A Hybrid of Genetic Algorithm and Particle Swarm Optimization for Antenna Design

W. T. Li, L. Xu, and X. W. Shi

National Key Laboratory of Antenna and Microwave Technology, Xidian University
Xi’an 710071, China

Abstract—In this paper, a new effective optimization algorithm called PGHA is presented, which combines in the most effective way the properties of two of the most popular evolutionary optimization approaches now in use for the optimization of electromagnetic structures, the Particle Swarm Optimization (PSO) and Genetic Algorithms (GA). Some improved genetic mechanisms based on non-linear ranking selection, competition and selection among several crossover offspring and adaptive change of mutation scaling are adopted in the paper to overcome the drawbacks of standard genetic algorithm. Furthermore, the proposed algorithm is successfully applied to design a linear array with ten elements and a circular array with thirty one elements and obtain the desired beam forms. We try to use a modified Bernstern polynomial to reduce the number of variables when calculating the circular array, and simulation results show the abroad foreground of PGHA in the antenna array design.

1. INTRODUCTION

The global synthesis of antenna arrays that generate a desired radiation pattern is a highly nonlinear optimization problem. Pattern synthesis is known as the process of choosing the parameters of an antenna array to produce desired radiation characteristics. Many analytical methods have been proposed for its solution. Examples of analytical techniques include the well-known Taylor method and the Chebishev method [1]. However, analytical or calculus-based methods are generally unable to optimize problem with precedence constraint. To this end, stochastic methods are necessary [2, 3] in order to efficiently deal with large nonlinear search spaces and to extend the analysis.

Advantages of evolutionary computation are the capability to find a global optimum, without being trapped in local optima, and the possibility to face nonlinear and discontinuous problems, with great numbers of variables. Genetic Algorithms (GA) [4] have proven to be a useful method of optimization for difficult and discontinuous multidimensional engineering problems. A new method of optimization, Particle Swarm optimization (PSO) [5], is able to accomplish the same goal as GA optimization in a new and faster way. Since PSO and GA both work with a population of solutions, combining the searching abilities of both methods seems to be a good approach. The purpose of this paper is to investigate the foundations and performance of the two algorithms when applied to the design of two antenna array designs.

2. A HYBRID OF GENETIC ALGORITHM AND PARTICLE SWARM OPTIMIZATION

Genetic Algorithm [6] is an iterative stochastic optimizer that works on the concept of survival of the fittest motivated by Darwin, using methods based on the mechanics of natural genetics and natural selection to construct search and optimization procedures that best satisfies a predefined goal. Floating-point GA uses floating-point number representation for the real variables and thus is free from binary encoding and decoding. It takes less memory space and works faster than binary GA. To overcome the drawbacks of standard genetic algorithm such as prematurity and easily trapping in local optimum, some improved genetic mechanisms are adopted in the paper, such as non-linear ranking selection [7], different from the conventional algorithm in which two parents only produce two offspring, the new heuristic crossover operators defined below:

First, it produces three chromosomes from two parents $I^A$ and $I^B$ according to the following mechanisms:

\[ I_1 = r I^A + (1 - r) I^B \]  
\[ I_2 = (1 - r) I^A + r I^B \]  
\[ I_3 = \frac{I^A + I^B}{2} \]
where \( r \) is uniform random number in \([0, 1]\). Then, among \( I_1 \) to \( I_3 \), the two with the largest fitness value are used as the offspring of the crossover operation.

The mutation operator is defined as follows: For a parent \( p \), if variable \( p_k \) was selected at random for this mutation, the result is: 
\[
\bar{p}_k \in \left\{ \max\left( p_k - \mu \frac{p_k^{\text{max}} - p_k^{\text{min}}}{2}, p_k^{\text{min}} \right), \min\left( p_k + \mu \frac{p_k^{\text{max}} - p_k^{\text{min}}}{2}, p_k^{\text{max}} \right) \right\}
\]
(4)

where \( p_k^{\text{max}}, p_k^{\text{min}} \) are upper and lower bounds of \( p_k \) respectively, \( \mu \) decreased with the increase of iterations.

\[
\mu(\tau) = 1 - r^{(1-(\tau/T))^b}
\]
(5)

Particle Swarm Optimization is one of the more recently developed evolutionary technique, and it is based on a suitable model of social interaction between independent agents (particles) and it uses social interaction knowledge (also called swarm intelligence) in order to find the global maximum or minimum of a genetic function [8]. While for the GA the improvement in the population fitness is assured by pseudobiological operators, such as selection, crossover and mutation, the main PSO operator is velocity update:

\[
\vec{v}_i(\tau + 1) = w\vec{v}_i(\tau) + c_1\phi_1(\vec{p}_i(\tau) - \vec{x}_i(\tau)) + c_2\phi_2(\vec{p}_g(\tau) - \vec{x}_i(\tau))
\]
(6)

\[
\vec{x}_i(\tau + 1) = \vec{x}_i(\tau) + \vec{v}_i(\tau + 1)
\]
(7)

where
- \( \vec{v}_i(\tau) \) = particle velocity
- \( \vec{x}_i(\tau) \) = particle variables
- \( \phi_1, \phi_2 \) = independent uniform random numbers
- \( c_1, c_2 \) = learning factors
- \( \vec{p}_i \) = local best solution
- \( \vec{p}_g \) = best global solution
- \( w = w_{\text{max}} - \frac{w_{\text{max}} - w_{\text{min}}}{T} \cdot \tau \)

The PSO algorithm updates the velocity vector for each particle then adds that velocity to the particle position or values. Velocity updates are influenced by both the best global solution and the best local solution in the present population. The advantages of PSO are that it is easy to implement and there are few parameters to adjust. Our PGHA consists essentially in a strong co-operation of the two evolutionary algorithms described above, since it maintains the integration of the two techniques for the entire run.

In each iteration the population is divided into two parts and they are evolved with the two techniques respectively. They are then recombined in the updated population, that is again divided into two parts in the next iteration for another run of genetic or particle swarm operators.

3. PROBLEM FORMULATION

In amplitude-phase synthesis of the linear array, the far field array factor of this array can be written as

\[
F(\theta) = \sum_{n=1}^{N} a_n \exp(j(n \frac{2\pi}{\lambda} d \sin \theta + \beta_n))
\]
(8)

where \( n \) the element number, \( \lambda \) the wavelength, \( \beta_n \) the excitation current phases of the elements, \( a_n \) the excitation current amplitudes of the elements, \( j \) the imaginary, \( d \) is the inter-element spacing, and \( \theta \) is the polar angle of far-field measured from broadside \((-90^\circ \text{ to } +90^\circ)\).
We try to use a modified Bernstein polynomial \[9\] to reduce the number of variables when calculating the circular array. The modified Bernstein polynomial is

\[
F(U) = \begin{cases} 
  B_1 + \frac{1 - B_1}{A^{MA} (1 - A)^{M(1-A)}} U^{MA} (1 - U)^{M(1-A)}, & 0 \leq U \leq A \\
  B_2 + \frac{1 - B_2}{A^{MA} (1 - A)^{M(1-A)}} U^{MA} (1 - U)^{M(1-A)}, & A \leq U \leq 1 
\end{cases}
\]  

(9)

where \(B_1, B_2, M, A\) are parameters in the polynomial, \(B_1\) and \(B_2\) specify the left and right endpoints \(F(0)\) and \(F(1)\) respectively, while increasing \(M\) sharpens and narrows the peak of \(F(U)\). For \(\theta = 90^\circ\), the far field array factor of the circular array is

\[
E(90^\circ, \phi) = \sum_{n=1}^{N} F(U)_n \exp \left( j \frac{2\pi}{\lambda} r (\cos (\phi - \beta_n) - \cos (\beta_n)) \right)
\]  

(10)

where \(F(U)_n\) is the \(n\)th excitation amplitude, being the \(n\)th equal sampling point value of \(F(U)\). \(\beta_n\) is the excitation current phases of the elements, and \(r\) is the circle radius of the array.

A key point of optimization is the construction of the target function. In this paper, the fitness function to be maximized for array optimization problem can be expressed as follows:

\[
Fitness = \frac{1}{\alpha \times |\text{MSLL} - \text{SLVL}| + \gamma \times |F_o(\theta) - F_d(\theta)|}
\]  

(11)

where MSLL the highest sidelobe level, SLVL the desired sidelobe level, and \(F_o(\theta)\) and \(F_d(\theta)\) are, respectively, the pattern obtained by using PGHA and the desired pattern. The values of \(\alpha\) and \(\gamma\) should be selected by experience such that the fitness function is capable of guiding potential solutions to obtain satisfactory array pattern performance with desired properties.

4. NUMERICAL RESULTS

With the aim to validate the effectiveness of the developed technique, two examples are considered here. Firstly, we consider a uniform linear array of 10 isotropic elements spaced 0.5\(\lambda\) along \(x\)-axis

![Linear array geometry.](Figure 1: Linear array geometry.)

![Normalized Power pattern (dB).](Figure 2: (a) Radiation pattern for the linear array. (b) Excitation amplitude distribution for the linear array. (c) Excitation phases in degree for the linear array.)
in order to generate a sector beam as illustrated in Fig. 1. In the amplitude-phase synthesis, the
phase is limited to 0 or \( \pi \). Fig. 2(a) shows the normalized absolute power patterns in dB. Fig. 2(b)
shows common amplitude distribution and Fig. 2(c) shows the phase distributions in degree.

Then, we consider a circular array of 31 isotropic radiators spaced 0.5\( \lambda \) apart along a circle of
radius 6 wavelengths as illustrated in Fig. 3. To reduce the number of variables, we try to use a
modified Bernstein polynomial according to Eq. (9). Fig. 4(a) shows the normalized absolute power
patterns in dB. Fig. 4(b) shows the optimized distribution.

Figure 3: Circular array geometry.

Figure 4: (a) Radiation pattern for the circular array, (b) Excitation amplitude distributions for the circular
array.

5. CONCLUSIONS

An optimization method for the synthesis of linear array pattern and circular array pattern functions
has been proposed and assessed. In order to take advantage of the peculiarities of these two methods,
the proposed algorithm integrates the main features of GA and PSO into the optimization process.
Results clearly show a very good agreement between the desired and synthesized specifications for
the two cases. Since the algorithm proposed in this paper is reliable and effective and this feature
makes it suitable for a wider application in electromagnetics.

ACKNOWLEDGMENT

The authors would like to thank the financial support form national natural science fund of P. R.
China (No. 60571057).

REFERENCES