Nature-inspired Cuckoo Search Algorithm for Side Lobe Suppression in a Symmetric Linear Antenna Array

Khairul Najmy ABDUL RANI[1], *Mohd. Fareq ABD MALEK*[2], *Neoh SIEW-CHIN*[3]

[1]School of Computer and Communications Engineering, Universiti Malaysia Perlis, 01000 Kangar, Malaysia
[2]School of Electrical Systems Engineering, Universiti Malaysia Perlis, 01000 Kangar, Malaysia
[3]School of Microelectronic Engineering, Universiti Malaysia Perlis, 01000 Kangar, Malaysia

khairul_najmy@yahoo.com, mfareq@unimap.edu.my, scneoh@unimap.edu.my

Abstract. *In this paper, we proposed a newly modified cuckoo search (MCS) algorithm integrated with the Roulette wheel selection operator and the inertia weight controlling the search ability towards synthesizing symmetric linear array geometry with minimum side lobe level (SLL) and/or nulls control. The basic cuckoo search (CS) algorithm is primarily based on the natural obligate brood parasitic behavior of some cuckoo species in combination with the Lévy flight behavior of some birds and fruit flies. The CS metaheuristic approach is straightforward and capable of solving effectively general N-dimensional, linear and nonlinear optimization problems. The array geometry synthesis is first formulated as an optimization problem with the goal of SLL suppression and/or null prescribed placement in certain directions, and then solved by the newly MCS algorithm for the optimum element or isotropic radiator locations in the azimuth-plane or xy-plane. The study also focuses on the four internal parameters of MCS algorithm specifically on their implicit effects in the array synthesis. The optimal inter-element spacing solutions obtained by the MCS-optimizer are validated through comparisons with the standard CS-optimizer and the conventional array within the uniform and the Dolph-Chebyshev envelope patterns using MATLAB*^{TM}*. Finally, we also compared the fine-tuned MCS algorithm with two popular evolutionary algorithm (EA) techniques include particle swarm optimization (PSO) and genetic algorithms (GA).*

Keywords

Modified cuckoo search, side lobe suppression, null control, linear array, isotropic radiator, Dolph-Chebyshev.

1. Introduction

In modern wireless applications, the antenna pattern is designed so as to produce or steer a strong beam towards the preferred signal according to signal direction of arrival and/or concurrently to cancel interfering signals (placing prescribed nulls). Such antenna system is called smart antenna array [1], which uses arrays of antenna elements efficiently and can integrate multiple antenna elements to process a signal [2] coming from various directions. In other definition, the term smart antenna generally refers to any antenna array, terminated in a sophisticated signal processor, which can adjust or adapt its own beam pattern in order to emphasize signals of interest and to minimize interfering signals [3]. In this case, array geometry synthesis plays an important role to determine the physical layout of the array that produces the radiation pattern closest to the desired pattern. The shape of the desired pattern can vary widely depending on the application [4].

The array pattern should possess high power gain, lower side lobe levels, controllable beam width [5] and good azimuthal symmetry. The desired radiation pattern of the antenna array can be realized by determining the physical layout of the antenna array and by choosing suitable complex excitation of the amplitude and phase of the currents that are applied on the array elements [6]. Many synthesis techniques are done through suppressing the side lobe level (SLL) while simultaneously maintaining the gain of the main beam. Other techniques focus on the null control to reduce the effects of undesired interference and jamming. In the linear array geometry context, this can be made through designing and/or optimizing the inter-element spacings with respect to the $\lambda/2$ distance either while preserving a uniform excitation amplitude or phase over the array aperture or employing nonuniform excitation and phased arrays [7].

2. System Description

Due to the high versatility, flexibility and capability to optimize complex multidimensional problem, modern evolutionary algorithm (EA) techniques such as genetic algorithms (GA) [8], [9], simulated annealing (SA) [10], and particle swarm optimization (PSO) [2], [6], [7], [11] have been applied for antenna array beam design, which directly manipulates the configuration of individual or isotropic radiating elements arranged in space and performs required beam forming techniques to produce a uniform directional radiation pattern. In addition, the trade-off between the SLL and the half-power beam width (HPBW) stimulate was answered through obtaining the narrowest possible beam width for a given SLL or the smallest SLL for a given beam width [12]. This was also possible by using the orthogonal functions of Chebyshev [13] in order to design an optimum radiation pattern. Fortunately, the use of modern EA has solved the deficiency and burdensome of matching the array factor expression with an appropriate Chebyshev function appears for large number of elements [14]. In this case, PSO [11], [15] and tabu search (TS) [16] were used to reduce the side lobes of linear arrays in Chebyshev sense through evaluating the gradient of some cost function.

This paper introduces the newly evolved cuckoo search (CS) metaheuristic algorithm developed recently by Xin-She Yang and Suash Deb in 2009 [17]. It was proven in [18] that the CS was more generic and robust than the PSO and GA in optimizing multimodal objective functions. Through simulations running on various standard test functions, CS was found to be more efficient in finding the global optima with higher success rates. This is partly due to the fact that there are fewer parameters to be fine-tuned in CS than in PSO and GA [18]. Furthermore, CS is still new and has never been used for any array geometry synthesis before. To the best of our knowledge, so far, CS was successfully used for mechanical engineering problems, which were spring design optimization and welded beam design [19]. Hence, CS has a great potential also to be as an effective alternative besides other evolutionary algorithms in handling electromagnetic or array optimization problems. In this paper, the newly modified CS (MCS) algorithm is proposed to optimize the distance between the n-th elements of the symmetric linear array to generate a radiation pattern with minimum side lobes and prescribed null placement control in the xy-plane or azimuth plane as shown in Fig. 1.

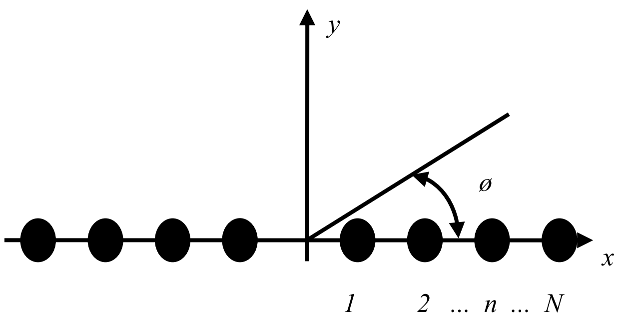

Fig. 1. Geometry of the *2N*-element symmetric linear array.

Besides, the MCS algorithm will be also used to synthesis an optimal linear array in the Chebyshev sense with equiripple side lobes radiation patterns. It was assumed throughout the experiment that the *2N*-isotropic radiators were placed symmetrically along the x-axis. The array factor in the azimuth plane can be stated as [7]

$$AF(\phi) = 2\sum_{n=1}^{N} I_n \cos[kx_n \cos(\phi + \varphi_n)] \tag{1}$$

where $k = 2\pi/\lambda$ is the wave number, and I_n, φ_n, and x_n are the excitation amplitude, phase, and location of the n-th element, respectively. For a uniform excitation of amplitude and phase, I_n is assumed to be 1.0 whereas φ_n is set to 0 for all elements. Hence, the array factor can be simplified to be as [7]

$$AF(\phi) = 2\sum_{n=1}^{N} I_n \cos[kx_n \cos(\phi)]\,. \tag{2}$$

Through the above simplification, the newly developed MCS algorithm is specifically used to find the locations, x_n of the symmetric linear array elements with minimum side lobes and/or nulls at specific direction.

3. Modified Cuckoo Search

The power and beauty of modern metaheuristic comes from the capability of emulating the best feature in nature, specifically biological systems evolved from natural selection over millions of years via two important characteristics, which are selection of the fittest in biological systems, and adaptation to the environment [18]. Blum and Roli in 2003 [20] classified two crucial attributes in the modern heuristics, which were intensification, and diversification. Precisely, intensification aims to search around the current best candidates (possible solutions) and through it, selects the best solutions, while diversification ensures the algorithm to explore the local or global search space efficiently.

Cuckoo search (CS) was inspired by the obligate brood parasitism of some cuckoo species by laying their eggs in the nests of other host birds (of other species). Some host birds can engage direct conflict with the intruding cuckoos. For example, if a host bird discovers the eggs are not their own, it will either throw these alien eggs away or simply abandon its nest and build a new nest elsewhere. Some cuckoo species such as the new world brood-parasitic *Tapera* have evolved in such a way that female parasitic cuckoos are often very specialized in the mimicry in colors and pattern of the eggs of a few chosen host species, thus having a greater chance for the cuckoo's eggs hatch successfully. The simplest approach of using new metaheuristic CS algorithm can be done through three idealized assumptions, which are: 1) Each cuckoo lays one egg at a time, and dump its egg in randomly chosen nest; 2) The best nests with high quality of eggs will carry over to the next generations; and 3) The number of available host nests is fixed where the egg laid by a cuckoo is discovered by the host bird with a measured fraction probability, $P_a \in [0, 1]$. In this case, the host bird may throw the egg away or may abandon the nest, hence build a completely new nest. The third assumptions can be approximated as the fraction P_a of the n nests is replaced by new nests (new random solutions). Many studies have shown that the flight behavior of many animals and insects has demonstrated the typical characteristics of Lévy flights [20-23]. Lévy flight is defined as a random walk with the step-lengths based on a heavy-tailed probability distribution [24]. Consequently, such behavior has been emulated to optimization and global optimal search with a promising capability [23], [25].

In this study, the standard CS algorithm has been modified to enhance its performance. In this case, when generating new solutions $x^{(t+1)}$ for a cuckoo i, a Lévy flight [18], [19] integrating with the inertia weight, w, which controls the search ability is performed

$$x_i^{t+1} = w.x_i^t + \alpha \oplus Lévy\ (\lambda) \tag{3}$$

where $\alpha > 0$ is the step size related to the scales of the problem of interest while the product $\oplus$ means entry-wise multiplications. Technically, the larger w has greater global search ability whereas the smaller w has greater local search ability. Based on (3), w was linearly decreased from a relatively large value to a small value through the course so that the MCS had a better performance compared with fixed w settings.

$$w = w_{\max} - \left[\left(w_{\max} - w_{\min}\right) \times iter\right] / \, maxIter \tag{4}$$

where $w_{\max} = initialWeight$, $w_{\min} = finalWeight$.

Conceptually, Lévy distribution for large steps applied a power law, thus has an infinite variance [24], depicted as

$$Lévy{\sim}u = t^{-\lambda}, (1 < \lambda \le 3)\,. \tag{5}$$

Lévy flight is classified as a Markov process where after a large number of steps, the distance from the origin of random walk tends to a stable distribution. Statistically, this is done through the stochastic process with both stationary and independent increments. In this paper, three types of α-stable distribution will be explained and simulated for comparison purposes. The first one is Mantegna's algorithm with $\alpha \in [0.3, 1.99]$ as the input parameters [26]. In Mantegna's algorithm, the step v can be calculated as

$$v = \frac{x}{|y|^{1/\alpha}} \tag{6}$$

where x and y are normally distributed stochastic variables with standard deviations, respectively

$$\sigma_x = \left[\frac{\Gamma(1+\alpha)\sin(\pi\alpha/2)}{\Gamma((1+\alpha)/2\alpha 2^{(\alpha-1)/2}}\right]^{1/\alpha}, \tag{7}$$

$$\sigma_y = 1\,. \tag{8}$$

If (6)-(8) are applied, the resulting distribution will have the same behavior of a Lévy distribution for large values of random variable ($|v| \ge 0$). To calculate the step size of Lévy flights, v will be then multiplied with n factor where $n \in \Re$. Normally, n is set to 0.01 from the fact that $L/100$ is the step size of walks or flights where L is the length scale of cuckoo's motions in searching for new nest. Proper step size factors must be set to ensure the Lévy flights do not be too aggressive, which makes new solutions jump outside of the design domain.

Secondly, we can use McCulloch's algorithm to generate α-stable generation of Lévy flights or noise [27], [28]. The algorithm returns matrix of random numbers with characteristic exponent α, scale c, and location parameter τ. In this case, α must be greater than 0.1 due to the non-negligible probability of overflow and no skewness ($\beta = 0$) is assumed. There are three cases to calculate the simplified step v of α-stable distribution:

1) Cauchy case ($\alpha = 1$)

$$v = c\tan(\varphi) + \tau \tag{9}$$

2) Gaussian case ($\alpha = 2$)

$$v = c2\sqrt{w}\sin(\varphi) + \tau \tag{10}$$

3) Other cases ($\alpha \ne 1$ or $\alpha \ne 2$)

$$v = c\left(\frac{\cos((1-\alpha)\varphi)}{w}\right)^{\frac{1}{\alpha}-1}\left(\frac{\sin(\alpha\varphi)}{\cos(\varphi)}\right)^{\frac{1}{\alpha}} + \tau \tag{11}$$

where $c > 0$, w are negative logarithm of random numbers, φ are random angles in radians, and for simplicity, $\tau = 0$. Thirdly, the simplest way to generate a stable Lévy distribution is by creating standard random walk which has a step v constantly equal to one.

Fig. 2 is the flowchart diagram, which shows the main steps of the conventional CS algorithm. Here the concept of fitness, F is used to guide the Lévy flights during the search for the optimum nest (solution) in the N-dimensional space. The N-dimensional solution refers to the N-symmetric element positions along the xy-plane.

In antenna design, the fitness optimization is done to examine or analyze scientifically on many aspects depending on the application wise such as directivity, gain, side lobe level (SLL), size, and weight. In this paper, the fitness optimization is primarily done to design the geometry of a linear antenna steering at the desired direction with minimum average SLL and/or nulls control using the following objective function:

$$Fitness = \sum_i \frac{1}{\Delta\phi_i}\int_{\phi_{li}}^{\phi_{ui}} \left|AF(\phi)\right|^2 d\phi + \sum_k \left|AF(\phi_k)\right|^2 \tag{12}$$

where $[\phi_{li}, \phi_{ui}]$'s are the spatial regions in which the side lobe is suppressed, $\Delta\phi_i = \phi_{ui} - \phi_{li}$, and ϕ_k's are the directions of the nulls. Precisely, the first-term on the right-hand side of the fitness function focuses on side lobe suppression whereas the second-term on the right-hand side is used for null control. In this study, the nest's location vector resulted the minimum value of the fitness function is chosen as the best nest's location (the best normalized elements' locations).

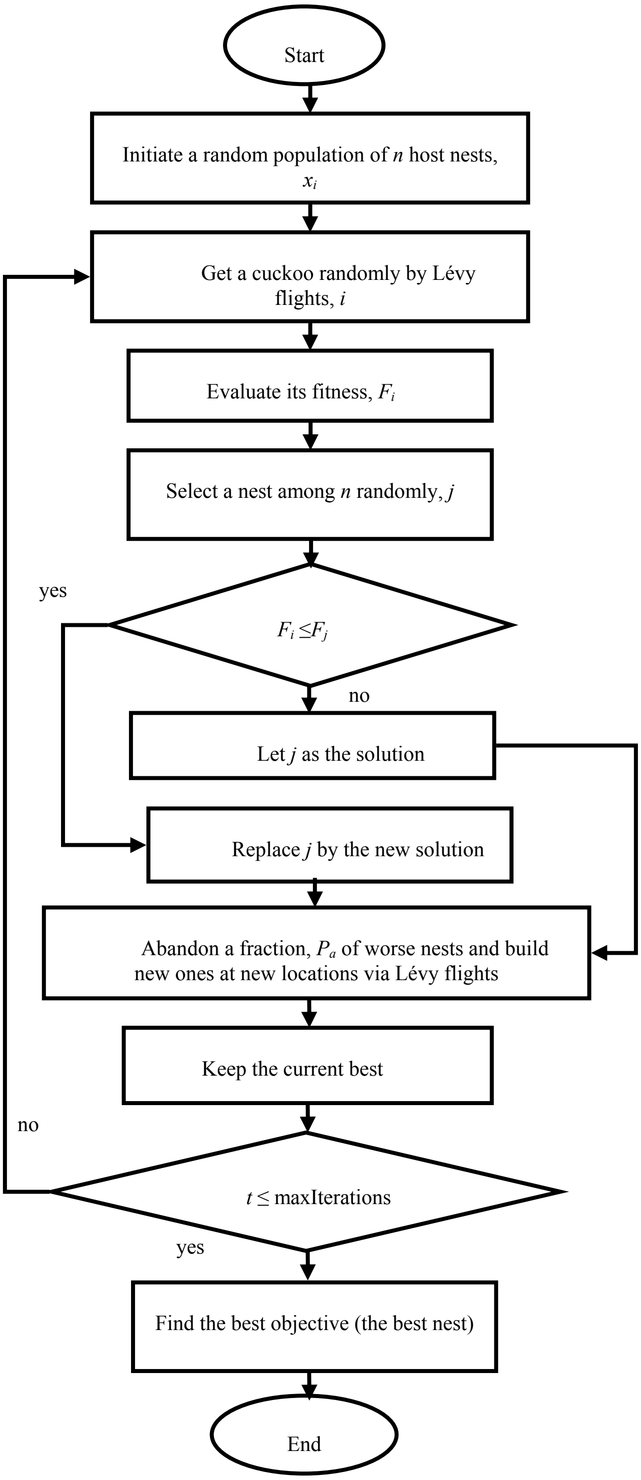

Fig. 2. Flowchart of conventional cuckoo search algorithm.

Below is the proposed pseudo-code of MCS algorithm used to search the optimal excitation location of the n-th elements of the symmetric linear array as shown in Fig. 1.

begin
Let *iter* denote the iteration number of MCS.
iter ←1;
Initialize population of host nests with size n at *iter*=1;
for each iteration
Operate the Roulette wheel selection to obtain the "fittest" host nests with size n;
Generate a new solution (host nest) but keep the current best (say, i) randomly by Lévy flights incorporating with inertia weight, w, which controls the search ability according to (3);
Evaluate new solution fitness, F_i according to (12);
Get a selected host nest among n (say, j) and calculate its fitness, F_j according to (12);
if $(F_i < F_j)$
Replace j by the new solution, i;
end
A fraction probability, P_a of worse nests is abandoned and a new nest (solution) is built;
Keep the best nests with quality solutions;
Rank the solutions and find the current best nest;
end
Post-process results and visualization;
end

In this case, the best nest is assumed to be as the most optimum decision variable (excitation location, x) of symmetric linear antenna array elements.

4. Simulation Results

In the preliminary study, four internal parameters in both modified CS (MCS) and standard CS optimizers will be analyzed specifically on their implicit effects of the normalized pattern. The parameters were α, distribution type, number of host nest (size of population), and fraction probability, P_a (discovery rate). Firstly, we analyzed the characteristic of α. In this case, the CS-optimizer with nest = 30, $P_a = 0.25$, step size factor = $L/100$, along with both $\alpha = 1.0$ (Cauchy) and $\alpha = 2.0$ (Gaussian) were examined on the $2N = 10$ array, respectively.

Fig. 3 shows the four normalized patterns after running through 2000 iterations where the MCS ($\alpha = 2.0$) outperformed the MCS ($\alpha = 1.0$) and the standard CS with both Gaussian and Cauchy distributions. On average, the MCS ($\alpha = 2.0$) has the side lobes level (SLL) 2.0-3.0 dB lower than other three counterparts whereas the MCS ($\alpha = 1.0$) has a similar performance with the standard CS ($\alpha = 2.0$).

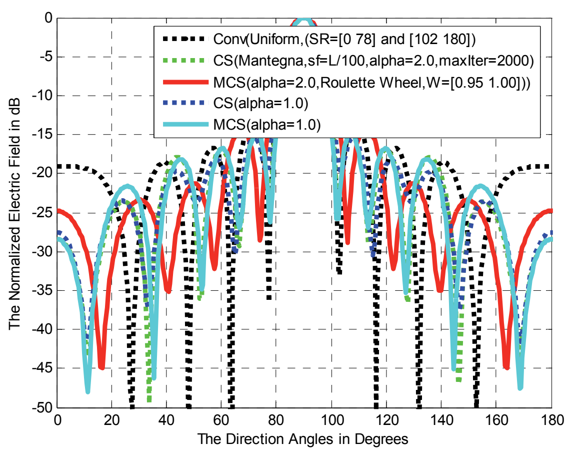

Fig. 3. Normalized pattern vs. α

Secondly, we also examined the characteristic of three different stable distribution types (as mentioned earlier in section 3) particularly on their effects in the normalized array pattern for the broadside case (main beam at 90°). Fig. 4 clearly indicates the MCS with the Mantegna's algorithm has the best side lobes suppression with the peak of 4.0-6.0 dB lower than other two MCS-optimizers and three standard CS-optimizers particularly within the direction angles domain of [20° 75°] and [105° 160°]. We also found that both MCS with the McCulloch's and standard random walk (Lévy flights step, $v = 1.0$) algorithms had a similar performance with other three CS counterparts within the [0° 78°] and [102° 180°] suppression region.

Thirdly, a simulation was also done to find out the effect of the number of population (host nest) on the array geometry synthesis. In this experiment, both the heuristic MCS and CS-optimizers with $P_a = 0.25$, Mantegna's stable distribution, $\alpha = 2.0$ (Gaussian), and step size factor = $L/100$ were simulated on $2N = 10$ linear array using three different population sizes (host nest = 10, 20, and 30), respectively. As shown in Fig. 5, the MCS optimizers (nest = 20 and 30) had approximately SLL of 5.0 dB lower than the MCS (nest = 10) optimizer and the three CS (nest = 10, 20, and 30) optimizers, within the predefined suppression region [0° 78°] and [102° 180°].

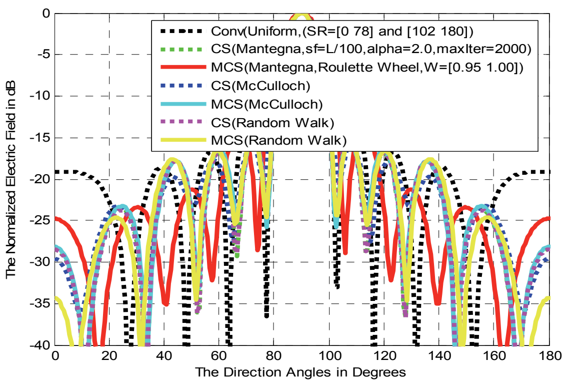

Fig. 4. Normalized pattern vs. distribution type.

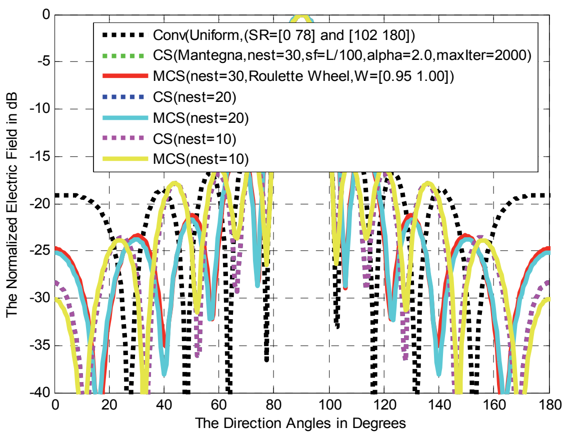

Fig. 5. Normalized pattern vs. nest.

Fig. 6 shows all the MCS optimizers were identical in terms of fitness convergence level ($fmin = 0.0125$) regardless the number of nest. It happened for CS optimizers too ($fmin = 0.0119$). Moreover, we found that both MCS and CS optimizers had a faster convergence rate as the number of population (nest) bigger. This was due to the higher capability or greater probability to find global minimum values since larger number of population (nest) occupied in the fixed search space. Nevertheless, the fluctuations of the optimal excitation locations, x tends to be unwavering quicker as the number of nest smaller as can be seen on the three MCS-optimizers.

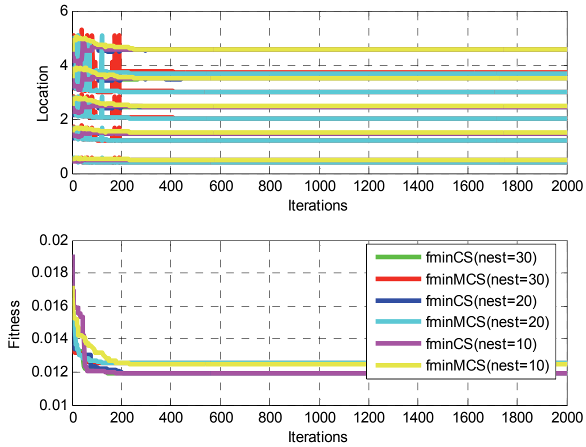

Fig. 6. Location and fitness curve for nest = 10, 20, and 30.

Fourthly, an investigation was done on $2N = 20$ linear array elements using three different fraction probability, P_a values. As shown in Fig. 7, MCS with $P_a = 0.05$ had the best performance (SLL 1.0-2.0 dB relatively lower within the [30° 80°] and [100° 150°] suppression domain) followed by CS-optimizer with $P_a = 0.05$. The competitors with $P_a = 0.95$ were the worst ones. Besides that, we also found the converged fitness for both MCS and CS-optimizers with $P_a = 0.05$ were the lowest ones ($fmin$= 0.0145 after about 250 iterations for MCS and $fmin$= 0.0134 after about 400 iterations for CS), respectively.

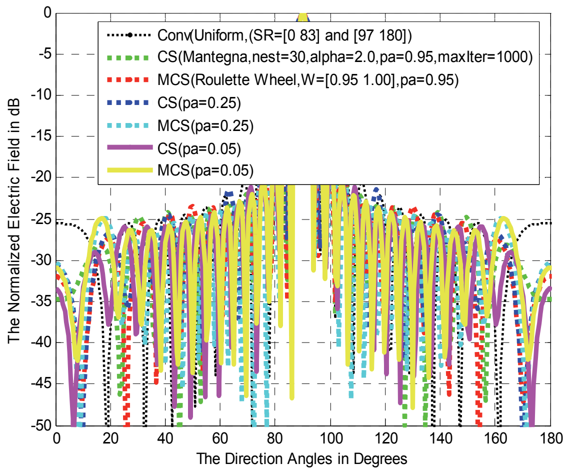

Fig. 7. Normalized pattern vs. fraction probability P_a.

Moreover, the converged was higher as the P_a values bigger as shown in Fig. 8. This agrees with the CS algorithm assumption: Whenever P_a or discovery rate greater, the possibility of egg laid by a cuckoo to be discovered by the host bird of other species becomes higher. As a result, the cuckoo's egg (candidate solution) could be abandoned or thrown away leading to a new host nest searching or replacement done by cuckoo.

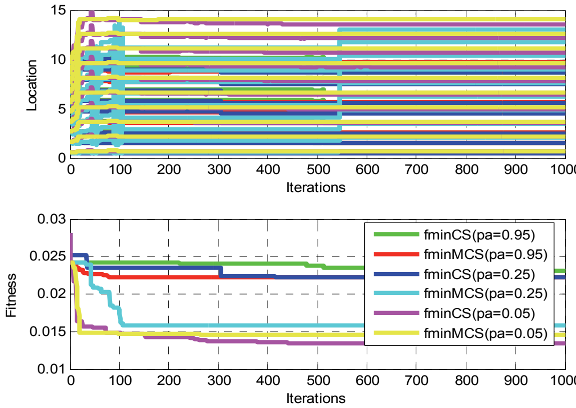

Fig. 8. Location and fitness curve for P_a=0.05, 0.25, and 0.95.

Fifthly, there was also a more rigid experimental simulation done on the non-broadside case (main beam steered to 135°) on $2N = 20$ array with three prescribed nulls at 20°, 50° and 155°. Both MCS and CS-optimizers with three distribution types, nest = 20, $P_a = 0.25$, $\alpha = 2.0$ (Gaussian) and the step size factor = $L/100$ were run for 1000 iterations. Fig. 9 generally shows the MCS-based arrays with the Mantegna' algorithm outperformed other rivals with the SLL of 2.0-8.0 dB relatively lower than the conventional array within the [20° 130°] suppression region. It was then followed by MCS with the standard random walk algorithm. The CS with the random walk algorithm was the worst one. The CS with the McCulloch's algorithm had the best nulls mitigation at 20° with SLL of -69.9 dB, the MCS with the McCulloch's algorithm at 50° with SLL of -59.8 dB, and MCS with random walk at 155° with SLL of -49.5 dB, respectively. This also indicated that regardless stable distribution type, the MCS and CS optimizers are capable to diminish the side lobes beam while maintaining the gain of the steered main beam and controlling the prescribed interferers, simultaneously.

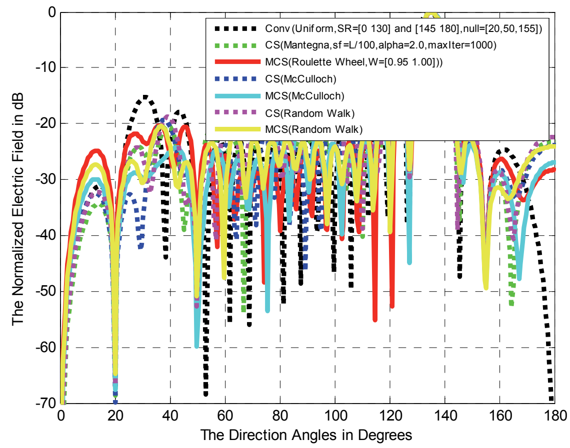

Fig. 9. Normalized pattern for $2N$=20 array with mean beam at 135° and three prescribed nulls at 20°, 50°and 155°.

Element	**1**	**2**	**3**	**4**	**5**
X_n [λ/2]	**±0.500**	**±1.500**	**±2.500**	**±3.500**	**±4.500**
MCS Mantegna	**±0.547**	**±1.642**	**±2.737**	**±3.832**	**±4.927**
CS Mantegna	**±0.538**	**±1.615**	**±2.693**	**±3.772**	**±4.847**
MCS McCulloch	**±0.542**	**±1.629**	**±2.714**	**±3.800**	**±4.883**
CS McCulloch	**±0.541**	**±1.616**	**±2.691**	**±3.774**	**±4.853**
MCS RandWalk	**±0.544**	**±1.633**	**±2.726**	**±3.818**	**±4.908**
CS RandWalk	**±0.537**	**±1.612**	**±2.686**	**±3.768**	**±4.834**
Element	**6**	**7**	**8**	**9**	**10**
X_n [λ/2]	**±5.500**	**±6.500**	**±7.500**	**±8.500**	**±9.500**
MCS Mantegna	**±6.022**	**±7.117**	**±8.201**	**±9.294**	**±10.39**
CS Mantegna	**±5.922**	**±6.995**	**±8.065**	**±9.246**	**±10.32**
MCS McCulloch	**±5.969**	**±7.046**	**±8.140**	**±9.224**	**±10.30**
CS McCulloch	**±5.918**	**±6.989**	**±8.064**	**±9.132**	**±10.18**
MCS RandWalk	**±5.998**	**±7.085**	**±8.160**	**±9.272**	**±10.36**
CS RandWalk	**±5.909**	**±6.983**	**±8.050**	**±9.226**	**±10.30**

Tab. 1. Optimal locations for $2N$=20 array with mean beam at 135° and three prescribed nulls at 20°, 50° and 155°.

Fig. 10 shows the convergence of the location and the fitness obtained for all the three distribution types. MCS with the Mantegna's algorithm clearly had the lowest converged fitness (*fmin* = 0.2713), followed by MCS with the standard random walk (*fmin* = 0.2733), and CS with the Mantegna's algorithm (*fmin* = 0.2740), accordingly. The optimal locations with respect to λ/2 for both MCS and CS with three different α-stable distributions versus the conventional ones can be seen in Tab. 1.

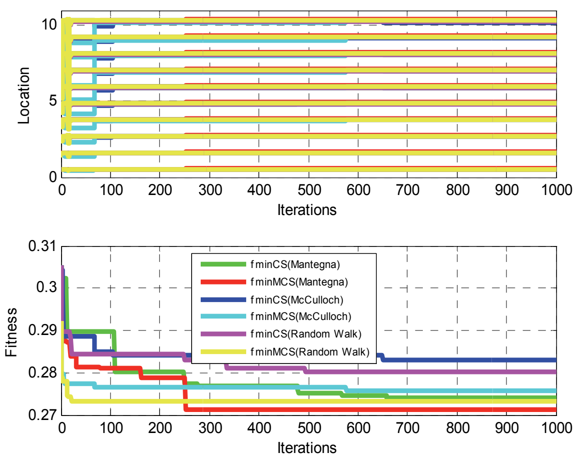

Fig. 10. Location and fitness curve for $2N = 20$ elements with mean beam at 135° and two prescribed nulls at 20°, 50°and 155°.

Sixthly, there was also an experimental simulation done on using the MCS to optimize the 20-symmetric elements to generate the Dolph-Chebyshev radiation pattern.

In this experiment, the $2N = 20$ fixed excitation Dolph-Chebyshev amplitude, I_n equals to [1.0000 0.8771 1.2009 1.5497 1.9052 2.2465 2.5522 2.8022 2.9793 3.0712] as shown in Fig. 11 was used whereas the excitation phase, φ_n was set to 0°. Based on Fig. 12, MCS with the Mantegna's algorithm had the best equiripple side lobes suppression with the gain relatively 32.1 dB below than the main beam followed by both MCS with the McCulloch's and the standard random walk algorithms with 31.9 dB below, respectively. The CS with the McCulloch's algorithm was the worst one with 31.6 dB below the main beam. Please refer Tab. 2 for optimal locations with respect to $\lambda/2$.

Element	**1**	**2**	**3**	**4**	**5**
X_n [$\lambda/2$]	**±0.500**	**±1.500**	**±2.500**	**±3.500**	**±4.500**
MCS Mantegna	**±0.743**	**±2.230**	**±3.716**	**±5.203**	**±6.689**
CS Mantegna	**±0.526**	**±1.578**	**±2.629**	**±3.681**	**±4.733**
MCS McCulloch	**±0.538**	**±1.615**	**±2.691**	**±3.767**	**±4.844**
CS McCulloch	**±0.500**	**±1.500**	**±2.500**	**±3.499**	**±4.499**
MCS RandWalk	**±0.584**	**±1.751**	**±2.918**	**±4.085**	**±5.252**
CS RandWalk	**±0.526**	**±1.578**	**±2.630**	**±3.682**	**±4.734**
Element	**6**	**7**	**8**	**9**	**10**
X_n [$\lambda/2$]	**±5.500**	**±6.500**	**±7.500**	**±8.500**	**±9.500**
MCS Mantegna	**±8.176**	**±9.662**	**±11.15**	**±12.64**	**±14.12**
CS Mantegna	**±5.785**	**±6.836**	**±7.888**	**±8.940**	**±9.992**
MCS McCulloch	**±5.920**	**±6.996**	**±8.073**	**±9.149**	**±10.23**
CS McCulloch	**±5.498**	**±6.498**	**±7.498**	**±8.498**	**±9.500**
MCS RandWalk	**±6.419**	**±7.586**	**±8.753**	**±9.920**	**±11.09**
CS RandWalk	**±5.787**	**±6.840**	**±7.893**	**±8.942**	**±9.995**

Tab. 2. Optimal locations for $2N = 20$ Dolph-Chebyshev array.

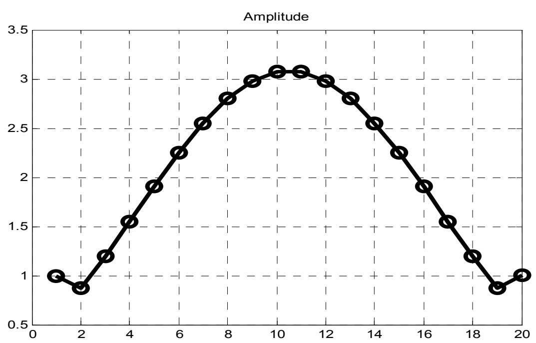

Fig. 11. The Dolph-Chebyshev excitation amplitude for $2N$=20

Fig. 13 shows the MCS with Mantegna's algorithm had the biggest optimal location, x fluctuations before stabilizing after about 800 iterations. Furthermore, we could see all the MCS-based optimizers (e.g. $fmin = 0.0111$ for Mantegna, $fmin = 0.0114$ for McCulloch and $fmin = 0.0121$ for standard random walk) had the relatively lower converged fitness compared to CS rivals. This proves that the MCS algorithms are capable to enhance the existing original CS optimization process through minimizing the global fitness at the lowest possible values.

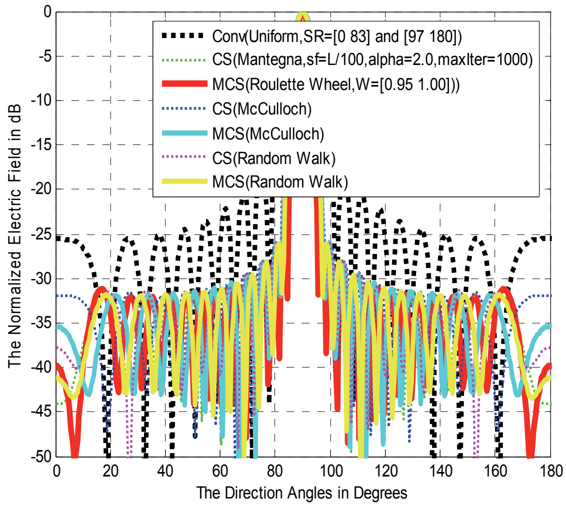

Fig. 12. Normalized Dolph-Chebyshev pattern for $2N$=20 array

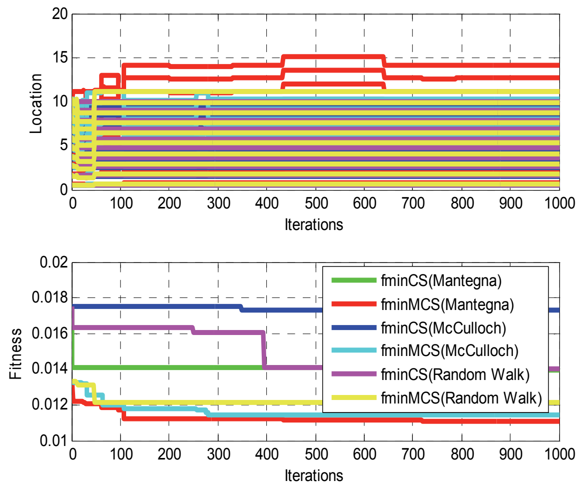

Fig. 13. Location and fitness for $2N$=20 Dolph-Chebyshev array.

Seventhly, we performed an experiment on $2N = 30$ symmetric Dolph-Chebyshev array in which the modified cuckoo search (MCS) algorithm based array was relatively compared with normal CS, genetic algorithms (GA), and particle swarm optimization (PSO) based arrays. Both MCS and CS optimizers were run for 1000 iterations with parameters, e.g. nest = 30, $P_a = 0.25$, step size factor = L/100, and $\alpha = 2.0$ (Gaussian). The fixed excitation Dolph-Chebyshev envelope as illustrated in Fig. 14 was used whereas for a simplification, the excitation phase, φ_n was set to 0° for all 30 elements. Precisely, the respective amplitude, I_n applied was [0.4235 0.2477 0.3127 0.3827 0.4564 0.5322 0.6083 0.6826 0.7532 0.8182 0.8756 0.9238 0.9613 0.9870 1.0000]. The conventional and the optimal locations generated by four nature-inspired optimizers with respect to $\lambda/2$ of $2N = 30$ array are shown in Tab. 3.

Element	**1**	**2**	**3**	**4**	**5**
$X_n[\lambda/2]$	**±0.5000**	**±1.5000**	**±2.5000**	**±3.5000**	**±4.5000**
MCS Mantegna	**±0.7269**	**±2.1807**	**±3.6344**	**±5.0882**	**±6.5420**
CS Mantegna	**±0.5000**	**±1.5002**	**±2.5004**	**±3.5005**	**±4.5007**
PSO	**±0.4996**	**±1.4996**	**±2.4996**	**±3.4996**	**±4.4996**
GA	**±0.5417**	**±1.6251**	**±2.7085**	**±3.7918**	**±4.8752**
Element	**6**	**7**	**8**	**9**	**10**
$X_n[\lambda/2]$	**±5.5000**	**±6.5000**	**±7.5000**	**±8.5000**	**±9.5000**
MCS Mantegna	**±7.9958**	**±9.4495**	**±10.9033**	**±12.3571**	**±13.8109**
CS Mantegna	**±5.5009**	**±6.5010**	**±7.5012**	**±8.5013**	**±9.5015**
PSO	**±5.4996**	**±6.4996**	**±7.4996**	**±8.4996**	**±9.4996**
GA	**±5.9586**	**±7.0420**	**±8.1254**	**±9.2087**	**±10.2921**
Element	**11**	**12**	**13**	**14**	**15**
$X_n[\lambda/2]$	**±10.5000**	**±11.5000**	**±12.5000**	**±13.5000**	**±14.5000**
MCS Mantegna	**±15.2646**	**±16.7184**	**±18.1722**	**±19.6259**	**±21.0797**
CS Mantegna	**±10.5013**	**±11.5018**	**±12.5020**	**±13.5021**	**±14.5018**
PSO	**±10.4996**	**±11.4996**	**±12.4996**	**±13.4996**	**±14.4996**
GA	**±11.3755**	**±12.4589**	**±13.5423**	**±14.6256**	**±15.7090**

Tab. 3. Optimal locations for $2N$= 30 Dolph-Chebyshev array.

The PSO optimizer was constructed with the population (particle) equals to 30, max/min velocity of ±0.1 and both the individuality and sociality accelerators set to 1.0, respectively. The GA optimizer with the Roulette wheel selection operators also had a population (chromosome) of 30, crossover probability of 0.9, and mutation probability of 0.1. Based on Fig. 15, all the MCS with the Mantegna's algorithm had the identical highest peak of equiripple side lobes gain relatively about 33.35 dB below the main beam and relatively 20.10 dB below the highest SLL peak of the uniform (conventional) pattern, respectively due to the average fitness convergence attainment with the lowest value of 0.0023 after about 400 iterations as shown in Fig. 16. The highest SLL peak of the uniform pattern produced by the conventional array was comparatively 13.26 dB below the main beam. On the other hand, the standard CS with the Mantegna's algorithm had the equiripple SLL of relatively about 33.11 dB below the main beam or 19.85 dB below the uniform pattern highest SLL peak. The GA counterparts had the equiripple SLL of 33.06 dB, which was 19.80 dB below the uniform highest SLL peak. Moreover, we found the PSO based array had the equiripple SLL of approximately 32.97 dB below the main beam, which was 19.71 dB below the uniform highest SLL peak.

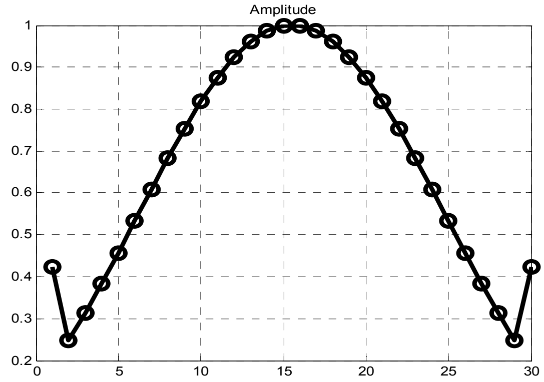

Fig. 14. The Dolph-Chebyshev excitation amplitude for $2N$=30.

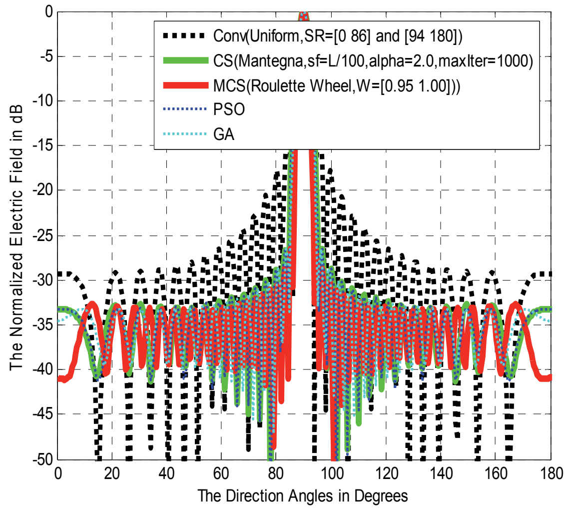

Fig. 15. Normalized pattern for $2N = 30$ Dolph-Chebyshev array.

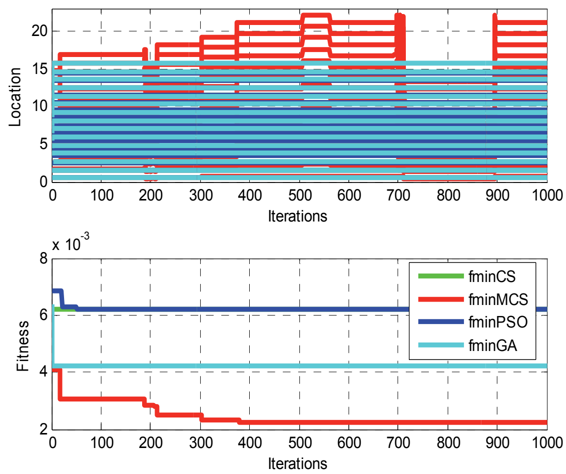

Fig. 16. Location and fitness for $2N = 30$ Dolph-Chebyshev array.

5. Conclusions

It is important to identify and understand the imperative characteristics of the entire internal modified cuckoo search (MCS) and standard CS parameters and fine-tune them before running with a sufficient maximum number of iterations to achieve the utmost convergence of optimal locations (solutions) and fitness. This can be made through the analysis of the internal parameters values and their effects on the array performances. As a matter of fact, various MATLAB™ simulations showed that the MCS metaheuristic algorithm with the appropriate internal parameter settings was proven can optimize the locations of linear array symmetric elements or isotropic radiators in the *xy*-plane to exhibit radiation patterns with clear suppressed side lobes and/or nulls mitigation in certain direction of arrivals while maintaining the gain of the main beam. This includes either for broadside case or non-broadside case (where the main beam steered to directions other than 90°) as required. In all cases, the MCS algorithm outperformed the normal CS counterparts due to its unique features, which are the Roulette wheel selection operator to obtain the "fittest" host nests and the dynamic inertia weight, *w* coefficient which controls the global search ability. The MCS algorithm was also proven slightly better than other evolutionary algorithm (EA) techniques, e.g. PSO and GA in suppressing side lobes at the minimum possible value below the main beam. Perhaps, more extensive study on symmetric linear array with *N*-elements will be made in the future to use the multiobjective MCS algorithm to optimize more than one decision variables simultaneously includes the excitation amplitudes and phases along with locations under the uniform and Dolph-Chebyshev environment.

References

[1] WAGIH, M. M. M., ELKAMCHOUCHI, H. M. Application of particle swarm optimization algorithm in smart antenna array systems. *Progress in Electromagnetics Research,* 2009, p. 461 to 476.

[2] HO, M.-H., LIAO, S.-H., CHIU, C.-C. A novel smart UWB antenna array design by PSO. *Progress in Electromagnetics Research C,* Aug 2010, vol. 15, p. 103-115.

[3] GROSS, F. *Smart Antennas for Wireless Communications with MATLAB*. New York: McGraw-Hill, 2005.

[4] PAL, S., BASAK, A., DAS, S., ABRAHAM, A. Linear antenna array synthesis with invasive weed optimization algorithm. In *International Conference of Soft Computing and Pattern Recognition*, 2009, p. 161-166.

[5] ZAHARIS, Z., KAMPITAKI, D., PAPASTERGIOU, A., HATZIGAIDAS, A., LAZARIDIS, P., SPASOS, M. Optimal design of a linear antenna array using particle swarm optimization. In *5th WSEAS Int. Conf. on Data Networks, Communications and Computers*. Bucharest (Romania), 2006.

[6] NIK ABD MALIK, N. N., ESA, M., SYED YUSOF, S. K., MARIMUTHU, J. Suppression of antenna's radiation side lobes using particle swarm optimization. In *Progress in Electromagnetics Research Symposium*. Moscow (Russia), 2009.

[7] KHODIER, M. M., CHRISTODOULOU, C. G. Linear array geometry synthesis with minimum side lobe level and null control using particle swarm optimization. *IEEE Trans. Antennas Propag.,* Aug. 2005, vol. 53, p. 2674-2679.

[8] GAVIRIA, N., MELDE, K. L., MCNEILL, K. M., RODRIGUEZ, J. J. Development of array distributions for smart antennas with low side lobes, interference-nulling, and effective radiated voltage constraints. In *IEEE Antennas and Propagation Society International Symposium.* Albuquerque (NM, USA), 2006, p. 3323-3326.

[9] CENGIZ, Y., TOKAT, H. Linear antenna array design with use of genetic, memetic and tabu search optimization algorithms. *Progress in Electromagnetics Research C,* 2008, vol. 1, p. 63-72.

[10] LEE, K. C. Frequency-domain analyses of nonlinearly loaded antenna arrays using simulated annealing algorithms. *Progress in Electromagnetics Research,* 2005, vol. 53, p. 271-281.

[11] KHODIER, M. M., AL-AQEEL, M. Linear and circular array optimization: A study using particle swarm intelligence. *Progress in Electromagnetics Research B,* 2009, vol. 15, p. 347-373.

[12] DOLPH, C. A current distribution for broadside arrays which optimizes the relationship between beam width and side lobe level. *Institute of Radio Engineers*, 1946, p. 335-348.

[13] GREENBERG, M. *Advanced Engineering Mathematics*. New Jersey: Prentice-Hall, 1998.

[14] BALANIS, C. A. *Antenna Theory Analysis and Design*. 3rd ed. San Francisco: John Wiley, 2005.

[15] BATAINEH, M. H. On Chebyshev array design using particle swarm optimization. *Journal of Electromagnetic Analysis and Applications,* 2011, vol. 3, p. 213-219.

[16] MERAD, L., BENDIMERAD, F., SIDI, M. Design of linear antenna arrays for side lobe reduction using the tabu search method. *The International Arab Journal of Information Technology,* 2008, vol. 5, p. 219-222.

[17] Cuckoo search. vol. 2011: *Wikipedia*, 2010.

[18] YANG, X.-S., DEB, S. Cuckoo Search via Lévy flights. In *Proc. World Congr. on Nature & Biologically Inspired Computing (NaBIC 2009)*. India, 2009, p. 210-214.

[19] YANG, X.-S., DEB, S. Engineering optimization by cuckoo search. *Int. J. Mathematical Modelling and Numerical Optimization,* Dec 2010, vol. 1, no. 4, p. 330-343.

[20] BLUM, C., ROLI, A. Metaheuristics in combinatorial optimization: Overview and conceptual comparison. *ACM Comput. Surv.,* 2003, vol. 35, p. 268-308.

[21] BROWN, C., LIEBOVITCH, L. S., GLENDON, R. Lévy flights in Dobe Ju/'hoansi foraging patterns. *Human Ecol.,* 2007, vol. 35, p. 129-138.

[22] REYNOLDS, A. M., FRYE, M. A. Free-flight odor tracking in Drosophila is consistent with an optimal intermittent scale-free search. *PLoS One,* 2007, vol. 2, p. e354.

[23] PAVLYUKEVICH, I. Lévy flights, non-local search and simulated annealing. *J. Comput. Physics,* 2007, vol. 226, p. 1830-1844.

[24] Lévy flight. Vol. 2011: Wikipedia, 2011.

[25] SHLESINGER, M. F. Search research. *Nature,* 2006, vol. 443, p. 281-282.

[26] MANTEGNA, R. N. Fast, accurate algorithm for numerical simulation of Lévy stable stochastic processes. *Physical Review E,* 1994, vol. 49, p. 4677-4683.

[27] Financial applications of stable distribution. In *Handbook of Stats*. vol. 14, 1996.

[28] CHAMBERS, J. M., MALLOWS, C. L., STUCK, B. W. A method for simulating stable random variables. *Journal of the American Stats. Assoc.,* 1976, vol. 71, p. 340-344.

About Authors…

Khairul Najmy ABDUL RANI received his B.Sc. in Electrical (Electronic) Engineering from the University of Missouri-Kansas City, USA in 1996 and M.Sc. in Information Technology from the Universiti Utara Malaysia in 2000, respectively. He is pursuing PhD in Communications Engineering in Universiti Malaysia Perlis. His research interests include antenna design, digital mobile communications and computer network.

Mohd. Fareq ABD MALEK received his B.Eng. (Hons) in Electronic and Communications Engineering from the University of Birmingham, UK in 1997, M.Sc. (Eng) in Microelectronic Systems and Telecommunications from the University of Liverpool, UK in 2004 and Ph.D in Electrical Engineering from the University of Liverpool, United Kingdom in 2008, respectively. His research interests include antenna, microwave engineering and applied electromagnetic.

Neoh SIEW-CHIN received her B.Tech. in Quality Control and Instrumentation from the School of Industrial Technology, Universiti Sains Malaysia (USM) in 2003 and Ph.D. from the School of Electrical and Electronic Engineering, USM, Nibong Tebal, Penang, Malaysia in 2009, respectively. Her research interests include production planning and optimization, intelligent manufacturing systems, evolutionary computing, operations management, and decision support systems.