^{j\psi_3} & \cdots & e^{j\psi_N} \end{bmatrix} \tag{7}$$

The phases ψ_n for $n = 1, \ldots, N$ should be chosen such that $F(u)$ approximates the template $D(u)$.

3. SOLUTION USING AGGREGATION GENETIC ALGORITHM

In the classical GA's, the solution space of each parameter should be specified in the genetic search with real values. This range of the solution space is usually unknown which will cause two problems; first, it is hard to guess the suitable range of each parameter; second, a large range should be adapted. The consequence will be a larger number of iterations is needed to find the optimal solution in this large solution space. On the other hand, the proposed method adopts small ranges for the increments of the parameters and the optimality criterion is to find the best increments of phases that should be aggregated to maximize the fitness function. This will cause the genetic algorithm to search in small solution space and find the solution in less number of iterations. The exact range of each parameter is not vital in this approach since the accumulation of the parameter increments will take care of this matter.

Let the array pattern at the kth aggregation step to be expressed as

$$F^k(u) = \mathbf{\Phi}^k \boldsymbol{S}(u) \tag{8}$$

where $\mathbf{\Phi}^k$ denotes the complex phase vector at the kth aggregation step, and is expressed as

$$\mathbf{\Phi}^k = \begin{bmatrix} e^{j\psi_1^k} & e^{j\psi_2^k} & e^{j\psi_3^k} & \cdots & e^{j\psi_N^k} \end{bmatrix} \tag{9}$$

It is desired to proceed from the initial phases to the final phases such that $F^k(u)$ approximates $D(u)$ in a predefined sense as the number of aggregation steps increases. To establish this procedure, let the phase vector at the kth aggregation step be expressed as

$$\boldsymbol{\Psi}^k = \boldsymbol{\Psi}^{k-1} + \boldsymbol{B}^k \tag{10}$$

where

$$\boldsymbol{\Psi}^k = \begin{bmatrix} \psi_1^k & \psi_2^k & \psi_3^k & \cdots & \psi_N^k \end{bmatrix} \tag{11}$$

and

$$\boldsymbol{B}^k = \begin{bmatrix} \beta_1^k & \beta_2^k & \beta_3^k & \cdots & \beta_N^k \end{bmatrix} \tag{12}$$

is the vector of phase increments at the kth aggregation step.

Previous works have been concentrated on finding the phase vector $\boldsymbol{\Psi}$ via GA's, in this work GA's are used to find the best vector of increments $\boldsymbol{B}$ such that $\boldsymbol{\Psi}$ will be optimum. This modification gives better rates of convergence and overcomes the problem of restricting the solution space which is inherent in GA's. Genetic algorithms maintain and manipulate a family or population of solutions and implement what is called survival of the fittest strategy during the search for better solution. GA's evolve a population of individuals based on the mechanics of natural selection, genetics and evolution. Each individual of the population represents a trial solution of the problem and is called a chromosome which is usually represented by binary strings. In [11] chromosomes are extended to real values, and it is shown that real-valued GA is more efficient than binary GA. The use of GA requires the determination of the following six fundamental issues [12]: chromosome representation, selection function, genetic operators making up the reproduction function, the creation of initial population, termination criteria, and the evaluation function.

The first issue is chromosome representation which is needed to describe each individual in the population of interest. The representation scheme determines how the problem is structured in the GA and the operators that are used. Floating-point numbers have been used to make up the sequence of genes for each chromosome.

The second issue is selection of individuals to produce successive generations which plays an important role in GA. The selection function determines which of the individuals will survive and continue on to the next generation. Various methods for selection are available such as Roulette Wheel [13], Normalized Geometric Ranking [12], and Tournament Selection. Normalized geometric ranking was adapted for this application.

The third issue is genetic operators, which provide the search mechanism of the GA. The operators are used to create new solutions based on existing ones. There are two basic types of operators, crossover and mutation. Crossover takes two individuals and produces two new individuals while mutation alters one individual to produce a single new solution. Operators for real-valued representation have been developed in [11]. In general, let $\boldsymbol{B_L}$ and $\boldsymbol{B_U}$ be the lower and upper bounds for the vector $\boldsymbol{B}$, respectively, i.e.,

$$\boldsymbol{B_L} = [\beta_{1L} \quad \beta_{2L} \quad \beta_{3L} \quad \cdots \quad \beta_{NL}] \tag{13}$$

$$\boldsymbol{B_U} = [\beta_{1U} \quad \beta_{2U} \quad \beta_{3U} \quad \cdots \quad \beta_{NU}] \tag{14}$$

Heuristic crossover produces a linear extrapolation of the two individuals. A new individual is created as follows:

$$\begin{aligned} \boldsymbol{B}_{1c} &= \boldsymbol{B}_{1p} + diag(\boldsymbol{r})(\boldsymbol{B}_{1p} - \boldsymbol{B}_{2p}) \\ \boldsymbol{B}_{2c} &= \boldsymbol{B}_{1p} \end{aligned} \tag{15}$$

where $\boldsymbol{B}_{1p}$, $\boldsymbol{B}_{2p}$ are two parent individuals and $\boldsymbol{B}_{1p}$ is better than $\boldsymbol{B}_{2p}$, $\boldsymbol{B}_{1c}$, $\boldsymbol{B}_{2c}$: are two children individuals, and $\boldsymbol{r}$ is a vector of uniform random number between (0,1).

For this type of crossover a feasibility check, f, on $\boldsymbol{B}_{1c}$ should be done as follows:

$$f = \begin{cases} 1 & \text{if } \beta_{nL} \le \beta_{n1c} \le \beta_{nU}, \quad n \in N \\ 0 & \text{otherwise} \end{cases} \tag{16}$$

where β_{n1c} is the nth increment for the first child chromosome. If f equals 1, the crossover takes place; otherwise, a new random number r is generated and the heuristic crossover is done again. If the number of failures exceeds a preset value, let the children equal the parents and stop.

The multi-non-uniform mutation operator applies the following operator to all the variables in the individual:

$$\beta_n = \begin{cases} \beta_n + (\beta_{nU} - \beta_{nL}) f(G) & \text{if} \quad r_1 < 0.5 \\ \beta_n - (\beta_{nU} + \beta_{nL}) f(G) & \text{if} \quad r_1 \ge 0.5 \\ \beta_n & \text{otherwise} \end{cases} \tag{17}$$

where $f(G) = (r_2(1 - \frac{G}{G_{\max}}))^b$, r_1, r_2 are uniform random numbers between (0,1), G is the current generation, $G_{\max}$ is the maximum number of generations, and b is a shape parameter.

The fourth issue is that the GA must be provided with an initial population, P_o. It is usually done by randomly generating solutions for the entire population within the search space. Improvements on the convergence can be done by seeding potentially good solutions which are taken form other optimization methods.

The fifth issue is termination of the search for the best individual should be stopped if one of the following three conditions is satisfied: reaching the maximum number of iterations, reaching the optimum solution, or the solution exceeds some tolerance based on the performance index.

The last issue is the evaluation function or better named the fitness function in GA and it is a vital part of this type of optimization method. The survival of the individuals is based on this evaluation function, the fitted individuals, i.e., individuals having the highest values of fitness will stay for the next generation and the others will be discarded. GA optimization is usually designed for maximization. If the optimization problem is to minimize a function $f(x)$, this is equivalent to minimizing a function g, where $g(x) = -f(x)$, i.e.,

$$\min\, f(x) = \max\, g(x) = \max\, \{-f(x)\} \tag{18}$$

In this work, the evaluation function is the minimization of the maximum deviation in decibel between the actual output of the array system $F^k(u)$ (hereafter will be denoted $F^k(u, \boldsymbol{B}^k)$) and the template specified by $D(u)$. By applying (17) for the minimization case, the performance index, $J^k(u, \boldsymbol{B}^k)$, will be

$$J^k(u, \boldsymbol{B}^k) = \min_{\boldsymbol{B}^k} \max_{u} \Big(F^k(u, \boldsymbol{B}^k)_{dB} - D(u)_{dB}\Big) \tag{19}$$

The optimization process in this case is to find the best individual $\boldsymbol{B}^k$ which will be used in (10) to minimize $J^k(u, \boldsymbol{B}^k)$; eventually it minimizes the maximum deviation between $F^k(u, \boldsymbol{B}^k)$ and $D(u)$.

The proposed algorithm will proceed as follows:

1. Initialization:
   - Set $max\_k = K,\ max\_i = I.$
   - Set $k = 0.$
   - Set $i = 0.$
   - Set $\boldsymbol{B_L}$, and $\boldsymbol{B_U}$
   - Set $\boldsymbol{\Psi}^0 = [0 \quad 0 \quad \cdots \quad 0].$
   - Set $\boldsymbol{B}^0_{best} = [0 \quad 0 \quad \cdots \quad 0].$
2. Randomly generate an initial population P_o of M individuals.
3. Include the individual $\boldsymbol{B}^k_{best}$ with the initial population.
4. Evaluate the respective performance index for each individual of the population.
   - $\boldsymbol{\Psi}^k = \boldsymbol{\Psi}^{k-1} + \boldsymbol{B}^k$
   - $\boldsymbol{\Phi}^k = \Big[e^{j\psi_1^k} \quad e^{j\psi_2^k} \quad e^{j\psi_3^k} \quad \cdots \quad e^{j\psi_N^k}\Big]$

- $F^k(u, \boldsymbol{B}^k) = \boldsymbol{\Phi}^k \boldsymbol{S}(u)$
- $J^k(u, \boldsymbol{B}^k) = \min_{\boldsymbol{B}^k} \max_{u} \Big(F^k(u, \boldsymbol{B}^k)_{dB} - D(u)_{dB} \Big)$

5. Generate an intermediate population P_r by selection operator.
6. Generate a new population P_{new} by crossover and mutation operators on P_r.
7. If $i < I$, set $i = i + 1$, and go to step 4; otherwise, proceed to the next step.
8. Use the best solution $\boldsymbol{B}^k_{best}$ to find the phase vector $\boldsymbol{\Psi}^k = \boldsymbol{\Psi}^{k-1} + \boldsymbol{B}^k_{best}$
9. Set $k = k + 1$,
   if $k < K$,
   - set $i = 0$
   - go to step 2

   else
   - $\boldsymbol{\Phi}^k = \Big[e^{j\psi_1^k} \quad e^{j\psi_2^k} \quad e^{j\psi_3^k} \quad \cdots \quad e^{j\psi_N^k} \Big]$
   - $F^k(u, \boldsymbol{B}^k_{best}) = \boldsymbol{\Phi}^k \boldsymbol{S}(u)$

   end.

From the above algorithm, the GA is used to find the best phase increments of the elements such that the performance index is optimum. Then, as proposed in the algorithm, the large phase perturbations of the antenna elements are calculated by aggregating those small phase increments.

4. PHASE CONTROL WITH ELEMENT FAILURES

Many practical factors such as aging and accidents may lead to failure of antenna elements. A random distribution of the element failures will degrade the initial pattern by filling the suppressed sectors and by modifying the SLL value. The amount of degradation of the array pattern depends on the number and the locations of the failed elements. Also, random element failures will yield a nonsymmetrical distribution of the array element coefficients [14]. Therefore, the corrected pattern after F failed elements has the following form

$$F(u) \sum_{n \in N_c} e^{j\phi_n} a_{0n} e^{(jd_n \kappa \sin\theta)} \tag{20}$$

where N_c is the set of non-failed elements in the linear array and it has N-F elements. The correction is done such that $F(u)$ approximates the template specified by $D(\theta)$ which is defined in Equation (5). Following the same derivation of the previous section, the performance index can be written as

$$J^k(u, \boldsymbol{B}_f^k) = \min_{\boldsymbol{B}^k} \max_u \Big(F^k(u, \boldsymbol{B}_f^k)_{dB} - D(u)_{dB} \Big) \tag{21}$$

where the vector $\boldsymbol{B}_f^k$ contains the phase increments of the non-failed elements at the kth aggregation step.

5. NUMERICAL EXAMPLES AND DISCUSSION

The new GA based method of phase-only control to suppress narrow and wide band interference is demonstrated using a 20 equispaced linear array elements of a half wave interelement spacing. The interference suppression is accomplished such that $F(u)$ approximates the template specified by $D(u)$ which is formed by maintaining the mainbeam directed towards the desired source, specifying the sidelobe level in the sidelobe region, and creating suppressed sectors in the direction of the interferences. In the following computer simulations, the desired main beam and the SLL of the template are assumed as a classical Dolph-Chebyshev linear array design with SLL of 30 dB.

The results of creating multiple suppressed narrowband interferences in the sidelobe region are presented. Let the template to be used contains three narrowband interferences at the angular directions $u_1 = [-0.5, -0.49]$, $u_2 = [0.4, 0.405]$, and $u_3 = [0.61, 0.62]$ as shown in Figure 1 (dashed). Using the proposed aggregation genetic approach with a maximum number of iteration, $K \times I = 200$ ($K = 10$ and $I = 20$) and the bounds on the phase increments $\beta_{nU} = -\beta_{nL} = \frac{\pi}{20}$ for all n, the phases of antenna elements are computed as given in column 2 of Table 1. The corresponding approximation pattern with the three prescribed suppressed sectors is shown in Figure 1 (Solid). From the Figure, the suppressed narrow sectors' depths $\{\alpha_j,\ j = 1, 2, 3\}$ are 79 dB, 80 dB, and 78 dB, respectively.

Also, this proposed method can impose symmetrical nulls around the mainbeam due to the capability of this aggregation method to obtain large phases even though small phase increments are imposed. Figure 2 shows the pattern which approximates the template with two symmetrical suppressed sectors imposed at $\Delta u_1 = (-0.405, -0.4)$ and $\Delta u_2 = (0.4, 0.405)$ with a maximum number of iteration, $K \times I = 200$ ($K = 10$ and $I = 20$) and the bounds on the phase increments $\beta_{nU} = -\beta_{nL} = \frac{\pi}{20}$ for all n. From the Figure, symmetrical suppressed sectors are achieved around the mainbeam with 75 dB depths for both sectors. Column 3 of Table 1 gives the computed element phase perturbations for the Figure. The above results show the ability of

Table 1. Computed element phases $\{\psi_n\}$ for Figures 1, 2, 3, and 4.

Element No.	ψ_n (Radians)			
	Fig.1	Fig. 2	Fig. 3	Fig. 4
1	-0.0353	-0.0761	0.0621	0.0946
2	-0.1366	0.0015	-0.0463	-0.0533
3	-0.0284	0.0173	-0.0894	-0.1036
4	0.0521	-0.1133	-0.0295	-0.0142
5	-0.0776	0.4289	0.1047	0.0414
6	0.1471	0.3536	-0.0510	0.1168
7	0.0797	-0.1171	-0.0082	-0.1533
8	0.0270	-0.0443	-0.0631	-0.1394
9	0.0377	-0.0968	-0.0358	-0.0199
10	0.0094	-0.2326	0.0529	-0.0519
11	-0.1454	-0.1257	-0.0348	-0.1240
12	-0.0314	-0.3205	-0.0817	-0.0688
13	0.0145	-0.0073	0.0567	0.1669
14	0.0573	-0.1389	0.0709	-0.0344
15	0.0283	-0.3439	-0.1014	-0.0313
16	0.0221	-0.2060	-0.0706	0.0268
17	-0.0848	0.1020	-0.0367	-0.1067
18	-0.0794	0.0829	-0.0051	0.1458
19	-0.0152	0.0971	0.0011	0.0952
20	0.1351	0.0500	-0.1046	-0.1303

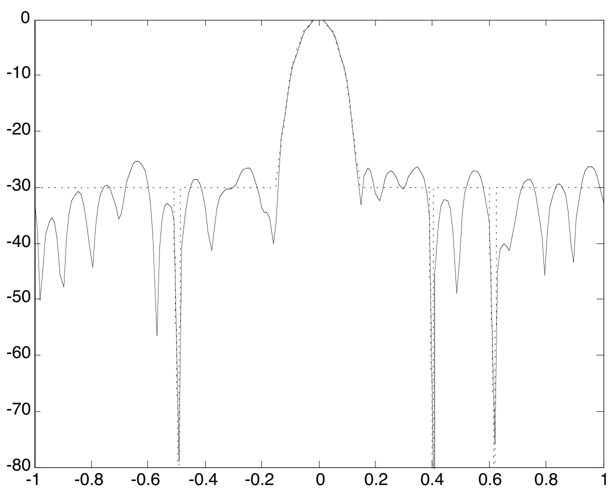

Figure 1. Pattern synthesis using the proposed technique with three suppressed sectors at $[-0.5, -0.49]$, $[0.4, 0.405]$, and $[0.61, 0.62]$ (solid), and the fitness template $D(\theta)$ (dashed).

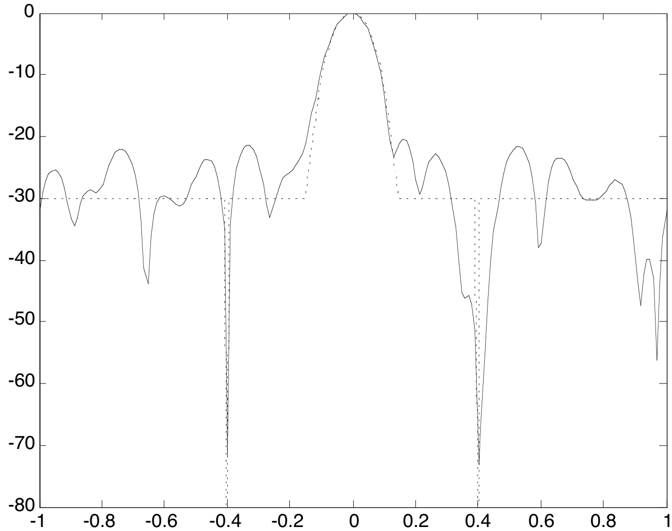

Figure 2. Pattern synthesis with two symmetrical null sectors at $[-0.405, -0.40]$ and $[0.4, 0.405]$ (solid), and the fitness template $D(\theta)$ (dashed).

this technique to suppress multiple narrow band interfering signals even when they are symmetrically located around the mainbeam.

To discuss the effect of the maximum number of aggregation K on the suppressed sector level with fixed number of iteration, $K \times I = 40$, the same template of the Chebyshev linear array with a SLL of 30 dB is assumed to contain one wide suppressed sector at the angular direction $\Delta u_1 = (0.4, 0.44)$ as shown in Figure 3 (dashed). Figure 3 shows the approximation pattern without aggregation, $K = 1$ and $I = 40$, while Figure 4 shows the approximation pattern with aggregation, $K = 4$, and $I = 10$ and the bounds on the phase increments $\beta_{nU} = -\beta_{nL} = \frac{\pi}{20}$ for all n. From Figure 3 the achieved suppressed sector level is 45 dB with SLL of 22 dB. However, Figure 4 shows that the suppressed sector level is 55 dB with SLL of 22 dB. The corresponding phase perturbations for Figure 3 and Figure 4 are given in column 4 and column 5 of Table 1. From the previous results, it is clear that better performance is achieved with the aggregation method compared to the other method denoted without aggregation. Also, Figure 5 shows the performance index, $J^k(u, \boldsymbol{B}^k)$, versus the number of iteration, $K \times I$, for the patterns shown in Figures 3 and 4. From Figure 5, the speed of convergence of the aggregation method is faster and the achieved minimum performance index value is less compared to the method

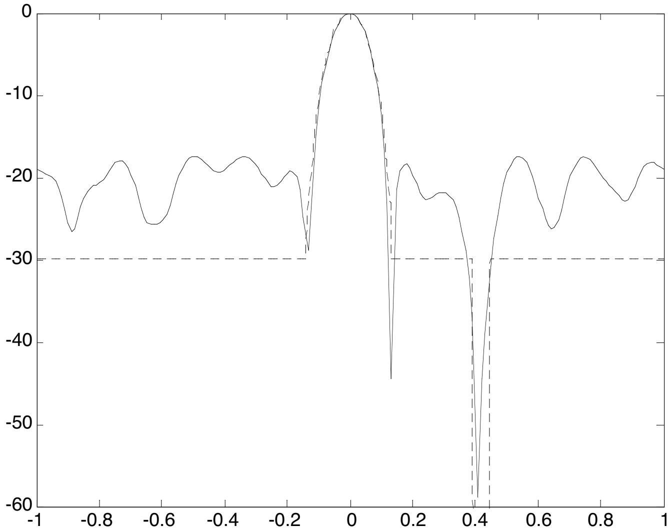

Figure 3. Pattern synthesis with a wide band null sector at $[0.4, 0.44]$ without aggregation, $K = 1$, and $I = 40$ (solid) and the fitness template $D(\theta)$ (dashed).

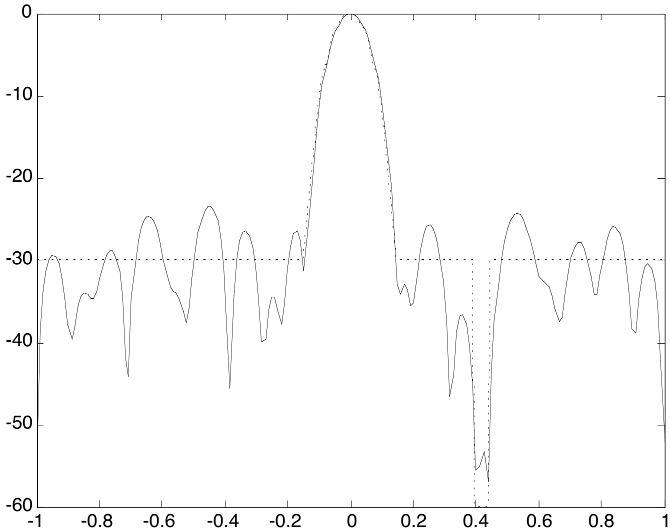

Figure 4. Pattern synthesis with a wide band null sector at $[0.4, 0.44]$ with aggregation, $K = 4$, and $I = 10$ (solid), and the fitness template $D(\theta)$ (dashed).

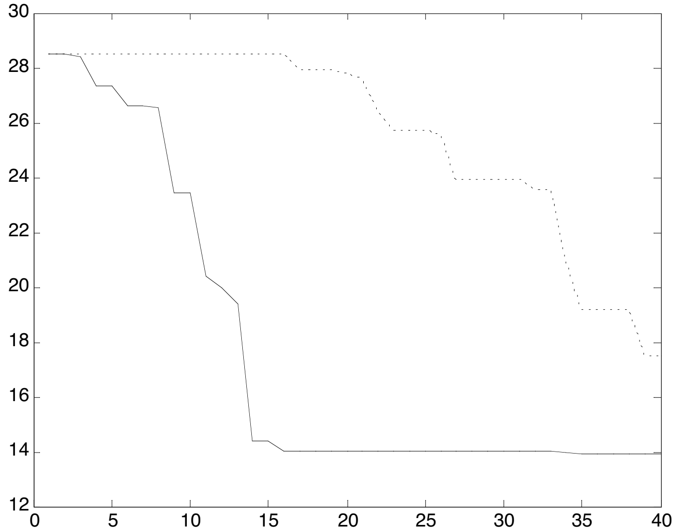

Figure 5. The performance index versus $(K \times I)$ for the results of Fig. 3 (dashed: $K = 1$, $I = 40$), and Fig. 4. (solid: $K = 4$, $I = 10$).

without aggregation.

The effects of limiting the maximum phase increments on the performance index, $J^k(u, \boldsymbol{B}^k)$, and the speed of convergence with the aggregation method are simulated. Figure 6 shows these effects with different bounds on the phase increments, ($\beta_{nU} = -\beta_{nL} = \frac{\pi}{2}, \frac{\pi}{20}, \frac{\pi}{40}$ for all n). From the Figure, the performance index, $J^k(u, \boldsymbol{B}^k)$, is minimum and the speed of convergence is faster when the ranges of the phase increments are bounded by $\beta_{nU} = -\beta_{nL} = \frac{\pi}{20}$. However, the speed of convergence is better in the early stages of the iterations when $\beta_{nU} = -\beta_{nL} = \frac{\pi}{2}$ due to the large solution space. The ranges of the phase increments should not be made very small, this will need more iterations to converge. This proposed aggregation method gives better rates of convergence and overcomes the problem of restricting the solution space which is inherent in GA.

To validate the correction of the array pattern with suppressed sectors with random element failures, assume that one suppressed sector was designed using the same initial pattern without element failures as shown in Figure 7 (dashed). From the Figure, the mainbeam width and the sidelobe level of the initial pattern are 5.3° and 22 dB, respectively, while the suppressed sector depth is 55 dB. The corresponding elements' coefficients of the designed pattern with the

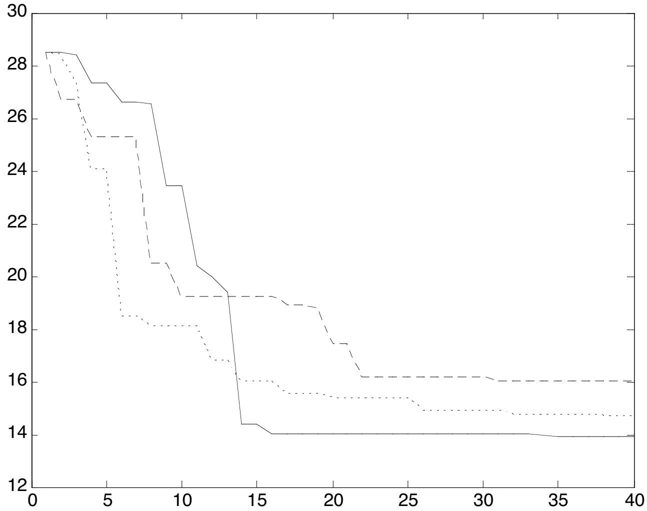

Figure 6. The performance index versus number of iterations ($K = 4$ and $I = 10$) for three different bounds on the phase increments ($\beta_{nU} = -\beta_{nL} = \frac{\pi}{2}, \frac{\pi}{20}, \frac{\pi}{40}$ for all n) shown as dashed, solid, and dash-dotted curves, respectively.

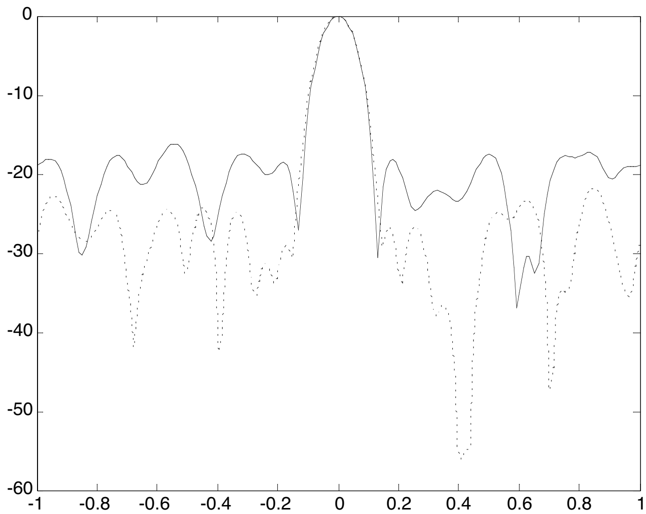

Figure 7. The initial array pattern with one suppressed sector (dashed), and the degraded array pattern due to the defective elements of indices 2 and 10 (solid).

Table 2. Phases in radians for the initial pattern and the corrected pattern with failure of elements 2 and 10.

Element No	Phases of Initial Pattern	Phases of Corrected Pattern
1	-0.0068	0.0665
2	-0.2173	0
3	-0.2231	-0.0185
4	0.0395	-0.490
5	0.0122	0.0211
6	-0.0832	0.0162
7	0.0887	0.1567
8	-0.1503	-0.0952
9	0.1159	-0.1875
10	-0.0471	0
11	0.2650	0.2907
12	-0.0850	0.2137
13	-0.0293	-0.0229
14	0.1857	-0.0353
15	-0.1906	-0.1801
16	-0.1987	-0.1318
17	-0.0026	-0.1796
18	0.2352	-0.0716
19	-0.1084	0.1437
20	-0.1183	0.0796

prescribed suppressed sector are given in Column 2 of Table 2. Now, assuming that the 2nd and the 10th elements are defective, then the corresponding array factor can be calculated using equation (1) with as shown in the Figure 7 (solid). From the Figure, the suppressed sector is filled and the sidelobe level is degraded due to the effect of the defective elements. Using the proposed aggregation GA, the array pattern is corrected using only the phases of the non-defective element set with maximum aggregation number, $K = 4$, and $I = 10$ where the bounds on the phase increments $\beta_{nU} = -\beta_{nL} = \frac{\pi}{20}$ for all n as given in column 3 of Table 2. Figure 8 shows the corrected pattern using the calculated phases (solid) compared with the template (dashed). The corrected pattern shows that the recovered suppressed sector depth is 55 dB and the sidelobe level is 21 dB while the mainbeam is almost unchanged.

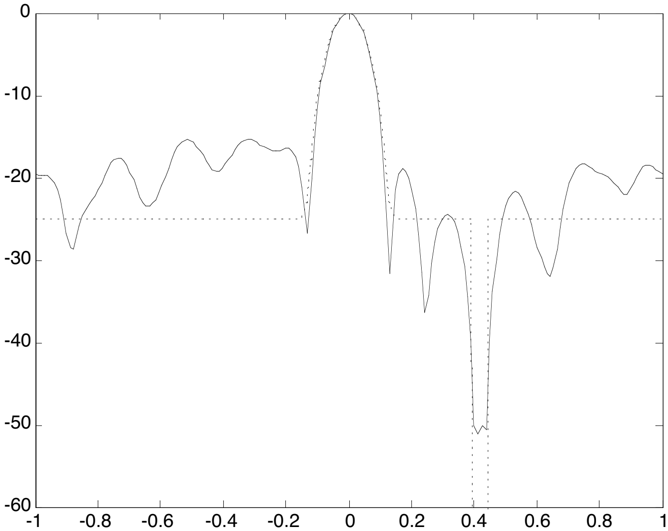

Figure 8. The corrected array pattern with defective elements of indices 2 and 10 (solid), and the fitness template $D(\theta)$ (dashed).

6. CONCLUSIONS

A new aggregation method using a genetic algorithm to suppress a multiple narrow and wide band interferences using phase-only control has been presented. Genetic algorithms can solve high nonlinear problems like the one at hand where large phases are required to obtain global minimum of the performance index. The large phase perturbations of the antenna elements are calculated by aggregating small phase increments using the proposed aggregation genetic algorithm. The modified GA is used to find the best phase increments of the elements such that the performance index is optimum. This modification gives better rates of convergence and overcomes the problem of the large solution space which is inherent in GA.

The computer simulation results show that the phase-only control using the aggregation method, is more efficient compared with the classical genetic method. Unlike small phase perturbations techniques, the aggregation genetic algorithm can impose symmetrical nulls around the main beam. The suppressed sectors using the full phase-only control are accomplished with and without element failures.

REFERENCES

1. Steyskal, H., R. A. Shore, and R. L. Haupt, "Methods for null control and their effects on radiation pattern," *IEEE Trans. on Antenna and Propagation*, Vol. 34, 404–409, 1986.
2. Steyskal, H., "Simple method for pattern nulling by phase perturbation," *IEEE Trans. on Antenna and Propagation*, Vol. 31, 163–166, 1983.
3. Vu, T. B., "Method of null steering without using phase shifters," *IEE Proceedings*, Vol. 131, Pt. H, No. 4, 1984.
4. Shore, R., "Nulling at symmetric pattern location with phase-only weight control," *IEEE Trans. on Antenna and Propagation*, Vol. 32, No. 5, 530–533, 1984.
5. Mismar, M. J. and T. H. Ismail, " Pattern nulling by iterative phase perturbation," *Progress in Electromagnetics Research*, PIER 22, 181–195, 1999.
6. Trastoy, A. and F. Ares, "Phase-only control of antenna sum patterns," *Progress in Electromagnetics Research*, PIER 30, 47–57, 2001.
7. Haupt, L. R., "Phase-only adaptive nulling with a genetic algorithm," *IEEE Trans. on Antenna and Propagation*, Vol. 45, No. 6, 1009–1015, 1997.
8. Yan, K. and Y. Lu, "Sidelobe reduction in array-pattern synthesis using genetic algorithm," *IEEE Trans. on Antenna and Propagation*, Vol. 45, No. 7, 1117–11122, 1997.
9. Ares-Pena, F., J. Rodriguez-Gonzalez, E. Villanueva-Lopez, and S. Rengarajan, "Genetic algorithms in the design and optimization of antenna array patterns," *IEEE Trans. on Antenna and Propagation*, Vol. 47, No. 3, 506–510, 1999.
10. Dawoud, M. and M. Nuruzzaman, "Null steering in rectangular planer arrays by amplitude control using genetic algorithms," *Int. J. Electronics*, Vol. 87, No. 12, 1473–1484, 2000.
11. Michalewicz, Z., *Genetic Algorithms+Data Structures=Evolution Programs*, 3rd edition, Springer-Verlag, Berlin Heidelberg, 1996.
12. Joines, J. A. and C. R. Houck, "On the use of non-stationary penalty functions to solve nonlinear constrained optimization problems with GA's," *Proceedings of the 1st IEEE Int. Conf. On Evolutionary Computation*, Vol. 1, 579–584, Orlando, June 27–29, 1994.
13. Holland, J. H., *Adaptation in the Natural and Artificial Systems*, University of Michigan Press, Ann Arbor, 1975.

14. Mismar, M. J. and T. H. Ismail, "Null steering with element failures using partial controlled linear arrays," *Electromagnetics*, Vol. 23, No. 5, 445–454, 2003.