

Generation of Low Side Lobe Difference Pattern using Nature Inspired Metaheuristic Algorithms

M. Vamshi Krishna
Research Scholar,
Dept. of ECE, Centurion
University of Technology
and Management,
Paralakhemundi, Odisha,
India.

G. S. N. Raju
Honorary Distinguished
Professor,
Dept. of ECE,
AU College of Engineering (A),
Andhra University,
Andhra Pradesh India

S. Mishra
Professor and Head of the
Dept.,
Dept. of ECE,
Centurion University of
Tech. & Management,
Paralakhemundi, Odisha, India

ABSTRACT

Pattern synthesis is one of the most important aspects in antenna design. Arrays are more flexible to produce desired radiation characteristics. Difference patterns are usually generated with conventional techniques and there is less control on side lobes. In view of this, optimization techniques are applied to synthesize and produce such patterns optimally. The simulated patterns are produced for different array configurations and the patterns are presented using Cuckoo Search Algorithm (CSA) and Accelerated Particle Swarm Optimization (APSO).

Keywords

Difference Pattern, Sidelobe Reduction, Antenna Array, Cuckoo Search Algorithm (CSA), Particle Swarm Optimization (PSO), Accelerated Particle Swarm Optimization (APSO)

1. INTRODUCTION

Radars are highly effective in ascertain the location of remote objects, but accuracy is greatly reduced when radar is jammed. The difference pattern consists of a null in the boresight direction with two major lobes adjacent to null. The null in the difference pattern is suppressing the jamming source and tracking accuracy was improved [1]. The sum patterns are generated by several methods but the methods of generation of difference patterns are limited. In difference patterns, the depth of the null is significant and the difference slope is not enough for many applications .It is required to produce them for optimum performance.

In this method it is difficult to obtain deep null and high difference slope but for fine angular tracking difference pattern should have deep null at boresight and high difference slope for accurate angular tracking. Bayliss [2,3] synthesized the difference pattern for continuous line source but was not able to find a limiting form for the ideal difference Pattern. Elliott [6] proposed a design technique which yields difference pattern with arbitrary sidelobe topographies together with the requisite continuous aperture distributions.

Yanchang et al. [7] proposed a new design method which determines the aperture distribution for difference pattern from Taylor distribution. The peak sidelobe level in the difference pattern can be easily controlled in this method. A method using Dolph-Chebyshev distribution for difference pattern generation is presented in the paper by Yanchan Guo [9].

So, the present work is focused on generation of difference patterns with deep null in boresight and very low sidelobe

levels. Optimization techniques like Cuckoo Search Algorithm (CSA) and Accelerated Particle Swarm Optimization (APSO) has been applied to different problems in antenna arrays. In the present work these techniques are used to find suitable amplitude excitations to generate required difference patterns. The patterns reported has sidelobe level around -35 dB.

2. FORMULATION

The difference pattern from a continuous line source is also obtained from

$$E_d(u) = \left[\int_{-1}^0 A(x) e^{j\frac{2\pi L}{\lambda} [ux + \alpha]} dx + \int_1^0 A(x) e^{j\frac{2\pi L}{\lambda} [ux + \alpha]} dx \right] \quad (1)$$

Here A(x) is excitation function

$2L/\lambda =$ Normalized array length

$u = \sin \Theta$

$x =$ position on line source

$\alpha =$ excitation phase

to generate a null in the boresight directions 180° phase shift is introduced to one half of the array i.e.,

$\alpha = 0$ for $-1 \leq x \leq 0$

$\alpha = \pi$ for $0 \leq x \leq 1$ (2)

Substituting equation (2) in equation (1), then radiation pattern is now given by

$$E(u) = \int_{-1}^0 A(x) e^{j\frac{2\pi L}{\lambda} ux} dx + \int_0^1 A(x) e^{j\frac{2\pi L}{\lambda} (ux + \pi)} dx \quad (3)$$

Let $a = 2\pi/L$ then

$$E(u) = \int_{-1}^0 A(x) e^{jau x} dx + \int_0^1 A(x) e^{ja(ux + \pi)} dx \quad (4)$$

$$E(\Theta) = A(x) \left\{ \int_{-1}^0 \cos(au x) + j \sin(au x) dx + A(x) \int_0^1 \cos(au x + \pi) + j \sin(au x + \pi) dx \right\} \quad (5)$$

As the line source is only a theoretical concept, a discrete array is used here .the radiation integral is replaced by a finite summation containing number of terms equal to the number of elements in the array.

The radiation pattern is now given by

$$E(u) = \sum_{n=1}^N A(X_n) e^{j \frac{2\pi L}{\lambda} [u x_n + \alpha(x_n)]} dx \quad (6)$$

Where

$$\alpha=0 \text{ for } -1 \leq x_n \leq 0$$

$$\alpha= \pi \text{ for } 0 \leq x_n \leq 1.$$

3. CUCKOO SEARCH ALGORITHM (CSA)

Cuckoo search is a new metaheuristic search algorithm, based on cuckoo bird's behaviour and has been recently developed by Yang and Deb in 2009[10]. Cuckoo search is a very new population heuristic algorithm for global optimization and it is one of the evolutionary technique, inspired by the reproduction strategy of cuckoos.

At the most basic level, 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. Each egg in a nest represents a solution, and a cuckoo egg represents a new solution [10].

The aim is to use the new and potentially better solutions (cuckoos) to replace a not-so-good solution in the nests. In the simplest form, each nest has one egg. The algorithm can be extended to more complicated cases in which each nest has multiple eggs representing a set of solutions for simplicity in describing the cuckoo search, the 3 idealized assumptions:

1. Each cuckoo lays one egg at a time, and dumps its egg in a randomly chosen nest.
2. The best nests with high quality of eggs will carry over to the next generation.
3. The number of available hosts nests is fixed, and the egg laid by a cuckoo is discovered by the host bird with a probability lies in between (0 and 1).

In this case, the host bird can either throw the egg away or abandon the nest to build a completely new nest in a new location. Random-walk style search is better performed by Lévy flights [10, 14] rather than simple random walk. An important advantage of this algorithm is its simplicity. In fact, comparing with other population- or agent-based metaheuristic algorithms such as particle swarm optimization, there is essentially only a single parameter in pa CSA (apart from the population size n. Therefore, it is very easy to implement.

The best nest is chosen as the optimum variable and in this case it refers to the new current amplitude of linear antenna array elements $x_i^{(t+1)}$ for, say cuckoo i , a Lévy flight is performed [14]

$$X_i^{(t+1)} = X_i^{(t)} + \alpha \oplus \text{Lévy}(\lambda), \quad (6)$$

The Levi flight equation represents the stochastic equation for random walk as it depends on the current position and the transition probability (second term in the equation). Where α is the step size related to problem of interest. In most cases, the values are $\alpha = (1)$. The product \oplus means entry-wise multiplications. Lévy flights essentially provide a random walk while their random steps are drawn from a Lévy distribution for large steps

$$\text{Lévy} \sim u = t^{-\lambda}, (1 < \lambda \leq 3) \quad (7)$$

Where, in the above equation, $t^{-\lambda}$, is a parameter dealing with fractal dimension and t being the step size. The value of probability Pa used in this paper is 0.25 as used by Yang and Deb [10].

The pseudo code of the CSA is shown below:

Begin

Objective function $f(X)$, $X=(x_1, x_2, \dots, x_d)$

Generate initial population of n host nests
 $X_i (i=1,2,\dots,n)$

While ($t < \text{Max Generation}$) or (stop criterion)

Get a cuckoo randomly by Levy flights evaluate its quality/fitness F_i

Choose a nest among n (say, j) randomly

If ($F_i > F_j$),

Replace j by the new solution;

End

A fraction (pa) of worse nests are abandoned and new ones are built;

Keep the best solutions (or nests with quality solutions);

Rank the solutions and find the current best

End while

Post process results and visualization

Figure 1: CSA Pseudo code

4. ACCELERATED PARTICLE SWARM ALGORITHM (APSO):

PSO was introduced by Russel Eberhart and James Kennedy in the year 1995. PSO is an evolutionary algorithm inspired by imitating social behavior of schools of fish or flocks of birds [15]. PSO consists of agents called particles where each particle represents a potential solution in the search space. Interaction and sharing of information among the particles is the main source of swarms searching capability. All the particles are allowed to move systematically in the search space.

The basic principle in their algorithm is that each and every particle adjusts its coordinates according to its own experience and that of the other particles [16]. This implies that each particle has memory of its own or self best position called as pbest. Another best value obtained so far by the neighborhood experiences is called as gbest. The working of PSO is associated with velocity vector component and position vector component.

The position and velocity of the particles are updated by using the following equations:

$$V_n(t+1) = w \cdot V_n(t) + c_1 \cdot r_1 (pbest_n - X_n(t)) + c_2 \cdot r_2 (gbest - X_n(t)) \quad (9)$$

$$X_n(t+1) = X_n(t) + V_n(t+1) \quad (10)$$

Here, w is the inertia coefficient of the particle which plays an important role in PSO was introduced by Eberhart and Shi [21]. $V_n(t+1)$ is the current particle's velocity, $V_n(t)$ is the previous particle's velocity, $X_n(t)$ is the current particle's position, $X_n(t)$ is the previous particle's position. r_1 and r_2 are

uniformly distributed random numbers in the range [0,1]. c_1 and c_2 are the acceleration constants which control the relative effect of the pbest and gbest particles. pbest_n is the current pbest value, gbest is the current gbest value. The PSO algorithm is developed in 4 steps which will stop when the exit criteria is met.

For

Each particle Initialize particle

End For

Do

For each particle Calculate fitness value

If

if fitness value is better than the best fitness value (pBest in history)

Set current value as the new pBest

End

Choose the particle with the best fitness value of all the particles as the gBest

For each particle Calculate particle velocity according to previous equations

Update particle position according to previous equations

End

While maximum iterations or minimum error criteria is not obtained

Figure 2: PSO pseudo code

In recent years, various attempts have been made to improve the performance of standard PSO. One among the variants of PSO is APSO which extends the standard PSO algorithm. The standard PSO uses both pbest and gbest value. In APSO, only gbest is used as it could accelerate the convergence of the algorithm. The version of APSO was introduced by Xin She Yang in the year 2008. The position and velocity vectors of the particles are initialized randomly and are updated with time. The velocity vector is generated by using the following equation.

$$V_n(t+1) = w \cdot V_n(t) + \alpha \cdot c_n + \beta(gbest - X_n(t)) \quad (11)$$

Here c_n is the random variable chosen between the interval (0, 1).

The position vector is updated by using the following equation

$$X_n(t+1) = X_n(t) + V_n(t+1) \quad (12)$$

In order to increase the convergence even further, the above two equations are combined into a single equation which is given by

$$X_n(t+1) = (1 - \beta)X_n(t) + \alpha \cdot c_n + \beta(gbest) \quad (13)$$

The typical values of APSO are $\alpha = 0.1$ to 0.4 and $\beta = 0.1$ to 0.7 . Here α is taken as 0.2 and β as 0.5 . The advantage of using this algorithm is to reduce the randomness as the numbers of iterations proceed. This can be done by using a monotonically decreasing function given by

$$\alpha = \gamma^t \quad (0 < \gamma < 1) \quad (14)$$

Here γ is the control parameter which is taken as 0.97 and t is the number of iterations or time steps where $t \in [0, t_{max}]$ and t_{max} is the maximum value of the iterations.

For

Initialize the swarm population within the search space with random positions and velocities.

End for

Do

Evaluate the fitness function of each particle in the swarm.

Repeat for each particle in the swarm.

Find the gbest value.

Evaluate the position of the particle using the equation. (13).

End

While maximum iterations or minimum error criteria is not obtained

Figure 3: APSO pseudo code

5. RESULTS

In this present work Optimization Algorithms like CSA & APSO are used to obtain difference pattern with deep null at the boresight and lowest possible sidelobe levels. The amplitude excitation coefficients obtained by applying above optimization techniques for typical N values of 40 and 50 element array respectively. The patterns are numerically computed and are shown in figures (4-23).

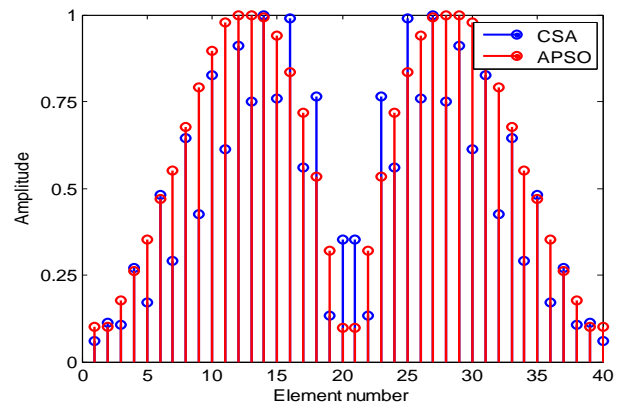


Figure 4: Element amplitude weights obtained with $d=0.40$ by CSA and APSO method for $N=40$

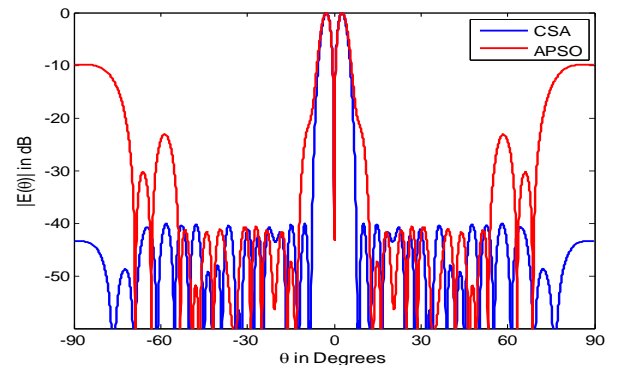


Figure 5: Optimized Difference Pattern with $d=0.40$ obtained by CSA and APSO method for $N=40$

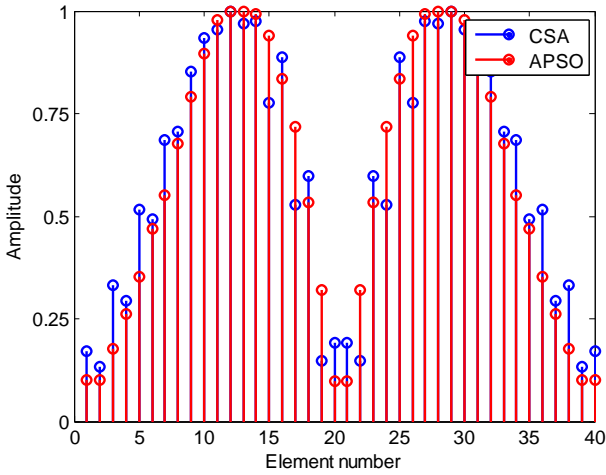


Figure 6: Element amplitude weights obtained with $d=0.45$ by CSA and APSO method for $N=40$

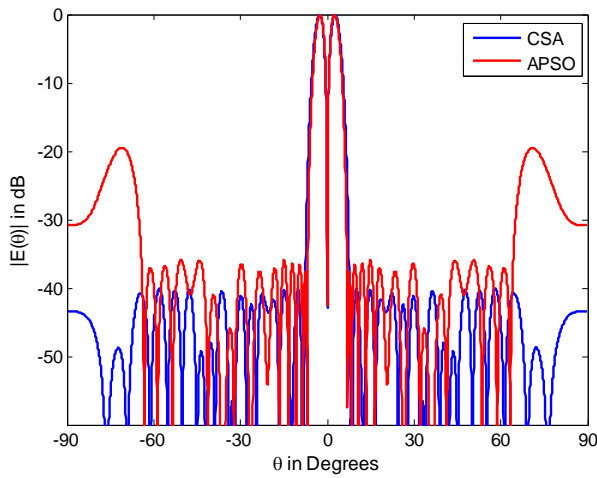


Figure 7: Optimized Difference Pattern with $d=0.45$ obtained by CSA and APSO method for $N=40$

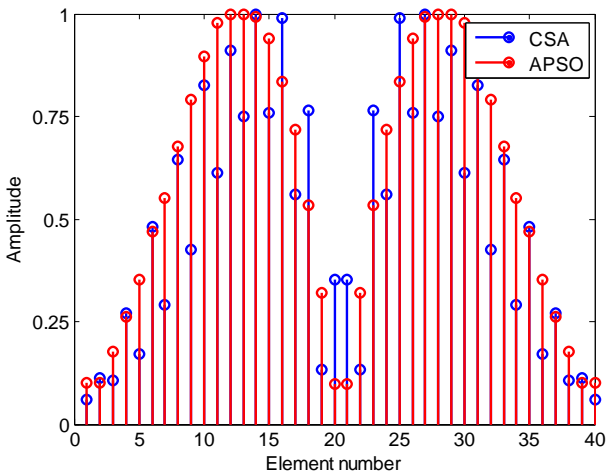


Figure 8: Element amplitude weights obtained with $d=0.5$ by CSA and APSO method for $N=40$

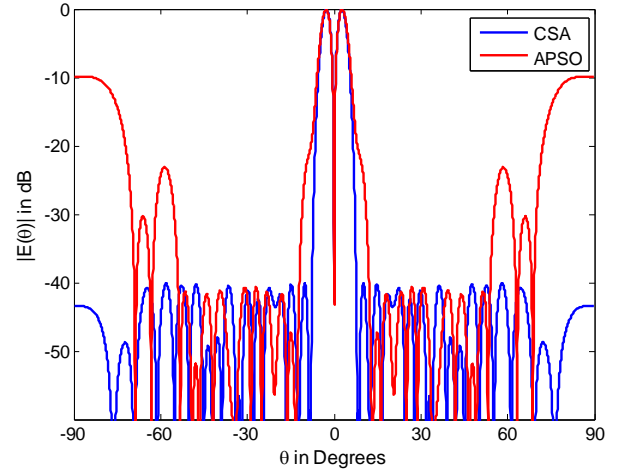


Figure 9: Optimized Difference Pattern obtained with $d=0.5$ by CSA and APSO method for $N=40$

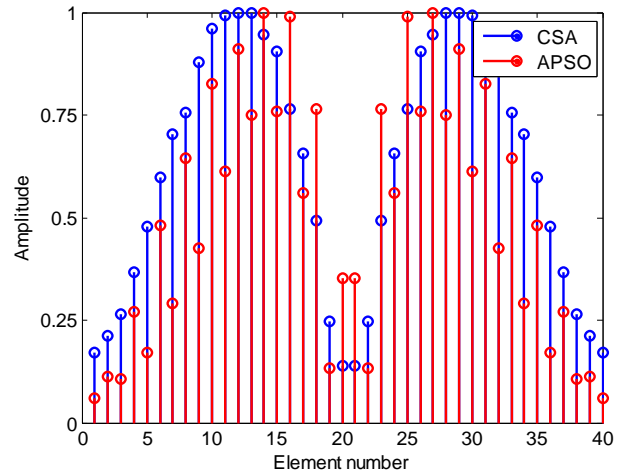


Figure 10: Element amplitude weights obtained with $d=0.55$ by CSA and APSO method for $N=40$

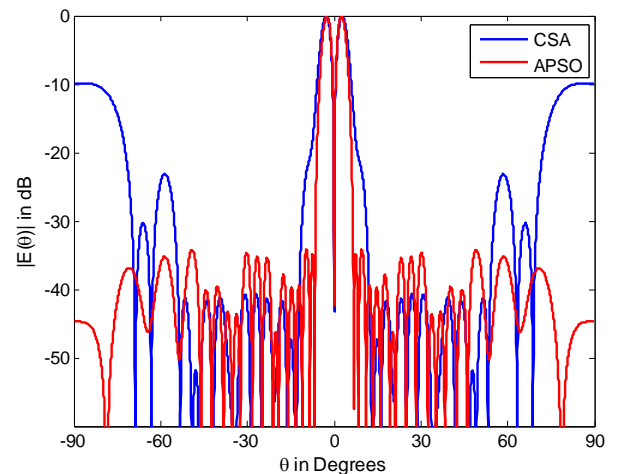


Figure 11: Optimized Difference Pattern with $d=0.55$ obtained by CSA and APSO method for $N=40$

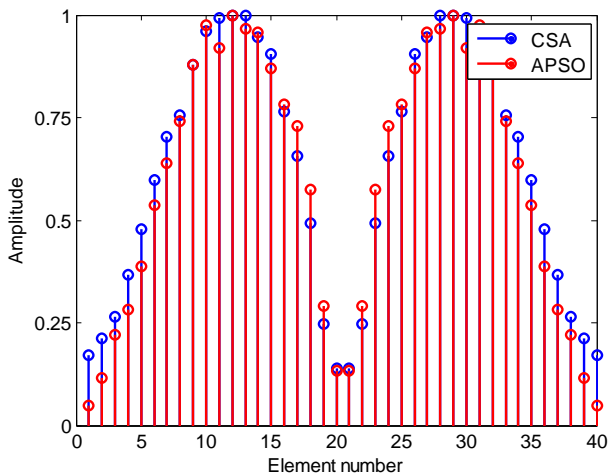


Figure 12: Element amplitude weights obtained with $d=0.60$ by CSA and APSO method for $N=40$

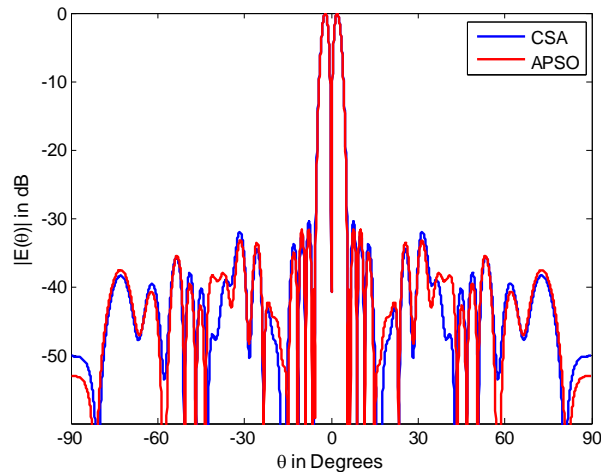


Figure 15: Optimized Difference Pattern with $d=0.40$ obtained by CSA and APSO method for $N=50$

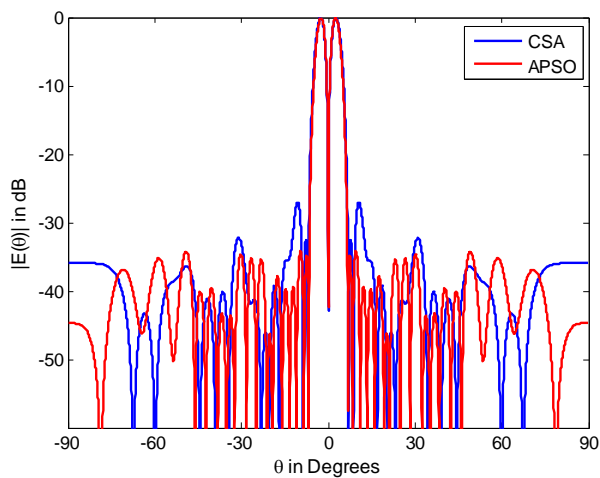


Figure 13: Optimized Difference Pattern with $d=0.60$ obtained by CSA and APSO method for $N=40$

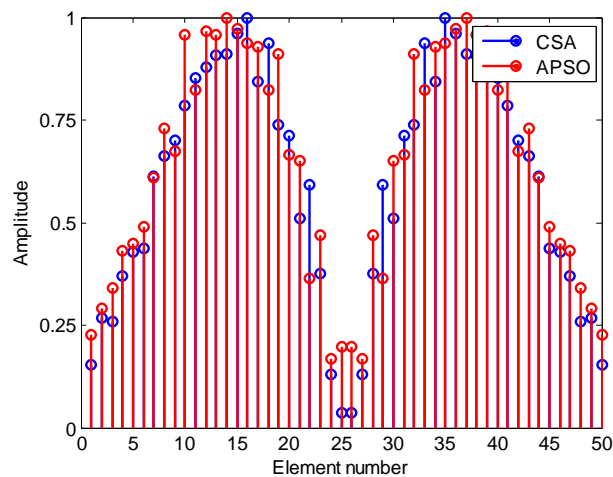


Figure 16: Element amplitude weights obtained with $d=0.45$ by CSA and APSO method for $N=50$

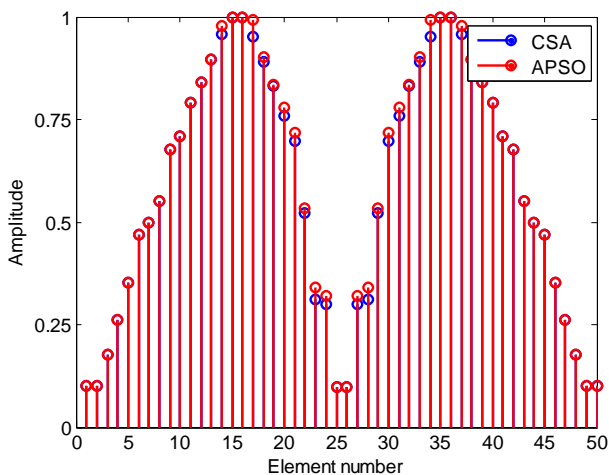


Figure 14: Element amplitude weights obtained with $d=0.40$ by CSA and APSO method for $N=50$

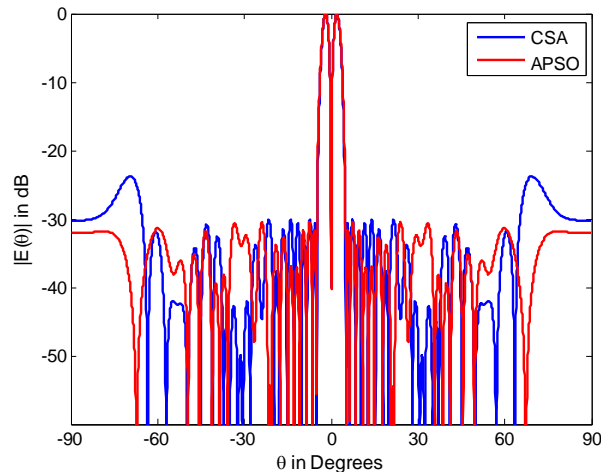


Figure 17: Optimized Difference Pattern with $d=0.45$ obtained by CSA and APSO method for $N=50$

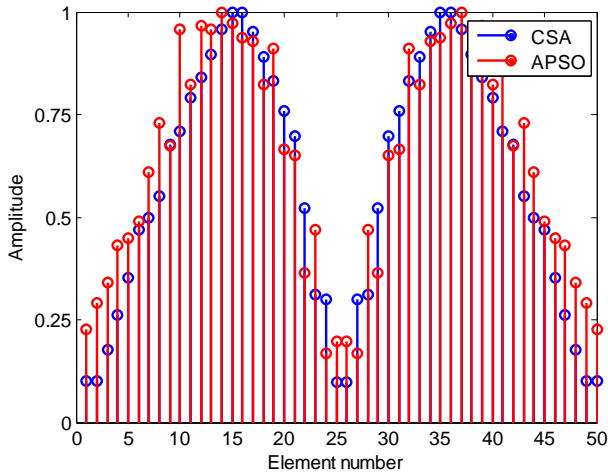


Figure 18: Element amplitude weights obtained with $d=0.5$ by CSA and APSO method for $N=50$

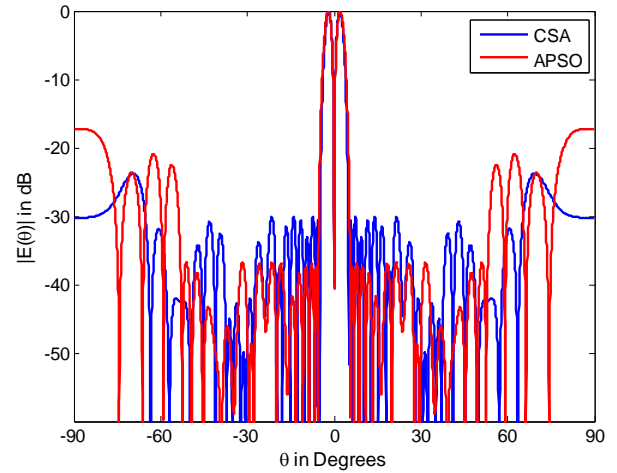


Figure 21: Optimized Difference Pattern with $d=0.55$ obtained by CSA and APSO method for $N=50$

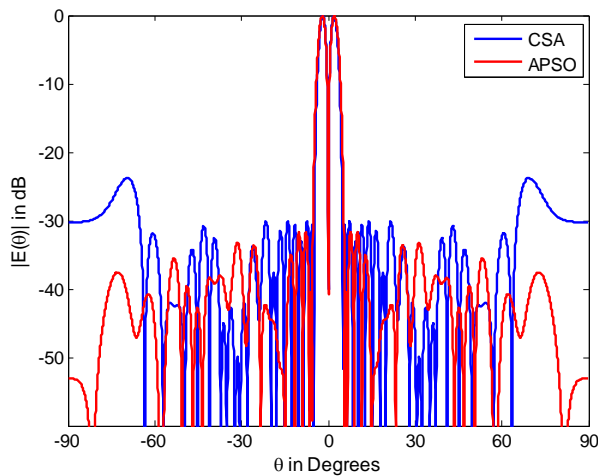


Figure 19: Optimized Difference Pattern obtained with $d=0.5$ by CSA and APSO method for $N=50$

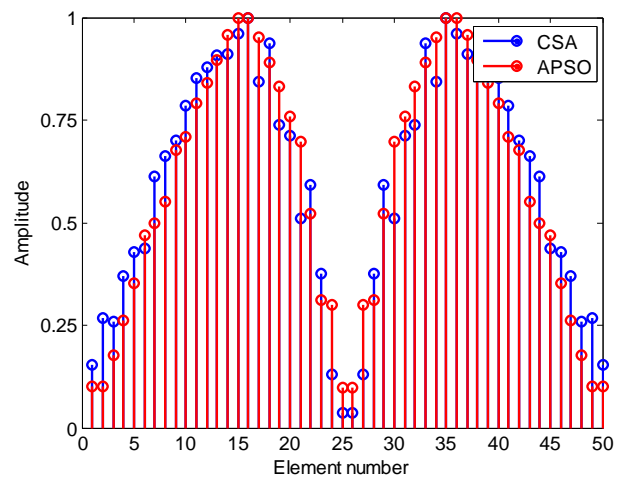


Figure 22: Element amplitude weights obtained with $d=0.60$ by CSA and APSO method for $N=50$

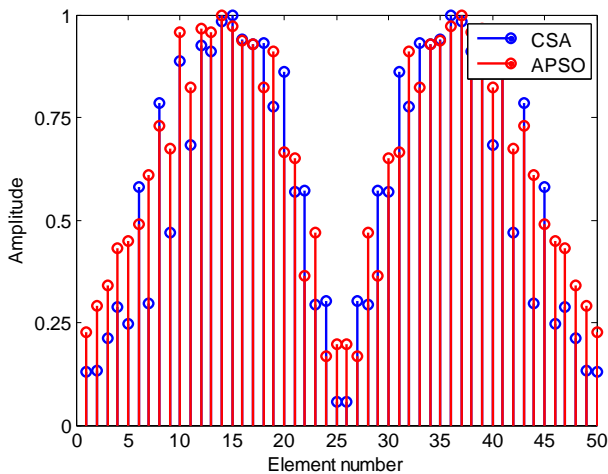


Figure 20: Element amplitude weights obtained with $d=0.55$ by CSA and APSO method for $N=50$

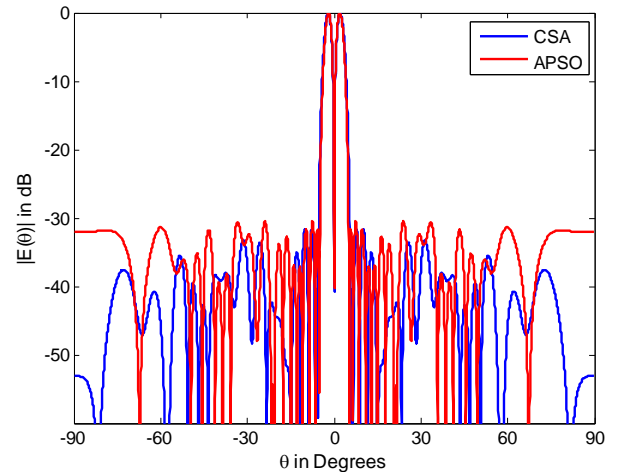


Figure 23: Optimized Difference Pattern with $d=0.60$ obtained by CSA and APSO method for $N=50$

6. CONCLUSION

It is evident from the above results that the difference patterns have ultra low sidelobe levels and deep null in their boresight makes them suitable for EMC applications. These patterns are very useful in high resolution tracking radars which are applicable to defence applications for identifying the

targets. The sidelobe level for APSO is around -38dB whereas for CSA it is about -35 db. As the number of elements increase the beam width has no change but for less values of elements the beam width varies. In future it can be done for greater values of N and achieve minimum side lobe levels so that these are very much useful for angular tracking..

7. REFERENCES

- [1] T. Isernia, F.J. Ares, O.M. Bucci, "A hybrid approach for the optimal synthesis of pencil beams through array antennas," *IEEE Transactions on Antennas and Propagation*, Vol. 52, pp. 2912-2918, November 2004.
- [2] E. Rajo-Iglesias and T. Quevedo-Ternel, "Linear array synthesis using ant colony optimization based algorithm," *IEEE Transactions on Antennas and Propagation Magazine*, Vol. 49, pp. 70-79, April 2007.
- [3] Song-Han Yang and Jean-Fu Kian, "Optimization of Asymmetrical Difference Pattern with Memetic Algorithm", *IEEE Transactions on Antennas and Propagation*, Vol. 62, No. 4, April 2014.
- [4] N.G. Gomez, J. Rodriguez, K.L. Melde and K. McNeil, "Design of low sidelobe linear arrays with high aperture efficiency and interference nulls," *IEEE Transactions on Antennas and Propagation*, Vol. 8, pp. 607-610, 2009.
- [5] G. S. N. Raju, "Antennas and Propagation," Pearson Education 2005.
- [6] R.S. Elliott, "Design of line source antennas for difference patterns with Sidelobe of individually arbitrary heights," *IEEE Transactions on Antennas and Propagation*, Vol. 24, No. 3, pp. 310-316, 1976.
- [7] Yanchang, G, Meng, M, Xianrong, S and Nenghang, F, "Design of aperture distributions for difference pattern from Taylor distributions," *Antennas and Propagation Society International Symposium*, Vol. 3, pp. 1586-1589, July 1993.
- [8] S. Pal, S. Das, and A. Basak, "synthesis of difference patterns for monopulse antennas with optimal combination of array-size and number of subarrays — a multi-objective optimization approach progress in electromagnetics research b, vol. 21, 257–280, 2010.
- [9] Yan Chang Guo and Meng Miao, "Design of antenna array aperture distributions for difference pattern using Dolph-Chebyshev distribution," *Asia-Pacific Microwave Conference (APMC) Proceedings*, Vol. 1, pp. 1-58, 1993.
- [10] Yang, X.-S., and Deb, S., "Engineering Optimization by Cuckoo Search", *Int. J. Mathematical Modelling and Numerical Optimization*, Vol. 1, No. 4, 330–343, 2010.
- [11] A. Chakrabarty, B.N. Das and G.S. Sanyal, "Beam shaping using nonlinear phase distribution in a uniformly spaced array," *IEEE Transactions on Antennas and Propagation*, Vol. 30, pp. 1031-1034, 1982.
- [12] F. John Deford and Om P. Gandhi, "Phase only synthesis of minimum peak sidelobe patterns for linear and planar arrays," *IEEE Transactions on Antennas and Propagation*, Vol. 36, pp. 191-201, February 1988.
- [13] A J Willis, "Low Sidelobe Difference Beams for Small Phased Arrays" *QinetiQ Malvern, QinetiQ 2005*.
- [14] Yang, X. S. and Deb, S., 2009. 'Cuckoo search via Lévy flights', *Proceedings of World Congress on Nature & Biologically Inspired Computing (NaBIC 2009, India)*, IEEE Publications, USA, pp. 210-214, 2007.
- [15] M.M. Khodier, C.G. Christodoulou, "Linear array geometry with minimum sidelobe level and null control using particle swarm optimization," *IEEE Transactions on Antennas and Propagation*, Vol. 53, pp. 2674-2679, August, 2005.
- [16] Yang, X. S., 'Biology-derived algorithms in engineering optimization' (chapter 32), in *Handbook of Bioinspired Algorithms and Applications* (eds Olariu & Zomaya), Chapman & Hall / CRC, 2005.
- [17] D. Marciano and F. Duran, "Synthesis of antenna arrays using genetic algorithms," *IEEE Antennas and Propagation Magazine*, Vol. 42, pp. 12-20, June 2000.
- [18] D Mandal, S K Ghosal, S Das, S Bhattacharjee and A K Bhattacharjee, "Improvement of radiation pattern for linear antenna arrays using genetic algorithm," *International conference on recent trends in Information, Telecommunications & Computing*, pp. 126-129, 12-13 March 2010.
- [19] H. Steyskal, "Simple method for pattern nulling by phase perturbation," *IEEE Transactions on Antennas and Propagation*, Vol. 31, pp. 163-166, January 1983.
- [20] R.A. Shore, "Nulling at symmetric pattern location with phase-only weight control," *IEEE Transactions on Antennas and Propagation*, Vol. 32, pp. 530-533, May 1984.
- [21] P. Rocca, I. Manica, and A. Massa, "Ant colony based hybrid approach for optimal compromise sum-difference patterns synthesis", *Technical Report # DISI-11-020* January 2010.