

CONCENTRIC CIRCULAR ANTENNA ARRAY SYNTHESIS USING COMPREHENSIVE LEARNING PARTICLE SWARM OPTIMIZER

S. Elsaïdy^{1,*}, M. Dessouky², S. Khamis¹, and Y. Albagory²

¹Electronics and Electrical Communications Engineering Department, Faculty of Engineering, Tanta University, Egypt

²Electronics and Electrical Communications Engineering Department, Faculty of Engineering, Menoufia University, Egypt

Abstract—Concentric circular antenna array (CCAA) is synthesized to generate pencil beam with minimum side lobe level (SLL). The comprehensive learning particle swarm optimizer (CLPSO) is used for synthesizing a ten-ring CCAA with central element. This Synthesis is done by finding the optimum current excitation weights and interelement spacing of rings. The computational results show that sidelobe level is reduced to -40.5 dB with narrow beamwidth about 4.1° .

1. INTRODUCTION

Synthesizing the array pattern of antenna arrays has been a subject to several studies and investigations to improve the performance of mobile and wireless communication systems through efficient spectrum utilization, increasing channel capacity, extending coverage area, tailoring beam shape etc. [1]. Among the different types of antenna array, Concentric circular antenna array (CCAA) has become most popular in mobile and wireless communications. CCAA that contains many concentric circular rings of different radii and number of elements has several advantages including the flexibility in array pattern synthesis and design both in narrowband and broadband beamforming applications [2, 3]. CCAA is also favored in direction of arrival (DOA) applications since it provides almost invariant azimuth angle coverage. In addition, the frequency invariant characteristics of CCAA [4, 5] have been proved for wideband applications.

Received 25 November 2011, Accepted 28 December 2011, Scheduled 8 January 2012

* Corresponding author: Elsayed Ibrahim Elsaïdy (s.elsaïdy@gmail.com).

Uniform concentric circular array (UCCA) is CCAA where all the elements in the array are uniformly excited and the interelement spacing in individual ring is kept almost half of the wavelength. The sidelobe level in the UCCA drops to about 17.5 dB, especially at large number of rings with uniform excitation. UCCA has high directivity but it usually suffers from high sidelobe level [6]. Central element existence in UCCA reduces the sidelobe level while minor increase in the beamwidth [6].

Several beamforming techniques exist. For example, in [7, 8] tapering window is used to reduce the sidelobe level in UCCA, but reduction in the sidelobe level increases the beamwidth. Recently, search algorithms, such as genetic algorithms (GAs) [9] and Particle Swarm Optimization (PSO) [10], have been used in array pattern synthesis. GA has been used in [11] to optimize the interelement spacing and number of elements in each ring. The thinning and synthesis of pencil beam pattern with minimum sidelobe based on PSO have been discussed in [12, 13]. Some studies have been devoted to compare between the GA and PSO [14, 15] and a general conclusion has been reached. The PSO shows better performance due to its greater implementation simplicity and minor computational time.

In this paper, we use CLPSO [16] to optimize both current excitations and interelement spacing of the rings. The paper is

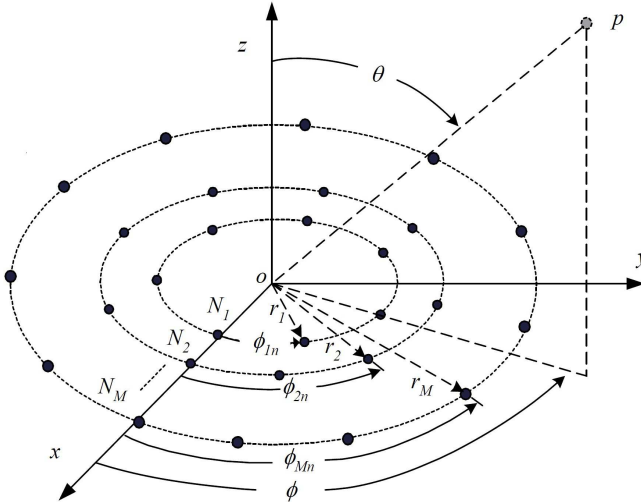


Figure 1. Concentric circular antenna array (CCAA).

organized as follows. In Section 2, Geometry of CCAA and design equation is presented. Then, in Section 3, particle swarm optimizer is employed. Computational results are presented in Section 4. Finally, we conclude the paper in Section 5.

2. GEOMETRY OF CCAA AND DESIGN EQUATION

2.1. Geometry of CCAA

The geometry of a concentric circular antenna array is shown in Figure 1. Where there are M concentric circular rings and the m th ring has a radius r_m and the corresponding number of elements is N_m where $m = 1, 2, \dots, M$. If all the elements (in all the rings) are assumed to be isotropic sources, then the radiation pattern of CCAA can be written in terms of its array factor only.

The array factor for the CCAA with a single element at the center (Figure 1) is given by [11]:

$$\text{AF} = 1 + \sum_{m=1}^M \sum_{i=1}^{N_m} W_m e^{j(Kr_m \sin \theta \cos(\phi - \phi_{mi}) + \alpha_{mi})} \quad (1)$$

where:

M = number of rings;

N_m = number of elements in ring m ;

W_m = excitation current of elements on m th ring;

r_m = radius of ring $m = \frac{N_m d_m}{2\pi}$;

d_m = interelement spacing of m th ring;

k = wave number = $\frac{2\pi}{\lambda}$;

λ = signal wavelength;

j = complex number;

θ = the zenith angle from the positive z axis;

ϕ = the azimuth angle from the positive x axis;

ϕ_{mi} = element angular separation measured from the positive x axis given by:

$$\phi_{mi} = 2\pi \left(\frac{i-1}{N_m} \right); \quad m = 1, \dots, M; \quad i = 1, \dots, N_m \quad (2)$$

α_{mi} = the phase difference between the individual elements in the array given by:

$$\alpha_{mi} = -kr_m \sin \theta_o \cos(\phi_o - \phi_{mi}); \quad m = 1, \dots, M; \quad i = 1, \dots, N_m \quad (3)$$

where θ_o and ϕ_o are the values of θ and ϕ ($\theta, \phi \in [-\pi, \pi]$) respectively where the highest peak of main lobe is obtained. If $\theta_o = 0$ and $\phi = \text{constant}$, the radiation pattern will be a broadside array pattern. The array factor in this case can be written as:

$$\text{AF} = 1 + \sum_{m=1}^M \sum_{i=1}^{N_m} W_m e^{j(Kr_m \sin \theta \cos(\phi - \phi_{mi}))} \quad (4)$$

Normalized absolute array factor in dB can be expressed as follows:

$$\text{AF (dB)} = 20 \times \left[\log_{10} \frac{|\text{AF}|}{|\text{AF}|_{\max}} \right] \quad (5)$$

2.2. Cost Function

The cost function or fitness function is important step for CLPSO algorithm that provides the link between the CLPSO and the problem. The cost function for this problem which is to be minimized is given by:

$$\text{CF} = W_1 \text{SLL}_{\max} + W_2 [\text{FNBW}_c - \text{FNBW}_u] \quad (6)$$

where SLL_{\max} is the maximum sidelobe level, FNBW_c , FNBW_u are the calculated first null beamwidth in radian for non uniform excitation case and for uniform excitation respectively, and W_1 , W_2 are positive weighting factors added to control the obtained results.

3. PARTICLE SWARM OPTIMIZER EMPLOYMENT

3.1. Particle Swarm Optimizers (PSO)

Particle Swarm Optimization (PSO) is an evolutionary algorithm that emulates the swarm behavior of bird flocking and fish schooling [17]. In PSO, each swarm member, called a particle or agent, represents a potential solution. The swarm initially has a population of random particles (solutions). Each particle adjusts its search direction by learning from its own experience and the other particles' experiences. Each particle velocity is updated by following two optimum values. The first one is the best solution (fitness) that has been achieved so far. This value is called *pbest*. The second one is the global best value obtained so far by any particle in the swarm. This best value is called *gbest*. Each D-dimensional vector of positions represents a possible solution [18]. The velocity and position of the *d*th dimension of the *i*th

particle at k th iteration are updated as follows [16]:

$$V_{i,d}^k = V_{i,d}^{k-1} + c_1 * rand1_{i,d}^k * (pbest_{i,d}^{k-1} - X_{i,d}^{k-1}) + c_2 * rand2_{i,d}^k * (gbest_d^k - X_{i,d}^{k-1}) \quad (7)$$

$$X_{i,d}^k = X_{i,d}^{k-1} + V_{i,d}^k \quad (8)$$

where $V_{i,d}^k$ is the i th particle velocity in the d th dimension, k denotes the current iteration and $k - 1$ the previous, $X_{i,d}^k$ represents position of i th particle in the d th dimension, $pbest_{i,d}^{k-1}$ is the best previous position yielding the best fitness value for the i th particle, $gbest_d^k$ is the best position discovered by the whole population. c_1 and c_2 are the acceleration constants reflecting the weighting of stochastic acceleration terms that pull each particle toward $pbest_{i,d}^k$ and $gbest_d^k$ positions, respectively, $rand1_{i,d}^k$ and $rand2_{i,d}^k$ are two random numbers in the range $[0, 1]$. A particle's velocity on each dimension is clamped to a maximum magnitude V_{max} . If $|V_{i,d}^k|$ exceeds a positive constant value $V_{d,max}$ specified by the user, then the velocity of that dimension is assigned to $sign(|V_{i,d}^k|)V_{d,max}$. The flowchart of the standard PSO is given in (Figure 2).

The classical Inertia Weight PSO (IWPSO) and Constriction Factor PSO (CFPSO) are the most common PSO algorithms [19]. The main deficiency of the classical IWPSO algorithm is the premature convergence when solving multimodal problems. In order to improve PSO's performance on complex multimodal problems, a variant of the PSO was proposed [16]. Several antenna design problems are multimodal and therefore require the use of an optimization method that does not get trapped easily in a local optimum [18].

3.2. Comprehensive Learning Particle Swarm Optimizer (CLPSO)

The new learning strategy in the CLPSO algorithm ensures that the diversity of the swarm is preserved to discourage premature convergence. This is achieved because each particle's velocity vector can be updated by using not only its own $pbest$, but also any other particle's $pbest$, which provides improved diversity in the population [18].

The velocity updating equation in CLPSO is given by:

$$V_{i,d}^k = w * V_{i,d}^{k-1} + c * rand_{i,d}^k * (pbest_{f_i(d),d}^k - X_{i,d}^{k-1}) \quad (9)$$

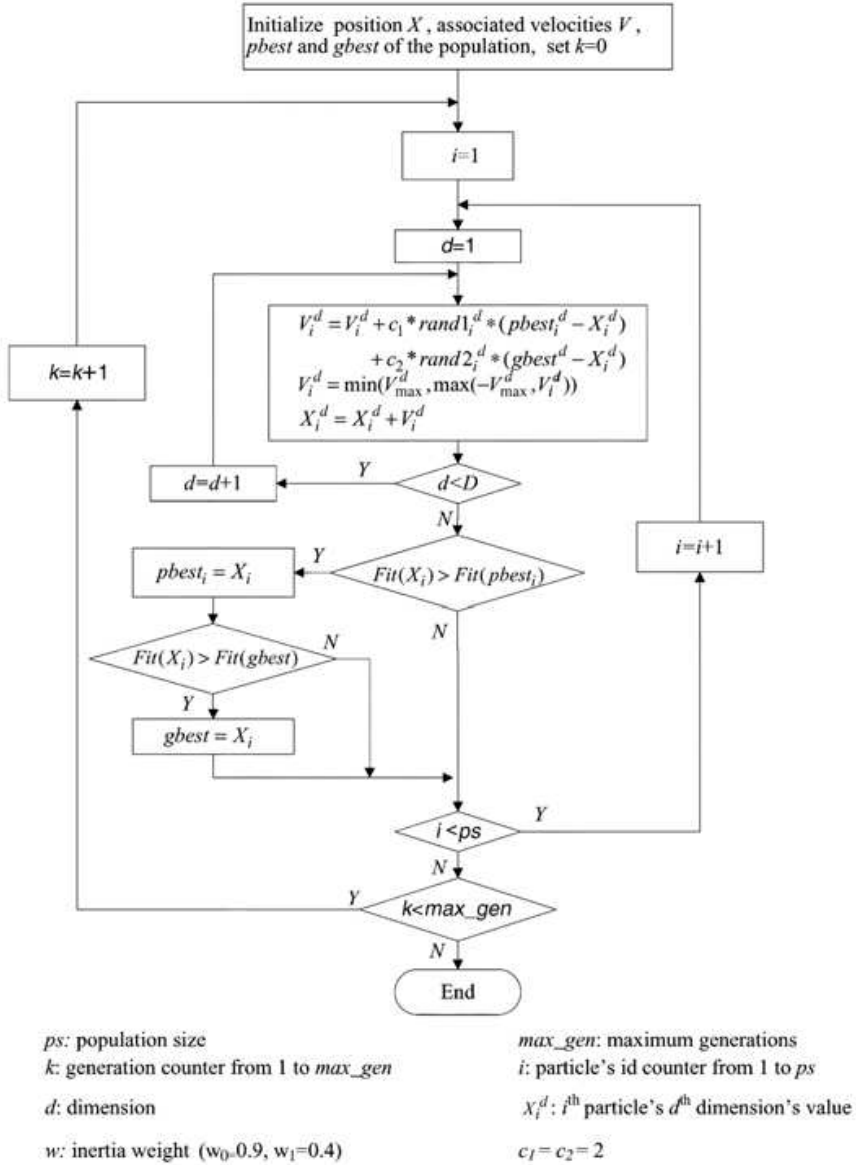


Figure 2. Flowchart of the conventional PSO.

where $f_i = [f_i(1), f_i(2), \dots, f_i(D)]$ defines which particles' *pbests* the particle i should follow. $pbest_{f_i(d),d}^t$ can be the corresponding dimension of any particle's *pbest* including its own *pbest*, and the decision depends on probability Pc , referred to as the learning probability, which can take different values for different particles. For each dimension of particle i , we generate a random number. If this random number is larger than Pc_i , the corresponding dimension will learn from its own *pbest*; otherwise it will learn from another particle's *pbest*. We employ the tournament selection procedure when the particle's dimension learns from another particle's *pbest* as follows:

- 1- We first randomly choose two particles out of the population which excludes the particle whose velocity is updated.
- 2- We compare the fitness values of these two particles' *pbest*'s and select the better one. In CLPSO, we define the fitness value the larger the better, which means that when solving minimization problems, we will use the negative function value as the fitness values.
- 3- We use the winner's *pbest* as the exemplar to learn from for that dimension. If all exemplars of a particle are its own *pbest*, we will randomly choose one dimension to learn from another particle's *pbest*'s corresponding dimension. The details of choosing $f_i(d)$ are given in (Figure 3).

We observe three main differences between the CLPSO and the original PSO [16].

- 1- Instead of using particle's own *pbest* and *gbest* as the exemplars, all particles' *pbest*'s can potentially be used as the exemplars to guide a particle's flying direction.
- 2- Instead of learning from the same exemplar particle for all dimensions, each dimension of a particle in general can learn from different *pbest*'s for different dimensions for a few generations. In other words, each dimension of a particle may learn from the corresponding dimension of different particle's *pbest*.
- 3- Instead of learning from two exemplars (*gbest* and *pbest*) at the same time in every generation as in the original PSO (7), each dimension of a particle learns from just one exemplar for a few generations. More details about CLPSO can be found in [16].

4. COMPUTATIONAL RESULTS

In this section, the capabilities of the CLPSO algorithm in the synthesis of CCAA with central element are demonstrated. For

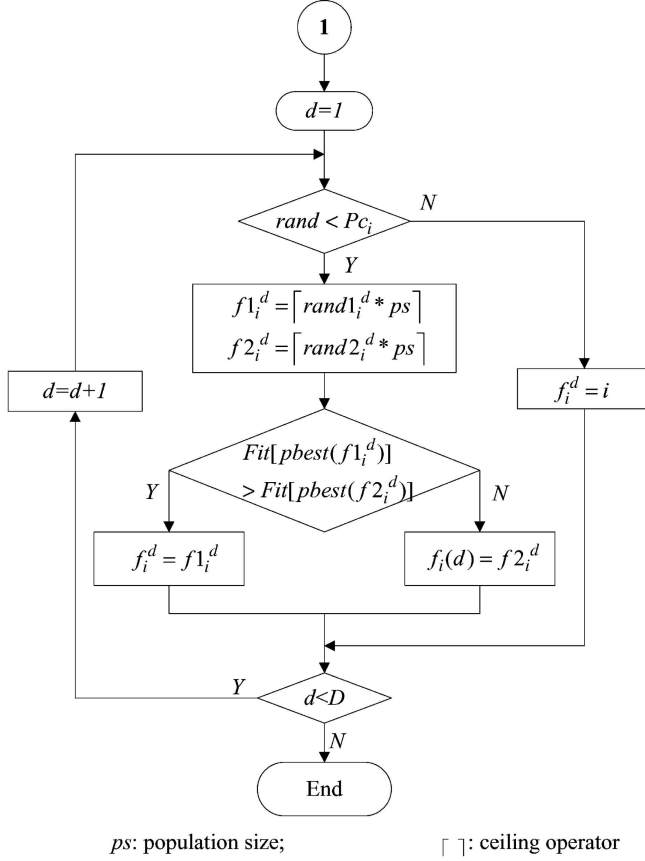


Figure 3. Selection of exemplar dimensions for particle i .

CLPSO algorithm, the population size is set to 120, iteration cycles for optimization = 1000, and $c = 1$. The simulation is made by MATLAB. We use MATLAB Release 2011a on a core 2 duo processor, 2.00 GHz with 2 GB RAM. The CLPSO algorithm is used to find the current excitations and interelement spacing for ten rings ($M = 10$) (5, 11, 17, 23, 29, 35, 41, 47, 53, and 59) using cost function (6). We assume that the number of elements in the 1st ring is 5 elements and the constant element increment is 6 elements per ring outwardly. The limit of interelement spacing is ranging from a half wavelength to one wavelength, d , ($d \in [\lambda/2, \lambda]$). For this case, $\theta_o = 0$ and $\phi = \pi/4$ are considered so that the main lobe will be at the origin, $BWFN_u$ is set to $14.26^\circ \times (\pi/180)$, which is about the beamwidth of the corresponding

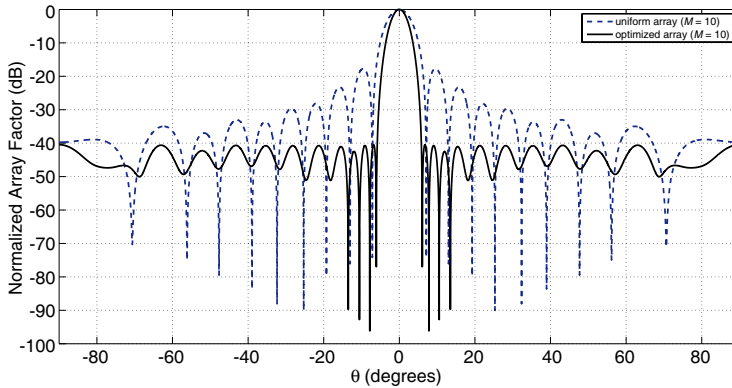


Figure 4. Radiation pattern for optimized CCAA and UCCA.

UCCA.

Figure 4 shows the radiation pattern of a ten-ring CCAA using the excitation currents and interelement spacing returned by the CLPSO (black solid line) and uniform excitation (blue dotted line). The returned current excitations and interelement spacing are given in Table 1. These results have achieved an SLL of -40.5 dB, a BWFN of 12.16 and a BWFN of 4.1° . Where HPBW is the half power beamwidth. We can notice the reduction in SLL with narrow beamwidth, but the paid cost is the increase in the aperture compared to UCCA. To achieve the same narrow beamwidth with SLL equals to -17.62 dB using UCCA, we need 15 rings ($M = 15$) (5, 11, 17, 23, 29, 35, 41, 47, 53, 59, 65, 71, 77, 83, and 89).

The same case ($M = 10$) is considered in [7] using Gaussian window ($\delta = 2.5$). The array factor obtained using the results of the CLPSO will be compared to the array factor obtained using the Gaussian window given by [7].

Figure 5 shows the array factor obtained using results in Table 1 (black solid line) compared to the results obtained using Gaussian window (blue dotted line). It can be shown that using CLPSO to synthesis CCAA gives a radiation pattern better than using the Gaussian window, where using the Gaussian window reduces the SLL to -38.96 dB while increasing HPBW to 8° and BWFN to 27.2° .

In this study, by using CLPSO we obtain CCAA with decreased SLL with narrow beamwidth but the array aperture is increased in comparison to ten rings synthesis using Gaussian window. For the same array aperture, Figure 6 shows the array factor obtained using results in Table 1 (ten rings) (black solid line) and compared to nineteen rings

Table 1. Excitation current weights (W_m), interelement spacing (d), SLL, BWFN and HPBW for optimized CCAA and UCCA.

	Optimized array W_m and d	UCCA with central element
current excitations (W_m) (W_1, W_2, \dots, W_m)	0.9357 0.9869 0.9345 0.8241 0.6658 0.5077 0.3837 0.2736 0.1732 0.1020	1
interelement spacing (d)	0.9416	0.5
sidelobe level (SLL)	-40.4654 dB	-17.69 dB
BWFN	12.16°	14.26°
HPBW	4.1°	5.98°

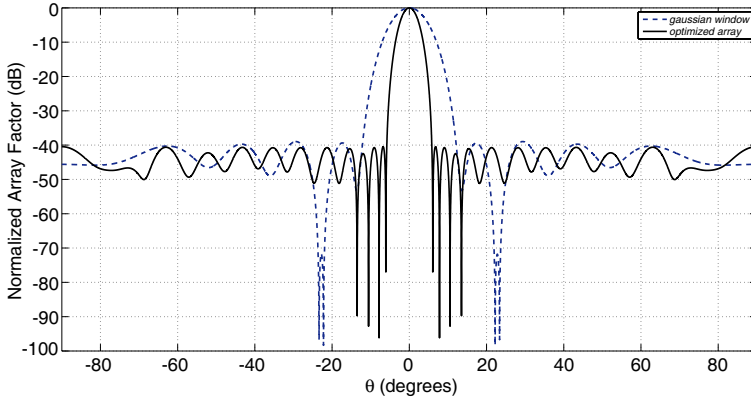


Figure 5. Radiation pattern for optimized CCAA and CCAA using Gaussian window weights [7].

(5, 11, 17, 23, 29, 35, 41, 47, 53, 59, 65, 71, 77, 83, 89, 95, 101, 107, and 113) that is obtained using Gaussian window [7] (blue dotted line). It can be show that optimized array using CLPSO has narrow beamwidth than the obtained from the Gaussian window and closed maximum SLL. The synthesized array using CLPSO has reduced 801 elements, i.e., a reduction of 71.5% of total elements used in case of same array aperture using Gaussian window. This will reduce the cost of designing the array substantially.

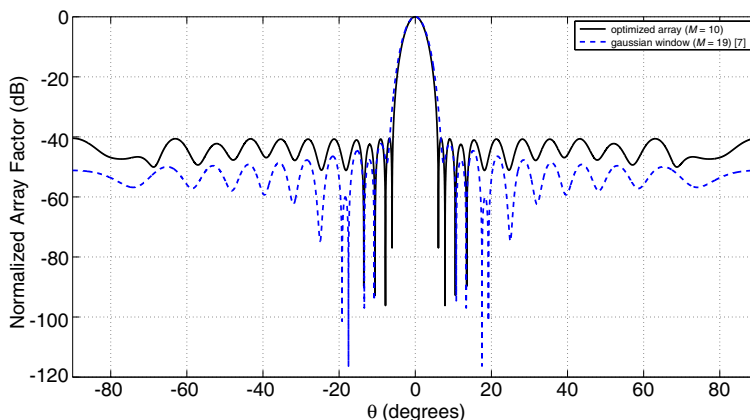


Figure 6. Radiation pattern for optimized CCAA and CCAA using Gaussian window weights [7].

The same case is taken in [20] but with eleven rings deployed (5, 11, 17, 23, 29, 35, 41, 47, 53, 59 and 65). This deployment uses Differential Invasive Weed Optimization Algorithm (DIWO) and the results are compared to CLPSO. The results of CLPSO are better than DIWO results where SLL is -25.12 dB where in the BWFN is equal to 13.81° .

5. CONCLUSION

The application of CLPSO to design CCAA has been demonstrated in this paper. Compared to the original PSO, CLPSO solving multimodal problems. CLPSO is best suited for solving multimodal problems that are common in several antenna design. The optimal design of a ten-ring CCAA with central element has been described. This optimal design is done by finding optimal excitation currents and optimal interelement spacing of rings. The simulated results reveal that the optimal design offers a considerable SLL reduction along with reduction of HPBW compared to the corresponding UCCA. The minimum achieved sidelobe level is -40.4654 dB for HPBW of 4.1° . The results are compared to CCAA which is synthesized by using Gaussian window at the same aperture size. The comparison shows a significant improve for the sidelobe level and the beamwidth with 70.5% reduction in the number of elements. This will reduce the cost of designing the antenna array substationally.

REFERENCES

1. Godara, L. C., "Applications of antenna arrays to mobile communications," *Proceedings of the IEEE*, Vol. 85, No. 7, 1031–1060, 1997.
2. Fletcher, P. N. and P. Darwood, "Beamforming for circular and semicircular array antennas for low-cost wireless lan data communications systems," *IEE Proceedings — Microwaves, Antennas and Propagation*, Vol. 145, No. 2, 153–158, Apr. 1998.
3. Li, Y., K. C. Ho, and C. Kwan, "A novel partial adaptive broadband beamformer using concentric ring array," *2004 Proceedings IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP'04*, Vol. 2, ii–177, 2004.
4. Chan, S. C. and C. K. S. Pun, "On the design of digital broadband beamformer for uniform circular array with frequency invariant characteristics," *IEEE International Symposium on Circuits and Systems, 2002, ISCAS 2002*, Vol. 1, I-693–I-696, 2002.
5. Chan, S. C. and H. H. Chen, "Uniform concentric circular arrays with frequency-invariant characteristics — theory, design, adaptive beamforming and DOA estimation," *IEEE Transactions on Signal Processing*, Vol. 55, No. 1, 165–177, Jan. 2007.
6. Dessouky, M. I., H. A. Sharshar, and Y. A. Albagory, "Efficient sidelobe reduction technique for small-sized concentric circular arrays," *Progress In Electromagnetics Research*, Vol. 65, 187–200, 2006.
7. Dessouky, M. I., H. A. Sharshar, and Y. A. Albagory, "Optimum normalized-Gaussian tapering window for side lobe reduction in uniform concentric circular arrays," *Progress In Electromagnetics Research*, Vol. 69, 35–46, 2007.
8. Dessouky, M., H. Sharshar, and Y. Albagory, "An approach for Dolph-Chebyshev uniform concentric circular arrays," *Journal of Electromagnetic Waves and Applications*, Vol. 21, No. 6, 781–794, 2007.
9. Johnson, J. M. and Y. Rahmat Samii, "Genetic algorithms in electromagnetics," *Antennas and Propagation Society International Symposium, 1996, AP-S, Digest*, Vol. 2, 1480–1483, Jul. 1996.
10. Robinson, J. and Y. Rahmat-Samii, "Particle swarm optimization in electromagnetics," *IEEE Transactions on Antennas and Propagation*, Vol. 52, No. 2, 397–407, Feb. 2004.
11. Haupt, R. L., "Optimized element spacing for low sidelobe concentric ring arrays," *IEEE Transactions on Antennas and Propagation*, Vol. 56, No. 1, 266–268, Jan. 2008.

12. Mandal, D., D. Sadhu, and S. P. Ghoshal, "Thinned concentric circular array antennas synthesis using improved particle swarm optimization," *Proc. of International Conference on Advances in Computer Engineering, ACEEE*, 2011.
13. Chatterjee, A., G. K. Mahanti, and P. R. S. Mahapatra, "Optimum ring spacing and interelement distance for sidelobe reduction of a uniform concentric ring array antenna using differential evolution algorithm," *IEEE Communication Systems, ICCS*, 2010.
14. Boeringer, D. W. and D. H. Werner, "Particle swarm optimization versus genetic algorithms for phased array synthesis," *IEEE Transactions on Antennas and Propagation*, Vol. 52, No. 3, 771–779, Mar. 2004.
15. Shavit, R. and I. Taig, "Array pattern synthesis using neural networks with mutual coupling effect," *IEEE Proceedings — Microwaves, Antennas and Propagation*, Vol. 152, No. 5, 354–358, Oct. 2005.
16. Liang, J. J., "Comprehensive learning particle swarm optimizer for global optimization of multimodal functions," *IEEE Transactions on Evolutionary Computation*, Vol. 10, No. 3, 281–295, Jun. 2006.
17. Kennedy, J. and R. Eberhart, "Particle swarm optimization," *Proceedings IEEE International Conference on Neural Networks*, Vol. 4, 1942–1948, 1995.
18. Goudos, S. K., V. Moysiadou, T. Samaras, K. Siakavara, and J. N. Sahalos, "Application of a comprehensive learning particle swarm optimizer to unequally spaced linear array synthesis with sidelobe level suppression and null control," *IEEE Antennas and Wireless Propagation Letters*, Vol. 9, 125–129, 2010.
19. Clerc, M., "The swarm and the queen: Towards a deterministic and adaptive particle swarm optimization," *IEEE Proceedings of the 1999 Congress on Evolutionary Computation, 1999, CEC'99*, Vol. 3, 1999.
20. Basak, A., S. Pal, S. Das, and A. Abraham, "Circular antenna array synthesis with a differential invasive weed optimization algorithm," *10th IEEE International Conference on Hybrid Intelligent Systems, HIS*, 153–158, 2010.