Progress In Electromagnetics Research B, Vol. 32, 169–190, 2011

FIRE FLY AND ARTIFICIAL BEES COLONY ALGORITHM FOR SYNTHESIS OF SCANNED AND BROADSIDE LINEAR ARRAY ANTENNA

B. Basu and G. K. Mahanti*

Department of Electronics and Communication Engineering, National Institute of Technology, Durgapur, Durgapur 713209, India

Abstract—This paper describes the application of two recently developed metaheuristic algorithms known as fire fly algorithm (FFA) and artificial bees colony (ABC) optimization for the design of linear array of isotropic sources. We present two examples: one for broad side arrays and the other for steerable linear arrays. Three instances are presented under each category consisting of different numbers of array elements and array pattern directions. The main objective of the work is to compute the radiation pattern with minimum side lobe level (SLL) for specified half power beam width (HPBW) and first null beam width (FNBW). HPBW and FNBW of a uniformly excited antenna array with similar size and main beam directions are chosen as the beam width constraints in each case. Algorithms are applied to determine the non-uniform excitation applied to each element. The effectiveness of the proposed algorithms for optimization of antenna problems is examined by all six sets of antenna configurations. Simulation results obtained in each case using both the algorithms are compared in a statistically significant way. Obtained results using fire fly algorithm shows better performances than that of artificial bees colony optimization technique provided that the same number of function evaluations has been considered for both the algorithms.

1. INTRODUCTION

To meet the ever-expanding demand of wireless mobile communication, different techniques are explored to maximize the spectral efficiency of the mobile network. The performances of the broadcasting system can be improved by controlling different radiation characteristics as pattern

Received 31 May 2011, Accepted 5 July 2011, Scheduled 14 July 2011
* Corresponding author: Gautam Kumar Mahanti (gautammahanti@yahoo.com).

main beam width, minimum achievable side lobe level, directivity, etc. However, there is a tradeoff among these parameters [1, 2]. Radiation patterns having minimal beam width and side lobe level fulfill the high gain demand of the wireless communication systems as well as reduce the inter-channel interferences. Furthermore the use of steerable linear array improves mobile network capacity and quality with better efficiency. A steerable linear array is an antenna array, in which the phases of the signals arriving to the antenna elements are controlled progressively, so that the direction of the radiation pattern can be steered accordingly [3–5]. Patterns with minimal beam width and side lobe level are steered toward intended user direction in order to reduce the interferences from other users and thereby provide better mobile coverage to the network. There exist different stochastic techniques to solve these constrained multi-objective antenna-designing problems [6]. Simulated annealing [7, 8], genetic algorithm [9], ant colony optimization [10], particle swarm optimization [11–13], invasive weed optimization [14], Taguchi's optimization method [15, 16], differential evolution [17–19] are some of the well-known algorithms used to model and solve different antenna design problems.

Numerous new algorithms, which are successfully used to solve complex computational problems in real world, come out of this growing interest in optimization algorithms.

In this paper, we applied two relatively new optimization algorithms, namely, fire fly algorithm and artificial bees colony algorithm [20–24] to design the linear array of isotropic sources. Algorithms are used to approximate the element excitation in order to minimize the side lobe level where HPBW and FNBW of the pattern are kept within specified constraints. Two examples are considered: in the first, pattern maxima are directed toward 90°, and in the second, patterns are steered to different scan angles from broadside. Six numerical instantiations of the design problem have been used to illustrate the application of the FFA and ABC. We incorporate an adaptive strategy for tuning the algorithm parameter α with iterations in FFA for enhancing the algorithm performances. Simulation results obtained using both the algorithms are compared. Detailed analysis of the optimal solutions obtained in each case reflects the superiority of FFA over ABC.

2. PROBLEM FORMULATION

A linear array of N isotropic elements positioned along z-axis and separated by a distance d is presented in Figure 1. The free space far

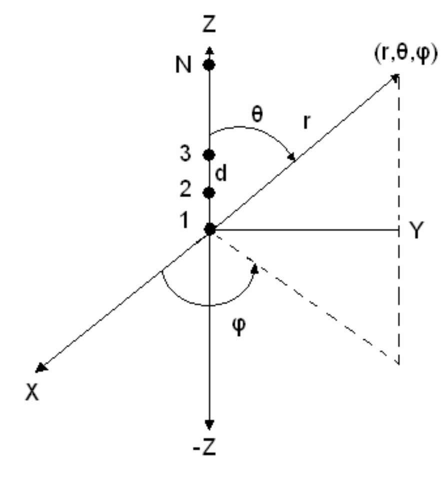


Figure 1. Linear array of isotropic sources along z-axis.

field pattern in vertical plane can be expressed using Equation (1).

$$F(\theta) = \sum_{n=1}^{N} I_n \exp(j(n-1)(kd\cos\theta + \beta)) \tag{1}$$

where I_n is the element excitation; $k = 2\pi/\lambda$ = wave number; d = inter-element spacing = 0.5λ; λ = wave length; θ = polar angle measured from z-axis; β is the progressive phase; and N is the total number of elements in the array.

In broadside case, pattern maxima is directed toward $\theta_d = 90°$, i.e., $\beta = 0°$. However, in scanned array, pattern maxima is oriented at an angle θ_d, so the progressive phase difference β between the elements is $-kd\cos\theta_d$ [25].

Normalized power pattern in dB can be expressed as follows.

$$P(\theta) = 10\log_{10}\left[\frac{|F(\theta)|}{|F(\theta)|_{\max}}\right]^2 = 20\log_{10}\left[\frac{|F(\theta)|}{|F(\theta)|_{\max}}\right] \tag{2}$$

The problem is to optimize the weights of the individual array elements placed 0.5λ apart for the minimization of SLL while HPBW and FNBW both are kept within some specified constraints. We consider three arrays consisting 20, 26 and 30 elements. In the first example, amplitude weight of individual element is controlled to generate patterns in broadside. The phase of each element is kept at zero. In the second example, individual amplitude weight is controlled for having maxima at 30° for 20 elements array, 45° for 26 elements array, and

60° for 30 elements array. Progressive phase differences between the elements steer the beam toward the desired direction. Thus combining all the objectives we formulate the cost function as

$$Cost = \begin{cases} \max(SLL) & \text{if} \quad X <= X_u \quad \text{and} \quad Y <= Y_u \\ 10^4 & \text{otherwise} \end{cases} \tag{3}$$

where X and Y are the half power and first null beam width of the pattern produced by the array considered for optimization. X_u and Y_u are the corresponding values obtained with the uniformly excited array of similar size and same main beam direction.

3. OVERVIEW OF FIRE FLY ALGORITHM

A branch of nature inspired algorithms, which are known as swarm intelligence, is focused on insect behavior in order to develop some meta-heuristics, which can mimic insect's problem solution abilities. Ant colony optimization, particle swarm optimization, invasive weed optimization etc. are some of the well-known algorithms that mimic insect behavior in problem modeling and solution. Fire fly algorithm is a relatively new member of swarm intelligence family [20, 21].

The presented technique is inspired by social behavior of fireflies and the phenomenon of bioluminescent communication. Fireflies are able to produce light due to the presence of photogenic organs situated very close to their body surface. Bioluminescent signal is shown to attract the attention of the partners as a part of their courtship rituals. Brighter flies are capable of attracting more attention. This swarm behavior is successfully employed in constrained optimization problem. We take the analogy of firefly and perceive each possible solution of the optimization problems as firefly. The intensity of the light signal emitted from each fly determines its ability to explore an efficient new solution.

I. Initialize Swarm: The algorithm uses swarm of N d dimensional parameter vectors as a population for each generation. The initial population x_i $(i = 1, 2, \ldots, N)$ is chosen randomly within the specified search space bounded by a maxima and minima u_b and l_b, respectively.

$$x_i(t) = lb + (ub - lb)rand1_i\,(1, d) \tag{4}$$

$rand1\,(1, d)$ is an 1-by d vector with uniformly distributed random numbers between 0 and 1.

II. Strategy for Searching New Swarms: Each parameter vector x_i is characterized by its attractiveness and the light intensity emitted by itself. x_i explores new population in its neighborhood within the specified search space. Attractiveness of x_i is determined using a monotonically decreasing function described by

$$\beta(t) = (\beta_0 - \beta_{\min})e^{-\gamma r_j(t)^2} + \beta_{\min} \tag{5}$$

where β_0 is the maximum attractiveness (a predetermined constant); γ is the absorption coefficient; and $r_j(t)$ is the distances between the i-th and an arbitrarily chosen j-th parameter vector where $j \in \{1, 2, \ldots, N\}$ and $j \neq i$. Light intensity I_i of any vector is proportional to the inverse of the cost function value produced by it.

$$I_i(t) = 1/f(x_i(t)) \tag{6}$$

New solution is generated by the attractiveness of the swarm member with higher intensity, i.e., i-th solution changes its position if there exists any x_j such that $I_j > I_i$ where $j \in \{1, 2, \ldots, N\}$ and $j \neq i$. New population is produced adding two weighted vectors, x_i and x_j, with a random step size u_i.

$$x_i(t+1) = (1-\beta)x_i(t) + \beta x_j(t) + u_i \tag{7}$$

If no brighter solution is found then FFA creates new solution perturbing the best found solution by a random step size u_i.

$$x_i(t+1) = x_i(t) + u_i \tag{8}$$

where

$$u_i = \alpha(rand2_i(1,d) - 0.5)\{abs(ub - lb)\} \tag{9}$$

Random step size u_i has a lower and upper bounds and depends on the algorithm parameter α. According to Equation (9) the absolute value of the range of the dynamic search space adds an envelope to u_i. It prevents the step size exceeding the specified solution space. Here $rand2\,(1,d)$ is an 1-by-d vector with uniformly distributed random numbers between 0 and 1.

In standard fire fly algorithm the value of the control parameter α is chosen as a real constant and $\alpha \in (0,1)$. Here we use a modification scheme for tuning the value of α in order to enhance the algorithm performance. It is seen that α is used to perturb any solution from its previous position. With the initial populations, we keep on exploring new solutions with adequate population diversity. However, as the algorithm approaches to its optima with iterations it is expected to suffer lesser perturbation so that it may undergo a fine search within a small neighborhood. Thus we need a balance between exploration and exploitation to prevent premature convergence. This is achieved by controlling the randomness of the solution vectors by tuning the parameter α with iteration. Equation (10) presents a scheme that ensures the nonlinear modification of α with iterations.

$$\alpha_{i+1} = \delta^{1/i}\alpha_i \tag{10}$$

where $\delta = 5.6 \times 10^{-3}$ and i is the current iteration number.

Thus the algorithm makes use of synergic local searches. Each member of the swarm takes into account the results obtained by other swarm members as well as applies its own randomized move to explore the problem space.

III. Selection: The new solution is compared with the present one using greedy criterion. If the new solution yields lower fitness value than the present one then it is selected for the next generation; otherwise the older value is retained.

In this way, the steps are repeated until the maximum number of iterations is reached, or a predefined optimization criterion satisfies.

4. OVERVIEW OF ARTIFICIAL BEES COLONY ALGORITHM

Artificial Bee Colony (ABC) proposed by Karaboga et al. [22, 23] is another swarm based metahuristic algorithm used to solve and model different optimization problems [24].

ABC tries to model natural behavior of real honey bees in food foraging [22]. The minimal model of forage selection that leads to the emergence of collective intelligence of honey bee swarms consists of two essential components: employed and unemployed foragers. Employed bees are associated with a particular food source, which they are currently exploiting. They carry the information about this particular source and share this information with a certain probability by waggle dance. Unemployed bees seek a food source to exploit. There are two types of unemployed bees: scouts and onlookers. Scouts search the environment for new food sources without any guidance. Occasionally, the scouts discover rich, entirely unknown food sources. On the other hand, onlookers observe the waggle dance and place themselves accordingly on the food sources by using a probability based selection process. As the nectar amount of a food source increases, the probability value with which the food source is preferred by onlookers increases. In the algorithm, the first half of the colony consists of the employed bees, and the second half includes the onlookers. For every food source, there is only one employed bee. Another issue that is considered in the algorithm is that the employed bee whose food source has been exhausted by the bees becomes a scout. In other words, if a solution representing a food source is not improved by a predetermined number of trials, then the food source is abandoned by its employed bee, and the employed bee is converted to a scout. The main steps of the algorithm are detailed below.

I. Initialization: ABC begins with a randomly initiated population of

SN D dimensional real-valued parameter vectors. Each vector forms a candidate solution to the multi-dimensional optimization problem. We shall denote subsequent cycles in ABC by $C = 0, 1, \ldots, MCN$.

Since the parameter vectors are likely to be changed over different cycles, we may adopt the following notation for representing the i-th vector of the population at the current generation:

$$x_{i,j} = [x_{i,1}, x_{i,2}, \ldots, x_{i,D}]$$

The initial population should cover the entire search space as much as possible by uniformly randomizing individuals within the search space constrained by the prescribed minimum and maximum bounds

$$x_i^j = x_{\min}^j + rand(0,1)\left(x_{\max}^j - x_{\min}^j\right) \tag{11}$$

where $x_{\min}^j$ and $x_{\max}^j$ are the lower and upper bound of the parameter j and $rand\,(0, 1)$ is a uniformly distributed random number between 0 and 1.

II. Exploiting New Solution: After initialization ABC exploits a new vector $v_{i,G}$, corresponding to each population member $x_{i,G}$ in the current generation through Equation (12)

$$v_{ij} = x_{ij} + \phi_{ij}(x_{ij} - x_{kj}) \tag{12}$$

where $k \in \{1, 2, \ldots, SN\}$ and $j \in \{1, 2, \ldots, D\}$ are randomly chosen indexes. Although k is determined randomly, it has to be different from i. $\varphi_{i,j}$ is a random number between -1 to 1. If the difference between the parameters of the $x_{i,j}$ and $x_{k,j}$ decreases, the perturbation on the position $x_{i,j}$ reduces. Thus, as the search approaches to the optimum solution, the step length is adaptively reduced so that it may undergo a fine search within a small neighborhood of the suspected optima. If a parameter value produced by this operation exceeds its preset limit, the parameter is reset to an acceptable value within that limit.

III. Selection: The next step of the algorithm calls for selection to determine whether or not the new vector should become a member of the next cycle $C = C + 1$.

The key sense of this selection is that the probability value p_i associated with the new solution $v_{i,j}$ is compared to that of the current population $x_{i,j}$ using the greedy criterion. If vector $v_{i,j}$ yields a higher p_i value than that of $x_{i,j}$, then $v_{i,j}$ is selected for the next generation; otherwise, the old value $x_{i,j}$ is retained. Probability associated with every solution is evaluated using the following expression (13)

$$p_i = \frac{fit_i}{\sum\limits_{n=1}^{SN} fit_n} \tag{13}$$

where fit_i is the fitness value of the solution i and calculated using Equation (14)

$$fit_i = \begin{cases} \frac{1}{1+f_i} & J_i \geq 0 \\ 1 + abs(f_i) & J_i < 0 \end{cases} \tag{14}$$

where f_i is the objective function value. Since designing problem is a minimization problem, a lower objective function value corresponds to higher fitness.

Table 1. Amplitude excitations obtained using FFA.

Category of the Array	Current amplitude distribution
$N = 20$ $\theta = 90°$	1, 0.27533, 0.49337, 0.49491, 0.62565, 0.53222, 0.58971, 0.78975, 0.6154, 0.81085, 0.46915, 0.61018, 0.81272, 0.59576, 0.4942, 0.39527, 0.61523, 0.43186, 0.70456,1
$N = 26$ $\theta = 90°$	0.69883, 0.67672, 0.4343, 0.25208, 0.59818, 0.23402, 0.71888, 0.40806, 0.72573, 0.40653, 0.56536, 0.46627, 0.52075, 0.76734, 0.58851, 0.5672, 0.73249, 0.51647, 0.33335, 0.61611, 0.58636, 0.033828, 0.71809, 0.28593, 0.7445, 0.88334
$N = 30$ $\theta = 90°$	0.99978, 0.55871, 0.61177, 0.10017, 0.4007, 0.89422, 0.16041, 0.53909, 0.41372, 0.75962, 0.38097, 0.61658, 0.57319, 0.97661, 0.71209, 0.39707, 0.38046, 0.69926, 0.66505, 0.65249, 0.40944, 0.61641, 0.75027, 0.46374, 0.17428, 0.49829, 0.42106, 0.79832, 0.52087, 0.92743
$N = 20$ $\theta = 30°$	0.98041, 0.76626, 0.36907, 0.55297, 0.90715, 0.20192, 0.51964, 0.84496, 0.50948, 0.98057, 0.51426, 0.53871, 0.80273, 0.55406, 0.88082, 0.40378, 0.33214, 0.46556, 0.50348, 0.94604
$N = 26$ $\theta = 45°$	1, 0.72429, 0.55905, 0.44837, 0.71979, 0.31947, 0.70758, 0.62037, 0.53995, 0.86304, 0.67322, 0.71588, 0.83498, 0.77957, 0.42717, 0.79538, 0.71365, 0.63019, 0.62672, 0.6301, 0.74732, 0.060126, 0.73871, 0.59844, 0.77826, 0.9975
$N = 30$ $\theta = 60°$	0.99575, 0.68442, 0.62997, 0.049937, 0.17937, 0.73457, 0.48525, 0.61814, 0.33365, 0.63189, 0.63643, 0.39343, 0.49183, 0.77247, 0.64547, 0.48407, 0.73964, 0.74411, 0.52797, 0.45014, 0.82218, 0.52901, 0.45825, 0.41905, 0.48686, 0.24166, 0.86684, 0.63618, 0.29691, 0.99934

5. SIMULATION RESULTS

Two examples of broad side and steerable linear arrays are presented in order to illustrate the proposed technique. We consider six instantiations with different numbers of array elements ($N = 20$, 26, 30). In the first example, all the three arrays with 20, 26, 30 elements are optimized in broadside direction. In the second example, three array configurations are optimized for different main lobe directions ($\theta_d = 30°$ for $N = 20$, 45° for $N = 26$, 60° for $N = 30$).

Table 2. Amplitude excitations obtained using ABC.

Category of the Array	Current amplitude distribution
$N = 20$ $\theta = 90°$	0.98241, 0.5487, 0.33293, 0.50042, 0.53676, 0.59935, 0.36735, 0.66119, 0.81346, 0.70536, 0.66457, 0.36531, 0.99167, 0.54632, 0.68836, 0.59763, 0.39796, 0.5254, 0.73636, 0.88989
$N = 26$ $\theta = 90°$	0.98985, 0.67187, 0.65793, 0.69976, 0.66963, 0.19943, 0.78815, 0.59953, 0.6124, 0.98883, 0.69772, 0.7644, 0.95048, 0.76006, 0.5575, 0.84152, 0.67148, 0.65134, 0.78077, 0.83281, 0.64491, 0.23598, 0.60915, 0.82725, 0.81795, 0.99999
$N = 30$ $\theta = 90°$	1, 0.55024, 0.54803, 0.37073, 0.38809, 0.69758, 0.14595, 0.66133, 0.318, 0.76438, 0.40833, 0.50108, 0.62223, 0.96195, 0.65987, 0.45627, 0.36411, 0.68257, 0.62436, 0.61087, 0.38545, 0.60515, 0.86564, 0.43131, 0.2224, 0.39246, 0.29024, 0.79482, 0.49202, 0.90367
$N = 20$ $\theta = 30°$	0.99019, 0.97498, 0.4316, 0.40012, 0.6937, 0.67015, 0.86845, 0.54775, 0.98904, 0.55144, 0.84704, 0.93989, 0.61825, 0.64675, 0.82054, 0.42283, 0.77915, 0.39712, 0.78957, 0.99533
$N = 26$ $\theta = 45°$	0.81896, 0.81102, 0.62887, 0.47964, 0.50314, 0.34313, 0.57283, 0.24977, 0.85275, 0.757, 0.28683, 0.9754, 0.46025, 0.50043, 0.69194, 0.77689, 0.34013, 0.42986, 0.6001, 0.49268, 0.7131, 0.36205, 0.34544, 0.089086, 0.72293, 0.89501
$N = 30$ $\theta = 60°$	0.99705, 0.67314, 0.63535, 0.19672, 0.19138, 0.75192, 0.48636, 0.61824, 0.39223, 0.60121, 0.65837, 0.38652, 0.60241, 0.74725, 0.74974, 0.4252, 0.83573, 0.64294, 0.59721, 0.48109, 0.83104, 0.57104, 0.44958, 0.43735, 0.5144, 0.23821, 0.86101, 0.79141, 0.24175, 0.99392

FFA and ABC both are used to compute the non-uniform excitation distribution applied to each element to minimize SLL keeping the HPBW and FNBW within a specified value corresponding

Table 3. Results for Broadside array with unequal excitation and unity excitation using FFA.

Design parameters	$N = 20$ Unequal excitation	$N = 20$ Unity excitation	$N = 26$ Unequal excitation
Max SLL (dB)	−15.57	−13.189	−15.84
HPBW (degree)	5.0	5.2	4.0
FNBW (degree)	11.4	11.4	8.8
Directivity	18.36	19.99	22.98
Design parameters	**$N = 26$ Unity excitation**	**$N = 30$ Unequal excitation**	**$N = 30$ Unity excitation**
Max SLL (dB)	−13.219	−16.02	−13.23
HPBW (degree)	4.0	3.4	3.4
FNBW (degree)	8.8	7.6	7.6
Directivity	25.99	25.92	29.99

Table 4. Results for broadside array with unequal excitation and unity excitation using ABC.

Design parameters	$N = 20$ Unequal excitation	$N = 20$ Unity excitation	$N = 26$ Unequal excitation
Max SLL (dB)	−15.56	−13.189	−15.63
HPBW (degree)	5.0	5.2	4.0
FNBW (degree)	11.4	11.4	8.8
Directivity	18.32	19.99	24.3
Design parameters	**$N = 26$ Unity excitation**	**$N = 30$ Unequal excitation**	**$N = 30$ Unity excitation**
Max SLL (dB)	−13.219	−16	−13.23
HPBW (degree)	4.0	3.4	3.4
FNBW (degree)	8.8	7.6	7.6
Directivity	25.99	26.16	29.99

Table 5. Results for scanned array with unequal excitation and unity excitation using FFA.

Design parameters	$N = 20$, $\theta = 30^\circ$ Unequal excitation	$N = 20$, $\theta = 30^\circ$ Unity excitation	$N = 26$, $\theta = 45^\circ$ Unequal excitation
Max SLL (dB)	-15.59	-13.19	-15.61
HPBW (degree)	10.2	10.4	5.5
FNBW (degree)	25.0	25.0	12.5
Directivity	17.6	19.99	23.9
Design parameters	$N = 26$, $\theta = 45^\circ$ Unity excitation	$N = 30$, $\theta = 60^\circ$ Unequal excitation	$N = 30$, $\theta = 60^\circ$ Unity excitation
Max SLL (dB)	-13.21	-15.97	-13.23
HPBW (degree)	5.6	4.0	4.0
FNBW (degree)	12.5	8.8	8.8
Directivity	25.99	26.05	29.99

Table 6. Results for scanned array with unequal excitation and unity excitation using ABC.

Design parameters	$N = 20$, $\theta = 30^\circ$ Unequal excitation	$N = 20$, $\theta = 30^\circ$ Unity excitation	$N = 26$, $\theta = 45^\circ$ Unequal excitation
Max SLL (dB)	-15.53	-13.19	-15.59
HPBW (degree)	10.2	10.4	5.5
FNBW (degree)	25.0	25.0	12.5
Directivity	18.48	19.99	22.53
Design parameters	$N = 26$, $\theta = 45^\circ$ Unity excitation	$N = 30$, $\theta = 60^\circ$ Unequal excitation	$N = 30$, $\theta = 60^\circ$ Unity excitation
Max SLL (dB)	-13.21	-15.8	-13.23
HPBW (degree)	5.6	4.0	4.0
FNBW (degree)	12.5	8.8	8.8
Directivity	25.99	26.43	29.99

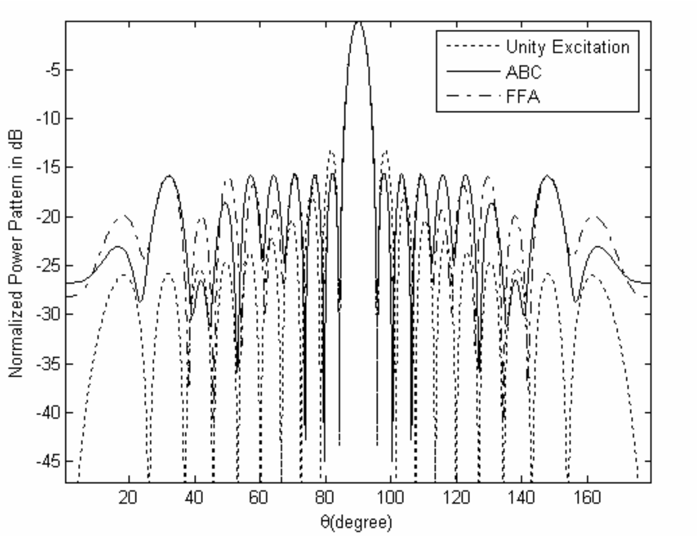
to a uniformly excited array of same size and main lobe direction. The phase is kept at zero degree for broad side array. However, a progressive phase difference ($-kd\cos\theta_d$) between the elements is used for scanning the array.

The current amplitude used to excite the array elements for all six instantiations using each algorithm are presented in Table 1 and Table 2. We used normalized current amplitude distributions where the maximum value of the current amplitude is set to 1.

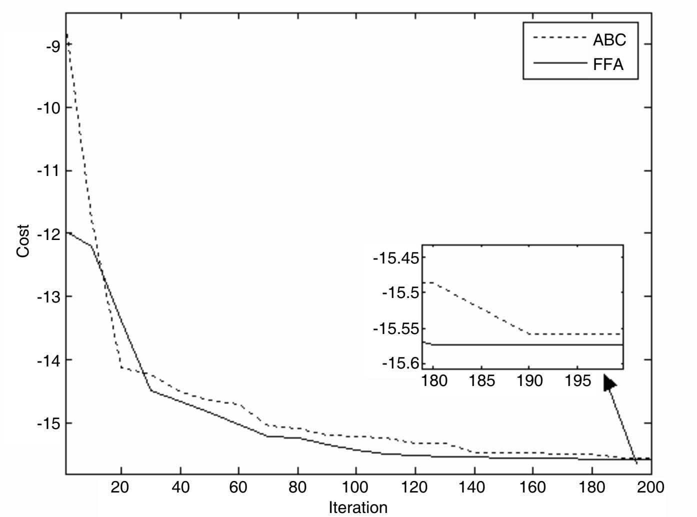
Results obtained for six configurations with FFA and ABC are tabulated in Table 3 and Table 4. Each array is analyzed with uniform as well as non-uniform excitation distributions (optimized set of data using FFA and ABC), and their performances are compared. It is seen that optimized set of excitation distribution yields lower SLL values than that of the corresponding array with unity excitation. However, as the number of array elements increases the improvement of the SLL performances under the constraints of HPBW and FNBW is not very significant. It is also observed that beams steered at different directions offer broader HPBW and FNBW compared to those of the broadside

Table 7. Results obtained using FFA and ABC.

Algorithms		FFA	ABC
$N = 20$	Average Cost function	-15.3935	-15.2211
	Standard Deviation	0.1152	0.1280
	P-value	NA	1.2800e-004
$N = 26$	Average Cost function	-15.6385	-15.2451
	Standard Deviation	0.1125	0.1356
	P-value	NA	3.3812e-007
$N = 30$	Average Cost function	-15.8035	-15.5201
	Standard Deviation	0.1187	0.2079
	P-value	NA	5.5132e-005
$N = 20$ at 30°	Average Cost function	-15.3850	-15.1970
	Standard Deviation	0.1332	0.1247
	P-value	NA	7.7410e-005
$N = 26$ at 45°	Average Cost function	-15.4370	-15.2645
	Standard Deviation	0.1184	0.1267
	P-value	NA	1.8744e-004
$N = 30$ at 60°	Average Cost function	-15.8225	-15.6775
	Standard Deviation	0.0686	0.0769
	P-value	NA	3.4192e-006



(a)



(b)

Figure 2. (a) Normalized Power Patterns with 20 elements broadside array using FFA and ABC. (b) Convergence curves of 20 elements broadside array using FFA and ABC.

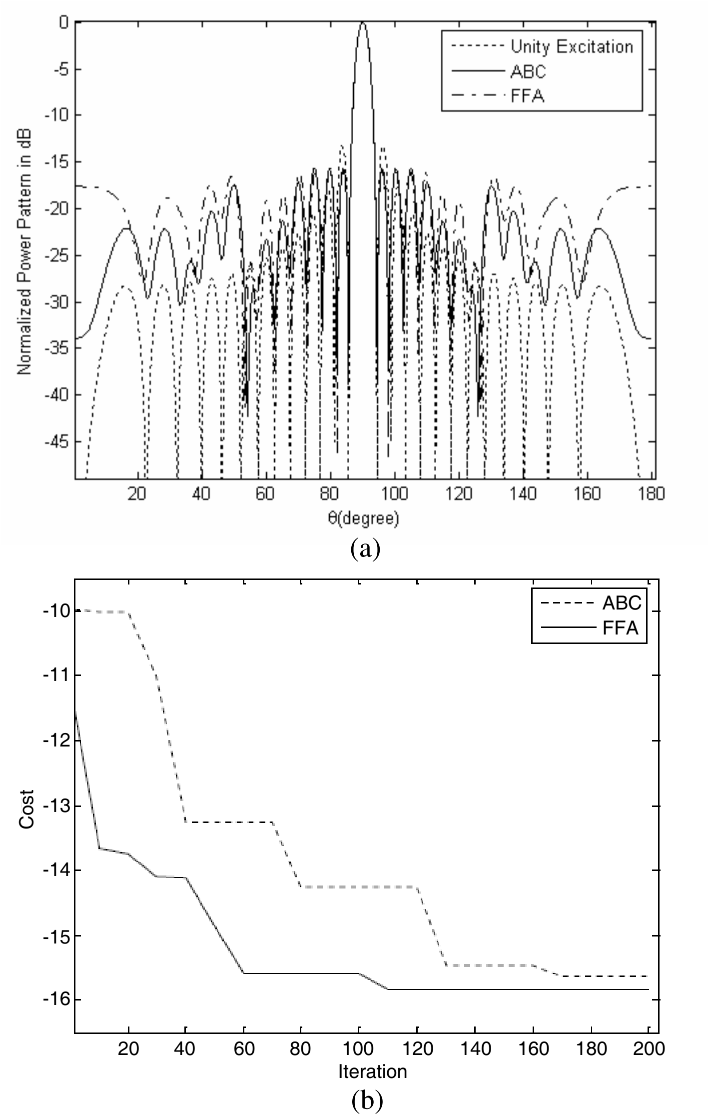


Figure 3. (a) Normalized Power Patterns with 26 elements broadside array using FFA and ABC. (b) Convergence curves of 26 elements broadside array using FFA and ABC.

linear arrays. The dissimilarity reduces as the scan angle approaches to 90°. SLL performance too degrades slightly due to scanning. We compute the directivity of all the array configurations with uniform and non-uniform excitations. However, maximization of directivity is not included in the optimization task.

For numerical experiments, the control parameters for FFA are set as suggested by [20, 21]: The maximum and minimum attractiveness β_0 and $\beta_{\min}$ are set at 1 and 0.2, respectively. We use a time varying algorithm parameter α with initial value 0.25. Absorption coefficient γ is set at 1. The algorithm is run for 200 iterations. A swarm size of 20 is used for the experiment where dynamic range of search space is bounded within (0, 1). To accelerate the convergence, α is adapted with iterations according to Equation (10).

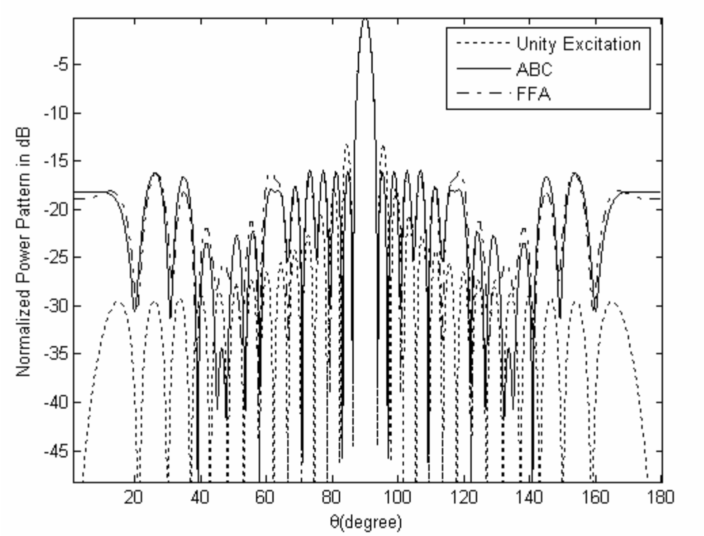
The parameters for ABC are set following the guidelines provided by [22–24]: The number of food sources which is equal to the number of employed or onlooker bees (SN) is chosen as 20; the value of limit is set to 25; and the maximum cycle number (MCN) used for the optimization is 200.

In an attempt to make a fair comparison between FFA and ABC, we use a similar number of function evaluations for both the cases ($20 \times 200 = 4000$).

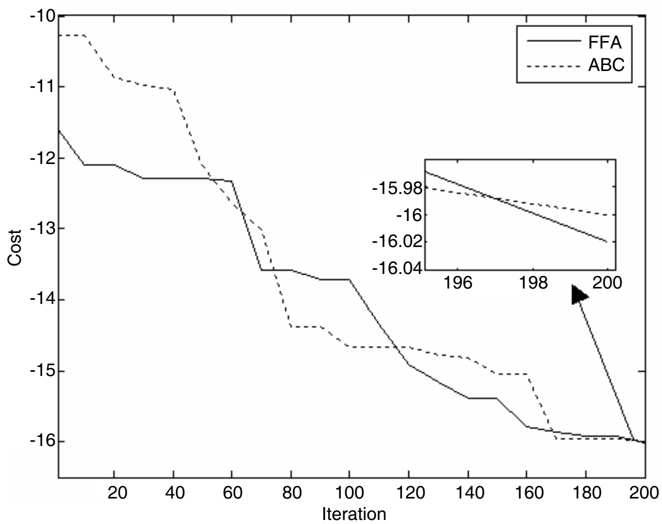
Table 7 compares the quality of the optimal solutions achieved in terms of the mean and standard deviation of the best results for 20 independent run using FFA and ABC. As the distributions of the best objective function values do not follow a normal distribution, the Wilcoxon two-sided rank sum test [26–28] was performed to compare the objective function values. Table 7 shows FFA produces smaller mean cost values over the six design instances. So we consider FFA as the best performing algorithm. As the P-values obtained through the Wilcoxon's rank sum test between the two algorithms for all the six configurations are less than 0.05, null hypothesis is rejected at the 5% significance level. It indicates that the better final cost value achieved by the best algorithm in each case is statistically significant and does not occur by chance. Here p values appear as NA stands for "Not Applicable" and occur for the best performing algorithm itself in each case.

Figures 2–4 ((a) only) depict the normalized radiation patterns of the broadside linear arrays with 20, 26 and 30 elements using FFA and ABC. Figures 5–7 ((a) only) present normalized radiation patterns of the 20 elements linear array steered at 30°, 26 elements linear array steered at 45°, 30 elements linear array steered at 60° using FFA and ABC.

Convergence characteristics of FFA and ABC over six designing



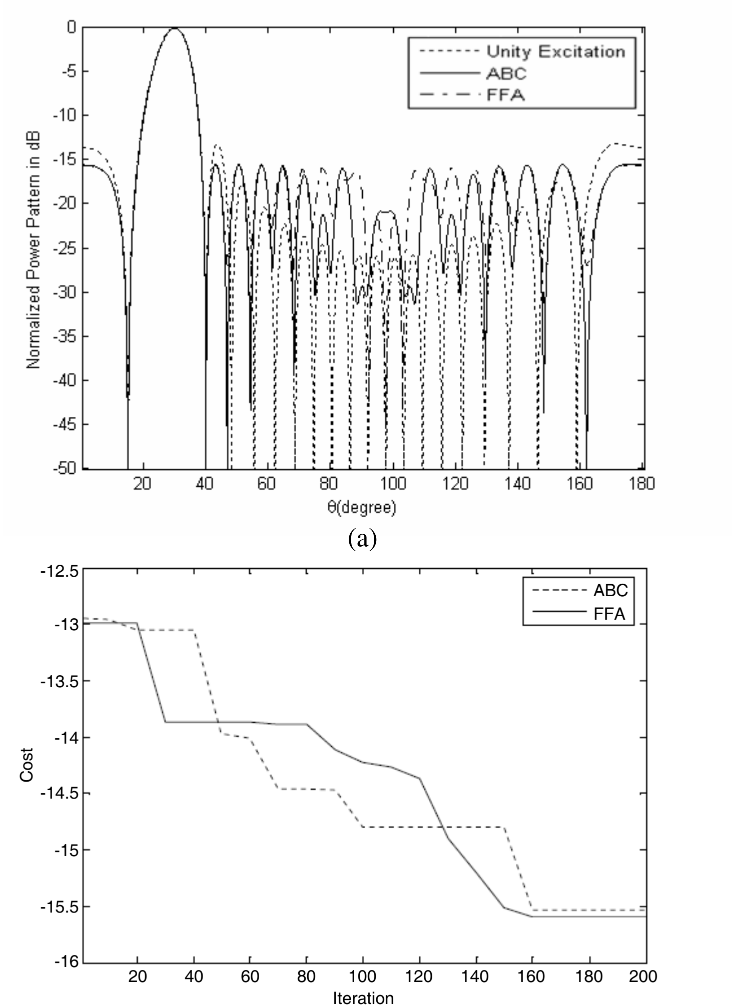
(a)



(b)

Figure 4. (a) Normalized Power Patterns with 30 elements broadside array using FFA and ABC. (b) Convergence curves of 30 elements broadside array using FFA and ABC.

instances are plotted in Figures 2–7 ((b) only). It is seen that FFA attains the minimum cost function value in each case taking less number of function evaluations. Moreover, it is seen that FFA takes less computation time to reach the minimum cost function value in each case compared to that of ABC.



(a)

(b)

Figure 5. (a) Normalized Power Patterns steered at 30° with 20 elements array using FFA and ABC. (b) Convergence curves of the scanned array steered at 30° with 20 elements using FFA and ABC.

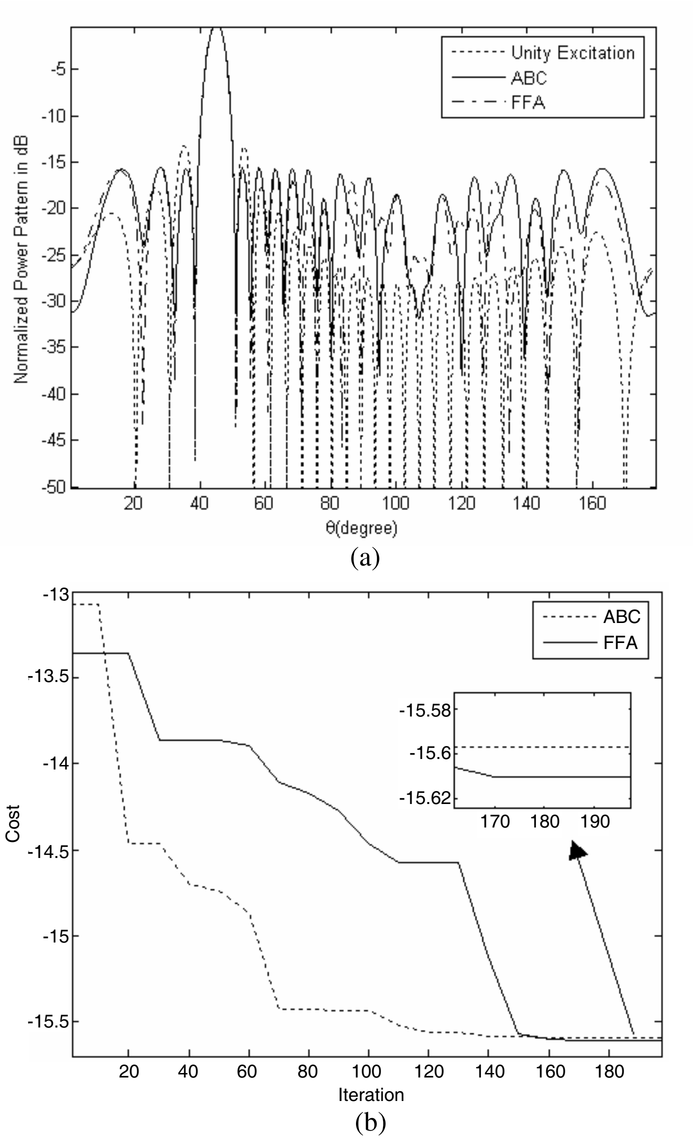
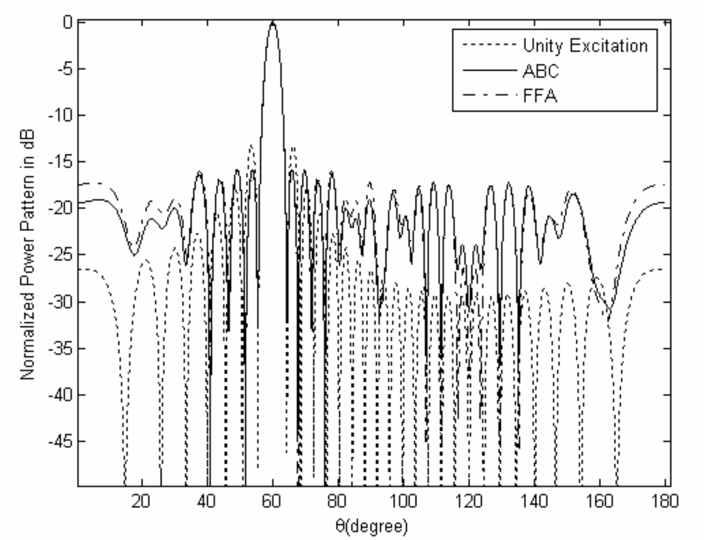
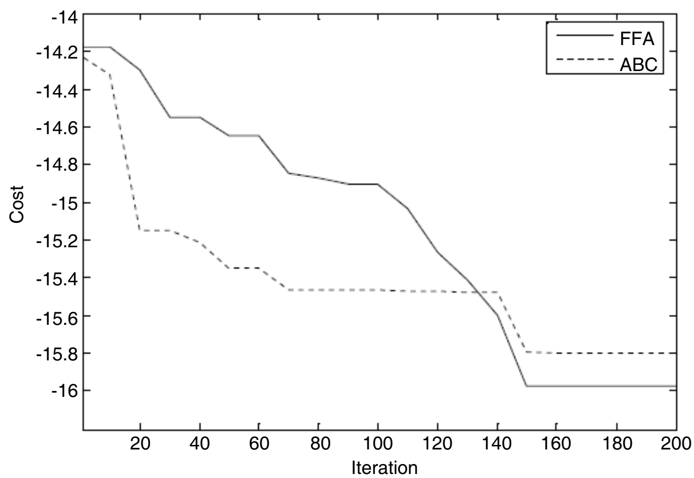


Figure 6. (a) Normalize Power Patterns steered at 45° with 26 elements array using FFA and ABC. (b) Convergence curves of the scanned array steered at 45° with 26 elements using FFA and ABC.



(a)



(b)

Figure 7. (a) Normalized Power Patterns steered at 60° with 30 elements array using FFA and ABC. (b) Convergence curves of the scanned array steered at 60° with 30 elements using FFA and ABC.

6. CONCLUSIONS

It is observed from the above study that the angular widths of the main beams are inversely related with the length of the array. Moreover, for a given length array, HPBW and FNBW both increase as the side lobe level is lowered. So the design specification must have a tradeoff between these conflicting array parameters. Thus the problem can be referred as multi-objective. We used two relatively new swarm based optimization techniques, namely FFA and ABC, to solve this multi-objective designing problem. Algorithms efficiently compute the non-uniform current distribution on each element for minimizing side lobe level while HPBW and FNBW are not allowed to exceed the subsequent values obtained with unity current excitation distribution. Effectiveness of the proposed algorithms is examined in various array configurations having different lengths and array pattern directions (broad side and steerable). Results suggest an improvement in side lobe level using optimized set of non-uniform current distribution. However, as the number of radiators increases we find a little improvement in SLL value under the constraints of HPBW and FNBW. For steerable linear arrays, as we move away from the broadside direction main beam broadens. There is a good agreement between the obtained and desired results using FFA and ABC. However, it is seen that FFA outperforms ABC in terms of convergence and cost minimization in a statistically meaningful way. The future research may opt for solving more complex designing problems using fire fly algorithm.

REFERENCES

1. Stutzman, W. L. and G. A. Thiele, *Antenna Theory and Design*, 2nd edition, Wiley, Chichester, New York, Brisbane, Singapore, Toronto, 1998.
2. Panduro, M. A., D. H. Covarrubias, C. A. Brizuela, and F. R. Marante, "A multi-objective approach in the linear antenna array design," *AEU — International Journal of Electronics and Communications*, Vol. 59, No. 4, 205–212, 2005.
3. EPFL, "Smart antenna systems for mobile communications," *Technical Report EPFL CH-1015*, Ecole Polytechnique Fédérale de Laussane, 2003.
4. Liberti, J. C. and T. S. Rappaport, *Smart Antennas for Wireless Communications: IS-95 and Third Generation CDMA Applications*, Prentice Hall, New Jersey, 1999.
5. Panduro, M., D. H. Covarrubias, and C. Brizuela, "Design

of electronically steerable linear arrays with evolutionary algorithms," *Applied Soft Computing*, Vol. 8, 46–54, 2008.

6. Eiben, A. E. and J. E. Smith, *Introduction to Evolutionary Computing*, Springer, 2003.
7. Xie, P., K.-S. Chen, and Z.-S. He, "Synthesis of sparse cylindrical arrays using simulated annealing algorithm," *Progress In Electromagnetics Research Letters*, Vol. 9, 147–156, 2009.
8. Kirkpatrik, S., C. Gelatt, and M. Vecchi, "Optimization by simulated annealing," *Science*, Vol. 220, 671–680, 1983.
9. Haupt, R. L., "Adaptive nulling with weight constraints," *Progress In Electromagnetics Research B*, Vol. 26, 23–38, 2010.
10. Hosseini, S. A. and Z. Atlasbaf, "Optimization of side lobe level and fixing quasi-nulls in both of the sum and difference patterns by using continuous ant colony optimization (ACO) method," *Progress In Electromagnetics Research*, Vol. 79, 321–337, 2008.
11. Kennedy, J. and R. Eberhart, "Particle swarm optimization," *Proc. IEEE Int. Conf. Neural Networks*, 1942–1948, 1995.
12. Pathak, N. N., B. Basu, and G. K. Mahanti, "Combination of inverse fast fourier transform and modified particle swarm optimization for synthesis of thinned mutually coupled linear array of parallel half-wave length dipole antennas," *Progress In Electromagnetics Research M*, Vol. 16, 105–115, 2011.
13. Kim, S.-Y. and N.-H. Myung, "An optimal antenna pattern synthesis for active phased array SAR based on particle swarm optimization and adaptive weighting factor," *Progress In Electromagnetics Research C*, Vol. 10, 129–142, 2009.
14. Mallahzadeh, A. R., H. Oraizi, and Z. Davoodi-Rad, "Application of the invasive weed optimization technique for antenna configurations," *Progress In Electromagnetics Research*, Vol. 79, 137–150, 2008.
15. Sheng, N., C. Liao, W. Lin, L. Chang, Q. Zhang, and H. Zhou, "A hybrid optimized algorithm based on EGO and Taguchi's method for solving expensive evaluation problems of antenna design," *Progress In Electromagnetics Research C*, Vol. 17, 181–192, 2010.
16. Dib, N. I., S. K. Goudos, and H. Muhsen, "Application of Taguchi's optimization method and self-adaptive differential evolution to the synthesis of linear antenna arrays," *Progress In Electromagnetics Research*, Vol. 102, 159–180, 2010.
17. Chowdhury, A., A. Ghosh, R. Giri, and S. Das, "Optimization of antenna configuration with a fitness-adaptive differential evolution algorithm," *Progress In Electromagnetics Research B*, Vol. 26,

291–319, 2010.

18. Pal, S., S. Das, and A. Basak, "Design of time-modulated linear arrays with a multi-objective optimization approach," *Progress In Electromagnetics Research B*, Vol. 23, 83–107, 2010.
19. Pal, S., B. Qu, S. Das, and P. N. Suganthan, "Linear antenna array synthesis with constrained multi-objective differential evolution," *Progress In Electromagnetics Research B*, Vol. 21, 87–111, 2010.
20. Lukasik, S. and S. Zak, "Firefly algorithm for continuous constrained optimization tasks," *Lecture Notes in Computer Science*, Springer Link, Vol. 5796, 2009.
21. Yang, X.-S., "Firefly algorithm, stochastic test functions and design optimization," *International Journal of Bio-Inspired Computation*, Vol. 2, No. 2, 2010.
22. Karaboga, D. and B. Basturk, "Artificial Bee Colony (ABC) optimization algorithm for solving constrained optimization problems," *LNCS: Advances in Soft Computing: Foundations of Fuzzy Logic and Soft Computing*, Vol. 4529, 789–798, Springer-Verlag, 2007.
23. Karaboga, D. and B. Basturk, "A powerful and efficient algorithm for numerical function optimization: Artificial Bee Colony (ABC) algorithm," *Journal of Global Optimization*, Vol. 39, No. 3, 459–471, Springer, 2007.
24. Baykasoglu, A., L. Ozbakir, and P. Tapkan, "Artificial bee colony algorithm and its application to generalized assignment problem," *Swarm Intelligence Focus on Ant and Particle Swarm Optimization*, 113–144, I-Tech. Education and Publishing, Vienna, Austria, 2007.
25. Elliott, R. S., *Antenna Theory and Design*, Wiley Interscience, New York, 2003.
26. Gibbons, J. D., *Nonparametric Statistical Inference*, 2nd edition, M. Dekker, 1985.
27. Roy, G. G., S. Das, P. Chakraborty, and P. N. Suganthan, "Design of non-uniform circular antenna arrays using a modified invasive weed optimization algorithm," *IEEE Transactions on Antennas and Propagation*, Vol. 59, No. 1, 110–118, Jan. 2011.
28. Panduro, M. A., C. A. Brizuela, L. I. Balderas, and D. A. Acosta, "A comparison of genetic algorithms, particle swarm optimization and the differential evolution method for the design of scanable circular antenna arrays," *Progress In Electromagnetics Research B*, Vol. 13, 171–186, 2009.