Optimization of Micro Strip Array Antennas Using Hybrid Particle Swarm Optimizer with Breeding and Subpopulation for Maximum Side-Lobe Reduction

Hichem CHAKER, Sidi Mohamed MERIAH, Fethi Tarik BENDIMERAD

Telecom Laboratory, Dept. of Electronic, Faculty of Engineering, Abou-Bekr Belkaid University- Tlemcen, P.O.Box230, Chetouane, 13000 Tlemcen, Algeria

mh_chaker2005@yahoo.fr, sm_meriah@mail.univ-tlemcem.dz , ft_bendimerad@mail.univ-tlemcem.dz

Abstract. *In this paper, a technique based on hybrid particle swarm optimiser with breeding and subpopulation is presented for optimal design of reconfigurable dual-beam linear array antennas and planar arrays. In the amplitude-phase synthesis, the design of a reconfigurable dual-pattern antenna array is based on finding a common amplitude distribution that can generate either a pencil or sector beam power pattern, when the phase distribution of the array is modified appropriately. The goal of this study is to introduce the hybrid model to the electromagnetic community and demonstrate its great potential in electromagnetic optimizations.*

Keywords

Hybrid particle swarm, breeding and subpopulation, global optimization, phased array, reconfigurable array, planar array.

1. Introduction

Optimizations of linear antenna arrays have received great attention in the electromagnetic community for many civilian and military applications. Multiple-beam antenna arrays have important applications in communications and radar. Reconfigurable antenna array that are capable of radiating with multiple patterns using a single power divider network are desirable for many applications. Recently, evolutionary algorithms have been successfully applied to antenna array synthesis problems like null steering in phased arrays by positional perturbations [1], reconfigurable phase differentiated array design [2], [3], and the corrugated horn antenna design [4]. To solve the antenna array pattern synthesis problems, among a number of optimization procedures, the artificial intelligence techniques such as genetic, simulated annealing and tabu search algorithms owing to their simplicity, flexibility and accuracy have received much attention in recent years. Genetic algorithm (GA) is a search technique based on an abstract model of Darwinian evolution. Simulated annealing (SA) technique is essentially a local search, in which a move to an inferior solution is allowed with a probability, according to some Boltzmann-type distribution, that decreases as the process progresses. Tabu search (TS) algorithm has been developed to be an effective and efficient scheme for combinatorial optimization that combines a hill-climbing search strategy based on a set of elementary moves and a heuristics to avoid to stops at sub-optimal points and the occurrence of cycles. Recently, particle swarm optimization algorithm (PSO) is proposed for solving global numerical optimization problem. The search techniques mentioned above are the probabilistic search techniques that are simple and easily be implemented without any gradient calculation. This study uses a new electromagnetic optimization technique, hybrid particle swarm optimizer with breeding and subpopulation [5], to design a linear reconfigurable dual-beam antennas array and planar array antennas. The result shows that the hybrid model can find a high quality solution even for a very high dimensional problem.

2. Problem Formulation

The design of a reconfigurable dual-pattern antenna array is based on finding a common amplitude distribution that can generate either a pencil beam or sector beam, when the phase distribution of the array is changed appropriately. All the excitation phases are kept constant at 0° to generate a pencil beam, and are varied in the range -180°≤θ≤ 180° to form a sector beam [2].

If the array element excitations are conjugate symmetrical about the center of the linear array, the far field factor of this array with even number of uniformly spaced isotropic elements (2N) can be given by (1)[6].

$$F(\theta) = 2\sum_{k=1}^{N} a_k \cos\left(\frac{2\pi}{\lambda} d_k \sin(\theta) + \delta_k\right) \tag{1}$$

where N is the element number; λ is the wavelength; δ_k are phases of the elements (-180°≤θ≤ 180°); a_k the amplitude of the elements; d_k the distance between position of i^{th} element and the array center; θ the scanning angle.

Evolutionary algorithms use the concept of fitness to represent how well an arbitrary solution satisfies the design parameters. Each of the parameters used to calculate the fitness is referred to as a fitness factor. The fitness factors must together quantify the solution.

For the reconfigurable dual-beam optimization, the fitness function must quantify the entire array radiation pattern. One possible method of doing so would be to compare the calculated pattern point-by-point with the desired pattern as follows:

$$Fitness = Max - \int_0^{\frac{\pi}{2}} \left|F_d(\theta) - F(\theta)\right| d\theta \,. \tag{2}$$

The fitness can be seen as the difference area between the desired pattern and obtained pattern. The greater value of the fitness function, the better match between the obtained pattern and the desired one. Equation (2) is used for evaluating the fitness value during the optimization process.

3. Hybrid Particle Swarm Optimizer with Breeding and Subpopulations

Both Eberhart and Angeline conclude that hybrid models of the standard GA and the PSO could lead to further advances. We present such a hybrid model. The model incorporates one major aspect of the standard GA into the PSO, the reproduction. In the following work we will refer to the used reproduction and recombination of genes only as "breeding". Breeding is one of the core elements that make the standard GA a powerful algorithm. Hence our hypothesis was that a PSO hybrid with breeding has the potential to reach better results than the standard PSO. In addition to breeding we introduce a hybrid with both breeding and subpopulations. Subpopulations have previously been introduced to standard GA models mainly to prevent premature convergence to suboptimal points [7]. Our motivation for this extension was that the PSO models, including the hybrid PSO with breeding, also reach suboptimal solutions. Breeding between particles in different subpopulations was also added as an interaction mechanism between subpopulations. The traditional PSO model, described by [8], consists of a number of particles moving around in the search space, each representing a possible solution to a numerical problem. Each particle has a position vector $X_i=(x_{i1},\ldots,x_{id},\ldots,\ x_{iD})$, a velocity vector $V_i=(v_{i1},\ldots,v_{id},\ldots,\ v_{iD})$, the position $P_i=(p_{i1},\ldots,p_{id},\ldots,\ p_{iD})$, and fitness of the best point encountered by the particle, and the index $^{(g)}$ of the best particle in the swarm. In each iteration the velocity of each particle is updated according to their best encountered position and the best position encountered by any particle, in the following way:

$$v_{id} = w \times v_{id} + c_1 \times rand() \times (p_{id} - x_{id}) + c_2 \times rand() \times (p_{gd} - x_{id}) \tag{3}$$

w is the inertia weight described in [9], [10] and p_{gd} is the best position known for all particles. c_1 and c_2 are random values different for each particle and for each dimension. If the velocity is higher than a certain limit, called $V_{\max}$, this limit will be used as the new velocity for this particle in this dimension, thus keeping the particles within the search space. The position of each particle is updated in each iteration. This is done by adding the velocity vector to the position vector;

$$x_{id} = x_{id} + v_{id} \,. \tag{4}$$

The particles have no neighborhood restrictions, meaning that each particle can affect all other particles. This neighborhood is of type star (fully connected network), which has been shown to be a good neighborhood type in [11]. Fig. 1 shows the structure illustration of the hybrid model.

```
Begin
    Initialise
    While (not terminate-condition) do
      Begin
        Evaluate
        Calculate new velocity vectors
         Move
         Breed
      End
End
```

Fig. 1. The structure of the hybrid model.

The breeding is done by first determining which of the particles that should breed. This is done by iterating through all the particles and, with probability pb equal to 0.6 (breeding probability); mark a given particle for breeding. Note that the fitness is not used when selecting particles for breeding. From the pool of marked particles we now select two random particles for breeding. This is done until the pool of marked particles is empty. The parent particles are replaced by their offspring particles, thereby keeping the population size fixed. The position of the offspring is found for each dimension by arithmetic crossover on the position of the parents:

$$child_1(x_i) = p_i \times parent_1(x_i) + (1.0 - p_i) \times parent_2(x_i) \,, \tag{5}$$

$$child_2(x_i) = p_i \times parent_2(x_i) + (1.0 - p_i) \times parent_1(x_i) \tag{6}$$

where p_i is a uniformly distributed random value between 0 and 1. The velocity vectors of the offspring are calculated as the sum of the velocity vectors of the parents normalized to the original length of each parent velocity vector.

$$child_1(\vec{v}) = \frac{parent_1(\vec{v}) + parent_2(\vec{v})}{\left|parent_1(\vec{v}) + parent_2(\vec{v})\right|} \times \left|parent_1(\vec{v})\right| , \tag{7}$$

$$child_2(\vec{v}) = \frac{parent_1(\vec{v}) + parent_2(\vec{v})}{\left|parent_1(\vec{v}) + parent_2(\vec{v})\right|} \times \left|parent_2(\vec{v})\right| \cdot \tag{8}$$

The arithmetic crossover of positions in the search space is one of the most commonly used crossover methods with standard real valued GAs, placing the offspring within the hypercube spanned by the parent particles. The main motivation behind the crossover is that offspring particles benefit from both parents. In theory this allows good examination of the search space between particles. Having two particles on different suboptimal peaks breed could result in an escape from a local optimum, and thus aid in achieving a better one. We used the same idea for the crossover of the velocity vector. Adding the velocity vectors of the parents results in the velocity vector of the offspring. Thus each parent affects the direction of each offspring velocity vector equally. In order to control that the offspring velocity was not getting too fast or too slow, the offspring velocity vector is normalized to the length of the velocity vector of one of the parent particles. The starting position of a new offspring particle is used as the initial value for this particle's best found optimum ($\vec{p}_i$).

The motivation for introducing subpopulations is to restrict the gene flow (keeping the diversity) and thereby attempt to evade suboptimal convergence. The subpopulation hybrid PSO model is an extension of the just described breeding hybrid PSO model. In this new model the particles are divided into a number of subpopulations. The purpose of the subpopulations is that each subpopulation has its own unique best known optimum. The velocity vector of a particle is updated as before except that the best known position ($\vec{p}_g$ in the formula) now refers to the best known position within the subpopulation that the particle belongs to. In terms of the neighborhood topology suggested by Kennedy in [11], each subpopulation has its own star neighborhood. The only interaction between subpopulations is, if parents from different subpopulations breed. Breeding is now possible both within a subpopulation but also between different subpopulations. An extra parameter called probability of same subpopulation breeding p_{sb} determines whether a given particle selected for breeding is to breed within the same subpopulation (probability p_{sb} equal to 0.6), or with a particle from another subpopulation (probability 1- p_{sb}). Replacing each parent with an offspring particle ensures a constant subpopulation size. The number of subpopulations used in our simulation is three with an initial population of 40 particles.

4. Results

We consider an array of 20 isotropic elements spaced 0.5 λ apart in order to generate a pencil beam and a sector beam with a common amplitude distribution and varying phase distribution. Because of symmetry, here only ten phases and ten amplitudes are to be optimized. Acceptable side lobe level (SLL) should be equal to or less than the desired value, and there are no side lobes exceeding the specified values, -40 dB for pencil beam and -30 dB for sector beam. Fig. 2 shows normalized absolute power pattern in dB for pencil beam and Fig. 3 for sector beam. After 250 iterations, the fitness value reached to its maximum, and the optimization process ended due to meeting the design goal as shown in Fig. 4. This again demonstrates the efficiency of the hybrid model. Amplitude and phase distributions in degree are shown in Tab. 1.

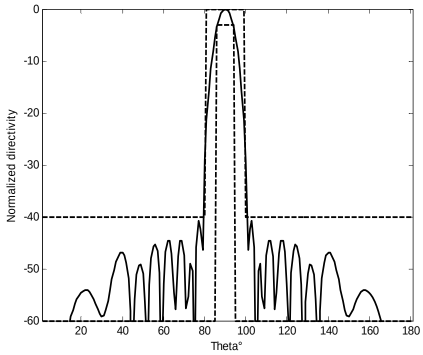


Fig. 2. Optimized pencil-shaped radiation pattern of a 20-element linear array.

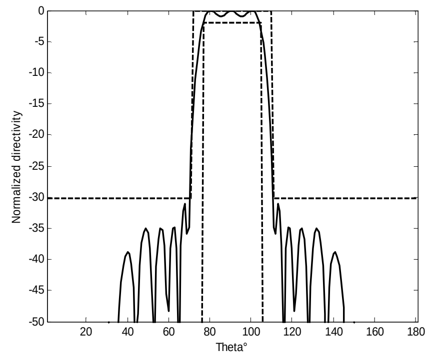


Fig. 3. Optimized sector-shaped radiation pattern of a 20-element linear array.

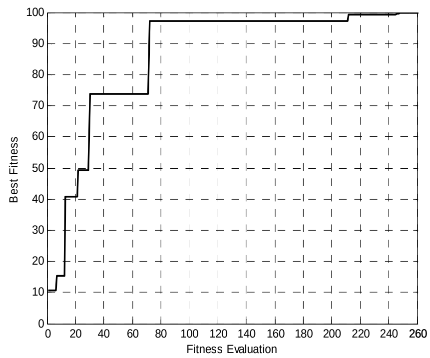
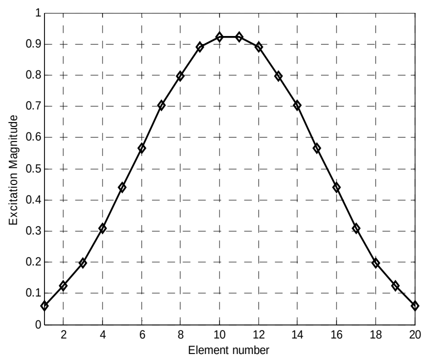


Fig. 4. Convergence curve.



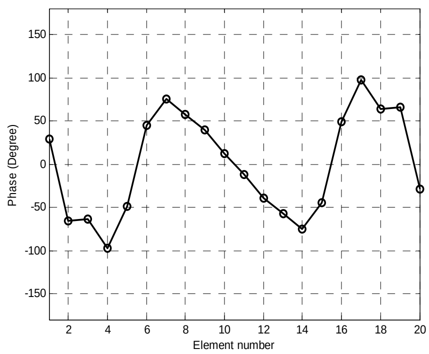


Fig. 5. The element excitation required to achieve the desired pattern.

Element N°	Pencil Beam		Sector Beam	
	Amplitude (Volt)	Phase (Degree)	Amplitude (Volt)	Phase (Degree)
1	0.0607	0.0000	0.0607	29.0547
2	0.1241	0.0000	0.1241	-65.3286
3	0.1983	0.0000	0.1983	-64.0510
4	0.3075	0.0000	0.3075	-97.5690
5	0.4390	0.0000	0.4390	-48.9306
6	0.5647	0.0000	0.5647	44.7308
7	0.7036	0.0000	0.7036	75.3726
8	0.7965	0.0000	0.7965	57.1296
9	0.89	0.0000	0.89	39.1330
10	0.9214	0.0000	0.9214	11.7284
11	0.9214	0.0000	0.9214	-11.7284
12	0.89	0.0000	0.89	-39.1330
13	0.7965	0.0000	0.7965	-57.1296
14	0.7036	0.0000	0.7036	-75.3726
15	0.5647	0.0000	0.5647	-44.7308
16	0.4390	0.0000	0.4390	48.9306
17	0.3075	0.0000	0.3075	97.5690
18	0.1983	0.0000	0.1983	64.0510
19	0.1241	0.0000	0.1241	65.3286
20	0.0607	0.0000	0.0607	-29.0547

Tab. 1. Amplitude and phase distributions.

In order to evaluate the performance of the proposed algorithm, we compare the numerical results calculated by the hybrid model, and the genetic algorithm using floating-point [12]. For comparison, a reconfigurable dual-beam linear array antenna is considered. We show the comparison of the far-field patterns among the hybrid model simulation results, and the genetic algorithm simulated results in [12]. The hybrid algorithm side lobe level is -40.43 dB and -31.01 dB for pencil and sector beams respectively, this results remain comparable to the genetic algorithm: -25.05 dB and -25.56 dB for pencil and sector beams respectively, An improvement of about 15.38 dB and 5.45 dB in the side lobe level of pencil beam and sector pattern is obtained. For the simulation convergence comparison between hybrid model and genetic algorithm, the hybrid model is run for 250 generations and the genetic algorithm is run for 1200 generations. Obviously, the hybrid algorithm is much faster than the genetic algorithm for array pattern synthesis.

This section presents a design method for planar arrays that permits control of the SLL and the beam width in the two principal planes corresponding to E plane (φ=0°) and H plane (φ=90°) respectively. As an illustrative example, we first consider the example problem that applied the hybrid model is the optimization of a 100 element planar array. Excited by a flat phase, the objects that should be optimized are the relative excitation amplitude on each element, along with the distance between elements, $\lambda/2$. In the plane φ=0°, the SLL is set to -35 dB, and in the plane φ=90° the SLL is set to -35dB too, as indicated in Fig. 6. The best fitness value returned versus the number of calls to the fitness evaluator was achieved after 200 as plotted in Fig. 7. The optimized excitations magnitudes elements according to the two axis are show in Fig. 8. Tab. 2 shows the element excitation value required to achieve this pattern. This figure shows the main beam accompanied by a decrease in the number of side lobes. In fact, the desired beam width is achieved and the specified SLL is respected.

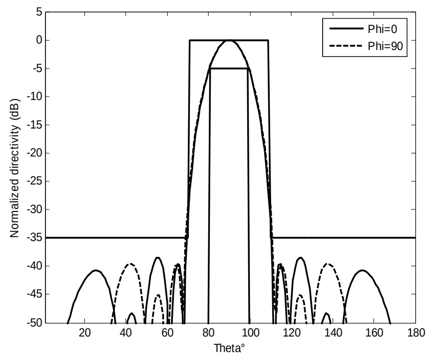


Fig. 6. Radiation pattern (both E and H plane) of a 100-element planar array.

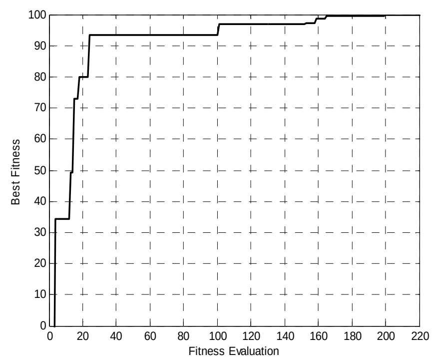
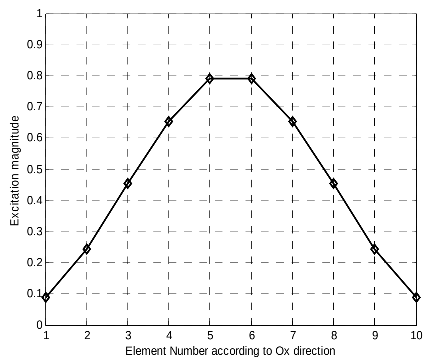


Fig. 7. Convergence curve.



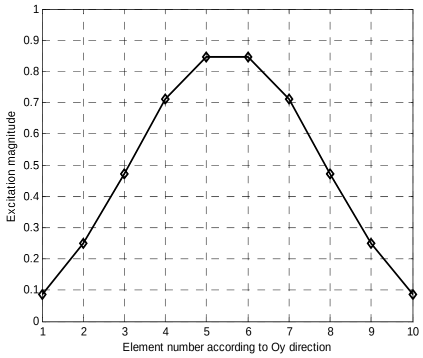


Fig. 8. The element excitation required to achieve the desired pattern.

Element N°	Amplitude (Volt)	
	Ox direction	Oy direction
1&10	0.0894	0.0855
2&9	0.2436	0.2506
3&8	0.4556	0.4733
4&7	0.6535	0.7116
5&6	0.7906	0.8468

Tab. 2. Amplitude distribution.

For the next example, we take an array with 64 elements, in amplitude-phase synthesis, the design of this array is based on finding the amplitude and phase distribution of each element, in the plane φ=0°, the SLL is set to -30dB, and in the plane φ=90° the SLL is set to -30 dB. Fig. 9 shows normalized absolute power pattern in dB, there is a very good agreement between desired and obtained results. The hybrid model is run for 200 iterations with an initial population of 40 as indicated in Fig. 10. The optimized excitation magnitudes and phases (degree) elements according to the two axis are shown in Fig. 11, and values are presented in Tab. 3. It is clear from Fig. 9 that in the shaped region, the patterns in the two planes have good performance, and there is no side lobe that exceeds the specified values. This property of the proposed design enables to choose the size (area) of the region to be covered by the main beam while keeping radiation in the other directions below a desired level.

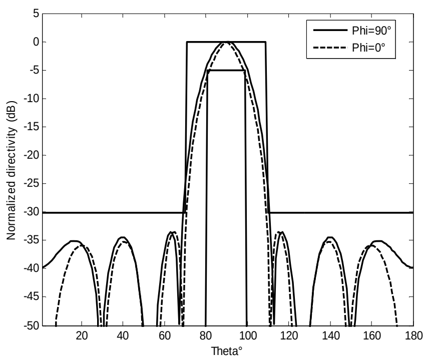


Fig. 9. Normalized absolute power pattern generated by 64 element planar array.

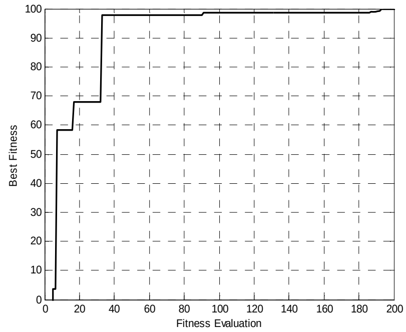


Fig. 10. Convergence curve.

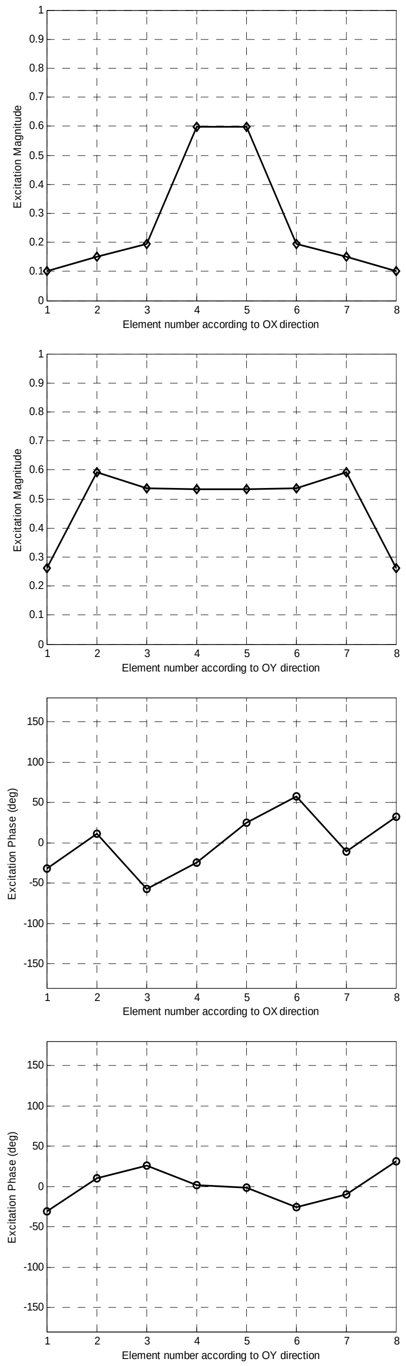


Fig. 11. The element excitation required to achieve the desired pattern.

Number N°	Amplitude (Volt)		Phase (Degree)	
	Ox direction	Oy direction	Ox direction	Oy direction
1	0.1012	0.2624	-32.332	-31.0658
2	0.1501	0.5930	11.4305	10.1471
3	0.1944	0.5363	-57.0952	25.6227
4	0.5982	0.5325	-24.5971	1.1459
5	0.5982	0.5325	24.5971	-1.1459
6	0.1944	0.5363	57.0952	-25.6227
7	0.1501	0.5930	-11.4305	-10.1471
8	0.1012	0.2624	32.332	31.0658

Tab. 3. Amplitude and phase distributions.

5. Conclusion

In this paper, Hybrid particle swarm optimizer with breeding and subpopulation is successfully used to the design of a reconfigurable dual-beam array and planar arrays. Results show that there is an agreement between the desired specifications and the synthesized one. This demonstrates the effectiveness of the proposed procedure. Advantages of this technique are ease of implementation, flexibility, accuracy, and can be very useful to antenna engineers for the pattern synthesis of antenna arrays.

References

[1] ALPHONES, A., PASSOUPATHI, V. Null steering in phased arrays by positional perturbations: a genetic algorithm approach. *Proc. IEEE Phased Arrays*, 1996, pp. 203-207.

[2] GIES, D., RAHMAT-SAMII, Y. Particle swarm optimization for reconfigurable phase differentiated array design. *Microwave and Optical Technol. Lett.*, August 2003, vol. 38, no. 3, pp. 168-175.

[3] ROBINSON, J., RAHMAT-SAMII, Y. Particle swarm optimization in Electromagnetic. *IEEE Trans. Antennas & Propagation*, Feb.2004, vol. 52, no. 2, pp. 397-407.

[4] ROBINSON, J., SINTON, S., RAHMAT-SAMII, Y. Particle swarm, genetic algorithm, and their hybrids: Optimization of a profiled corrugated horn antenna. *IEEE AP-S*, 2002, pp. 314-317.

[5] LOVBJERG, M., RASMUSSEN, T. K., KRINK, T. Hybrid particle swarm optimizer with breeding and subpopulations. In *Proceedings of the Genetic and Evolutionary Computation Conference (GECCO)*. San Francisco (USA), July 2001.

[6] MERIAH, S. M., BENDIMERAD, F.T. Low level side lobes in thinned antennas arrays by switched supply law. Optimization by genetic algorithm. *Annals of Telecommunications*. 55, N°11-12, 2000, pp. 615-621.

[7] SPEARS, W. M. Simple subpopulation schemes. In *Proceeding of the Evolutionary Programming Conference,* 1994, pp. 296-307.

[8] KENNEDY, J., EBERHART, R. C. Particle swarm optimization. In P*roceeding of the 1995 IEEE International Conference on Neural Networks*, vol. 4, p. 1942-1948.

[9] SHI, Y., EBERHART, R. C. A modified particle swarm optimizer. *IEEE International Conference on Evolutionary Computation.* Anchorage (Alaska), May 4-9, 1998.

[10] SHI, Y., EBERHART, R. C. Parameter selection in particle swarm optimization. *Evolutionary programming VII (1998), lecture notes in computer science 1447, 591-600. Springer*

[11] KENNEDY, J. Small words and mega-minds: Effects of neighbourhood topology on particle swarm performance. In P*roceedings of the 1999 Congress of Evolutionary Computation,* vol. 3, p. 1931-1938. IEEE Press.

[12] MAHANTI, G. K., DAS, S., CHAKRABORTY, A. Design of phase-differentiated reconfigurable array antennas with minimum dynamic range ratio. In *IEEE Antennas and Wireless Propagation Letters,* 2006, vol. 5, p. 262-264.

About Authors...

Hichem CHAKER was born in Ghazaouet (Algeria) in 1981. He received his degree in Electrical Engineering from the University of Tlemcen, where he is now in the Department of Electronics. He is currently working toward the Ph.D. His research interests include micro strip antenna arrays.

Sidi Mohammed MERIAH was born in Tlemcen (Algeria) in 1970. He received the diploma of state engineer in Electronics (1992) from National Polytechnic College, Magister degree in Electronics (1997) and doctorate degree from Tlemcen University (2005). He is a Senior Lecturer in Tlemcen University and Researcher within the Telecommunications Laboratory. His current research interest include numerical techniques in electromagnetic radiation, analysis and design smart antenna and array antennas pattern synthesis.

Fethi Tarik BENDIMERAD was born in Sidi Belabbes (Algeria) in 1959, he received the diploma of state engineer in Electronics (1983) from Oran University, Master degree in Electronics (1984) and doctorate degree from Nice Sophia-Antipolis University (1989). He worked as a Senior Lecturer in Tlemcen University and he is the Director of the Telecommunications Laboratory. His main area of research is the microwave techniques and radiation and he is responsible of the antenna section.