Progress In Electromagnetics Research M, Vol. 86, 153–162, 2019

A Two-Step Method for the Low-Sidelobe Synthesis of Uniform Amplitude Planar Sparse Arrays

Lanmei Wang[1], Xin-Kuan Wang[2, *], Gui-Bao Wang[2], and Jian-Ke Jia[2]

Abstract—A two-step method combining the algorithms of iterative Fourier transform (IFT) and differential evolution (DE), called IFT-DE, is proposed in this paper for the low sidelobe synthesis of a uniform amplitude planar sparse array (PSA). Firstly, the entire aperture of the array is divided into a set of square lattices that are spaced at half wavelength. Then the elements are forced to be located on the lattices through performing IFT, so that a planar thinned array (PTA) is formed across the aperture. Undoubtedly the interval between adjacent elements of the PTA is an integer multiple of half wavelength. In the second step, for each column of PTA the elements spaced greater than or equal to a wavelength are selected as the candidates whose locations need to be optimized by DE procedure, as long as the renewed inter-element spacing is not less than half wavelength. Consequently, a PSA with reduced sidelobe level may be obtained. According to the aforementioned selection rule, only a small part of elements that account for the total number need to be relocated, which denotes that the number of individual parameters waiting for optimizing by DE is decreased considerably, and thereby greatly accelerates the convergence speed of the algorithm. A set of synthesis experiments for PSA ranging from small to moderate size are presented to validate the effectiveness of the proposed method.

1. INTRODUCTION

Pattern synthesis for sparse array is attractive for the advantages in reducing power consumption, antenna weight, and cost [1]. However, it is not an easy issue to be dealt with due to the dilemma that the synthesis for a sparse array always refers to multiple constraints including the aperture size, total number of elements, minimum spacings between adjacent elements, etc.

Over the past few decades, a number of tools have been proposed to address the issue. The earlier synthesis approaches mainly include dynamic programming [2], technique of iterative least squares [3], simplex search method [4], mixed linear programming [5], etc. Most of these traditional optimization methodologies can only perform single-point search, and thereby they may be trapped in local optima when dealing with more complicated, strong nonlinear synthesis cases. Therefore, evolutionary algorithms, such as simulated anealing [6, 7], real genetic algorithm (RGA) [8, 9], approach of differential evolution (DE) [10–12], and particle swarm optimization (PSO) [13, 14], have been extensively exploited. Recently, some deterministic approaches are proposed for the synthesis of both linear isophoric sparse arrays and concentric ring isophoric sparse arrays [15–20]. These methods are started from identifying a reference continuous aperture source (RCAS) fulfilling at best the radiation requirements, and then deriving the array layout as a discretization of the RCAS [20] by recurring to the density taper technique. As a consequence, the synthesis of sparse isophoric arrays is achieved by some emulation of the continuous aperture source at a low computational burden. Other notable contribution in sparse array synthesis is the way through employing compressive sampling (CS) technique [21–25].

Received 6 August 2019, Accepted 6 November 2019, Scheduled 21 November 2019

* Corresponding author: Xin-Kuan Wang (wxkuan@snut.edu.cn).

[1] School of Physics and Optoelectronic Engineering, Xidian University, Xi'an, China. [2] School of Physical and Telecommunication Engineering, Shaanxi University of Technology, Hanzhong, China.

However, the CS-based approaches require that the element is un-uniformly excited, and the element phase is also taken into consideration.

In this paper, by recurring to the procedure of stochastic global optimization, we try to exploit the method for synthesis of planar sparse array (PSA) with apertures ranging from small to moderate size. As shown in the aforementioned references [6–14], the stochastic algorithms have achieved good performance in synthesis of both linear and some planar sparse arrays (PSAs). However, this kind of method is only limited to a linear or small size planar array. According to our knowledge, very limited studies [8, 9] focus on the synthesis of moderate ('moderate' denotes the array with the number of elements greater than 50, but less than 200) planar sparse array (MPSA). Although the reports using RGA in [8] refer to the synthesis of MPSA including 100 and 108 elements, the actual numbers of elements needed for optimization are respectively equal to 25 and 27 due to the quadrant symmetry of the array. Despite that, a single trial of the RGA requires about two hours through a PC equipped with an Intel Celeron processor [8]. Therefore, it is no doubt that the stochastic algorithm will face heavy calculation burden in synthesizing MPSA.

Inspired by the efficiency of the iterative Fourier transform (IFT) [26–28] in synthesizing both linear and planar thinned arrays (PTAs), a hybrid method based on the IFT and the algorithm of DE, called IFT-DE, was proposed in [29] for the synthesis of a uniformly amplitude linear sparse array (LSA) including many elements. In the first step of the method, the aperture of LSA is divided into a set of half wavelength spaced lattices. Then, a linear thinned array (LTA) with reduced sidelobe level is obtained through performing the IFT, and the filling factor of the LTA is determined by the total number of elements within the LSA. In the second step, for the same thinned array, the elements spaced equal to or greater than a wavelength are selected as the candidates whose locations will be optimized by the DE procedure. Thanks to the high calculation speed of the FFT. The use of IFT helps to efficiently find an LTA with improved sidelobe performance. Therefore, the CPU-time is mainly occupied by the DE. Not only that, due to the specified selection rule, the number of elements needed for optimization is considerably reduced. As a result, the LSA could be synthesized at relative low computational burden.

In this paper, the IFT-DE is further extended to the synthesis of a uniform amplitude PSA with apertures ranging from small to moderate size. There are mainly two differences between the method in this paper and the algorithm proposed in [29]. Firstly, the 2D-FFT is performed over 2D lattices to get a planar thinned array (PTA). Secondly, unlike the strategy of DE/rand/1 used in [29] to generate the mutant vectors, the strategy of DE/local-to-best/1 is adopted in this paper.

The remaining part of the paper is organized as follows. Section 2 describes the procedures of the IFT-DE. Section 3 gives several numerical experiments to validate the effectiveness of this method. Finally, Section 4 concludes the whole paper.

2. THE PROCEDURES OF THE PROPOSED METHOD

Supposing a rectangular planar array including $M \times N$ equally spaced isotropic elements without consideration of phase, the array factor can be described as follows [28]

$$F(u,v) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} A_{mn} e^{jkd(mu+nv)} \tag{1}$$

where k represents the wavenumber and d the element spacing equal to $\lambda/2$ (λ denotes the wavelength). $u = \sin(\theta)\cos(\phi)$, $v = \sin(\theta)\sin(\phi)$, θ and ϕ respectively indicate the elevation and azimuth angle. A_{mn} is the element excitation located in (m, n). We can see from Eq. (1) that the collection of element excitations $\{A_{mn}\}$ is related to the array factor $F(u, v)$ through a 2D Fourier transform; therefore, the 2D-FFT can be used to speed up the calculation. Based on Equation (1), the detailed procedures of the IFT-DE are listed as below.

Step 1: Divide the given aperture into a set of $\lambda/2$-spaced square lattices. Then the elements are forced to locate at the lattices through performing the IFT, and thereby a planar thinned array with reduced sidelobe level is formed.

Suppose an array with aperture size of $a\lambda \times b\lambda$, and the aperture can be divided into $M \times N\lambda/2$-

spaced lattices. The value of M and N is relation to a and b through the equation

$$\begin{cases} M = floor(2a+1) \\ N = floor(2b+1) \end{cases} \tag{2}$$

where $floor(A)$ denotes the MATLAB function that rounds the value of A to the nearest integer less than or equal to A. Assume that a total of N_0 elements are contained in the target PSA; therefore, a rectangular PTA with N_0 elements is obtained firstly by exploiting the IFT. The procedures of IFT are described as follows [28].

1) Randomly initialize the element excitations equal to one or zero with the same probability.
2) Compute $F(u,v)$ from $\{A_{mn}\}$ through a $K \times K$ points 2D-IFFT, where the value of K should be greater than $4 \cdot \max(M,N)$ to meet the sampling theorem.
3) Adjust the values of $F(u,v)$ to match the prescribed sidelobe level (PSL). In more detail, select the sampling points in the sidelobe region of $F(u,v)$, and force their value beneath the level of PSL (PSL is determined by trial and error).
4) Compute $\{A_{mn}\}$ from $F(u,v)$ by a $K \times K$ points 2D-FFT.
5) Truncate $\{A_{mn}\}$ from $K \times K$ samples to $M \times N$ samples that coincide with the columns and rows of the rectangular lattices.
6) Force the N_0 samples of element excitations with higher amplitudes equal to one while the rest equal to zero.
7) Repeat 2) to 6) until the same element distribution between two adjacent iterations is obtained, or the number of iterations arrives at its maxima.

Step 2: DE optimizing [29–31]

1) Select the elements waiting for optimizing.

For each column of the just obtained PTA, the elements spaced equal to or greater than λ are selected as the candidates whose locations need to be optimized by the DE procedure. We specify that the selected elements of PTA are numbered sequentially within the aperture from top to bottom and left to right. A 6×6 rectangular PTA including 22 elements is described in Fig. 1 to illustrate the selection rule, where the symbols of cross represent empty lattices, and the symbols of both hollow and solid circles indicate the presence of elements located on the lattices. Particularly, the solid circles denote the element candidates coincide with the selection rule (the same as follows). We can see from Fig. 1 that a total of 8 elements (numbered from 1st to 8th), respectively located at the positions (x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5), (x_6, y_6), (x_7, y_7), (x_8, y_8), are selected among the 22 candidates. In other words, only 8 element locations would be optimized in the next procedure.

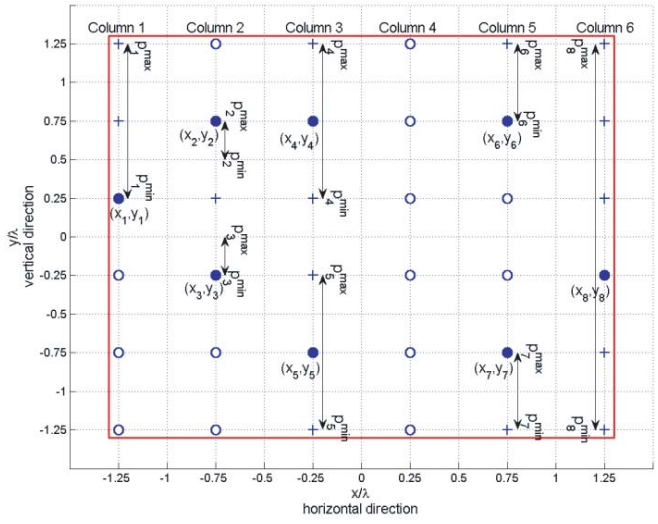

Figure 1. A 6×6 rectangular PTA including 22 elements to illustrate the selection rule.

2) Determine the upper and lower bounds for each selected element that may move.

Assume that the current population of DE can be written as

$$P_g = \{Y_{q,g} | q = 1, 2, ..., Q\} \tag{3}$$

Subscript g indicates the generation number with $g_{\max}$ the maxima, and Q is the population size. Vector $Y_{q,g}$ describes the qth individual of the gth generation, which is constituted by the locations of selected elements waiting for optimizing, and can be recorded as

$$Y_{q,g} = \{p_{l,q,g} | l = 1, 2, ..., L\} \tag{4}$$

where parameter $p_{l,q,g}$ indicates the location of the lth element, and L denotes the total number. The value of $p_{l,q,g}$ is set to be bounded by the inequality

$$p_l^{\min} \le p_{l,q,g} \le p_l^{\max} \quad (l = 1, 2, ..., L,\ q = 1, 2, ..., Q) \tag{5}$$

$p_l^{\min}$ and $p_l^{\max}$ represent the lower and upper bounds of $p_{l,q,g}$, respectively. We specify the constraint that the minimum element spacing after DE optimizing is not less than $\lambda/2$, thus the values of $p_l^{\min}$ and $p_l^{\max}$ can be determined according to the initial element locations under three different cases.

Case 1: when the lattices between two adjacent elements are empty, the elements can move toward each other, and the maximum distances of the two elements moving to the counterpart are set equal. Not only that, when the two elements respectively reach their maximum ranges, the spacing between them is equal to $\lambda/2$. Taking the elements numbered in the 4th and 5th in Fig. 1, column 3 as an example, there are two empty lattices existing between them. Therefore, the values of y_4, y_5, $p_4^{\min}$, and $p_5^{\max}$ should conform to the following equation

$$\begin{cases} y_4 - p_4^{\min} = p_5^{\max} - y_5 \\ p_4^{\min} - p_5^{\max} = 0.5\lambda \end{cases} \tag{6}$$

Similarly, the elements numbered in the 2nd and 3rd in column 2 also coincide with the case, and thereby the location ranges that they move to each other should meet the equation

$$\begin{cases} y_2 - p_2^{\min} = p_3^{\max} - y_3 \\ p_2^{\min} - p_3^{\max} = 0.5\lambda \end{cases} \tag{7}$$

Differently, since there are no empty lattices above the 2nd element and below the 3rd element, the two selected elements cannot move to the directions, thus we have the equation

$$y_2 = p_2^{\max}, \quad y_3 = p_3^{\min} \tag{8}$$

Case 2: when the lattices between two adjacent elements are filled, the elements cannot move toward each other and can only move in the opposite direction. The elements numbered in the 6th and 7th in Fig. 1 column 5 describe this case.

Case 3: when there are only empty lattices between the selected element and the column end, to which we specify that the maximum range of the element could move. The 1st element and 4th to 8th elements in Figure 1 illustrate the case.

More clearly, the location ranges of the selected elements can also be described by the double arrows, as described in Fig. 1.

3) Initialization: Initialize the individual parameters to form the population through the equation

$$\begin{cases} p_{l,1,g} = y_l \\ p_{l,q,g} = p_l^{\min} + \text{rand} \cdot \left(p_l^{\max} - p_l^{\min}\right) \end{cases} \quad (l = 1, 2, ..., L,\ q = 2, ..., Q) \tag{9}$$

where y_l denotes the longitudinal coordinate of the initial position for the lth element located at (x_l, y_l), and rand is a uniformly distributed random number within (0, 1).

4) Mutation: Use the strategy of DE/local-to-best/1 to generate a population of Q mutant vectors, and the vector can be written as [30]

$$V_{q,g} = Y_{q,g} + F \cdot (Y_{\text{best}} - Y_{q,g}) + F \cdot (Y_{r_1,g} - Y_{r_2,g}), \quad q \in [1, Q] \tag{10}$$

Indices r_1 and r_2 indicate the subscripts of individual vectors that are randomly chosen from current population; Y_{best} is the best individual till now; F denotes the scaling factor.

5) Crossover: Binomial crossover is employed to produce the trial vector.

Assume that the trial vector is recorded as

$$W_{q,g} = \{w_{l,q,g} | l = 1, 2, ..., L\} \tag{11}$$

Then, parameter $w_{l,q,g}$ is obtained by binomial crossover as below [30]

$$w_{l,q,g} = \begin{cases} v_{l,q,g} & \text{if } R_l \leq C_r \\ p_{l,q,g} & \text{otherwise} \end{cases} \quad l = 1, 2, ..., L \tag{12}$$

where $v_{l,q,g}$ denotes the lth parameter of $V_{q,g}$, and R_l is the lth random number that is uniformly distributed in (0, 1). C_r represents the crossover probability, which controls the fraction of parameter values inherited from the mutant vector.

6) Selection: Specify the fitness function as the equation

$$F_{\text{fit}} = h(Y) \tag{13}$$

where Y refers to the individual vector, and F_{fit} is the fitness value that actually represents the maximum sidelobe level (MSLL) of the sparse array. Based on Equation (13), the target vector in the next generation is thereby given by

$$Y_{q,g+1} = \begin{cases} W_{q,g} & \text{if } h(W_{q,g}) < h(Y_{q,g}) \\ Y_{q,g} & \text{otherwise} \end{cases} \tag{14}$$

Through Equation (14) we can see that if the trial vector $W_{q,g}$ has lower fitness value than that of the individual vector $Y_{q,g}$, it replaces $Y_{q,g}$ in the next generation, or else $Y_{q,g}$ is retained.

When the new population is produced, the elite strategy is employed to retain the optimum individual. Then, the above process of mutation, crossover, and selection is repeated until it reaches the pre-specified value of $g_{\max}$. Furthermore, once the overall steps of the method are finished, the current trial is terminated, and the method will proceed to the next numerical trial.

3. NUMERICAL EXPERIMENTS

To demonstrate the performance of the proposed method, the IFT-DE is exploited to the synthesis of some PSAs with the elements respectively distributed within the rectangular and circular aperture. The sampling points of IFT are 1024×1024, and the parameters involved in DE, including the population size, crossover probability, as well as scaling factor, are equal to 20, 0.9, and 0.7, respectively. The number of numerical trials is 50 for different instances.

3.1. Planar Sparse Array with Rectangular Aperture

Suppose that 39 elements need to be filled in a PSA with the aperture size of $3.5\lambda \times 3.5\lambda$, thus the aperture can be divided into 8×8 rectangular lattices according to Equation (2). After 50 trials of the IFT-DE, the 3D-pattern of the best PSA has the MSLL of -18.65 dB (as shown in Fig. 2 trial #41), almost 3 dB sidelobe reduction compared to the value of corresponding thinned array obtained by the IFT, the first step of this method. The u-cut as well as the v-cut far field pattern of the PSA is described in Fig. 3. Furthermore, Fig. 4 and Fig. 5 compare the element layouts of the PTA and the PSA obtained at the same trial. Fig. 4 shows that a total of 11 elements have been selected as the candidates for DE optimization, while Fig. 5 indicates that the final locations of selected elements are no longer confined in the lattices, but are randomly moved a distance along the column direction. However, the spacing between adjacent elements is still not less than half wavelength.

Another instance refers to a PSA including 70 elements with aperture size of $5.5\lambda \times 5.5\lambda$, where the aperture can be divided into 12×12 rectangular lattices, and when all the elements are assigned into the lattices through performing the IFT, the filling factor of the PTA is about 48%. The best PSA after 50 trials of the IFT-DE, as shown in Fig. 6, has the MSLL of -20.91 dB. The performance of PSA shows a sidelobe suppression of 3.65 dB compared to the value obtained by the IFT and 1.51 dB compared to that obtained by the M-cGA [32]. Fig. 7 gives the 3D far field pattern.

Both the results mentioned above and the MSLL performance of some other instances with the aperture size ranging from $2.5\lambda \times 2.5\lambda$ to $5.5\lambda \times 5.5\lambda$ are listed in Table 1, where T represents the

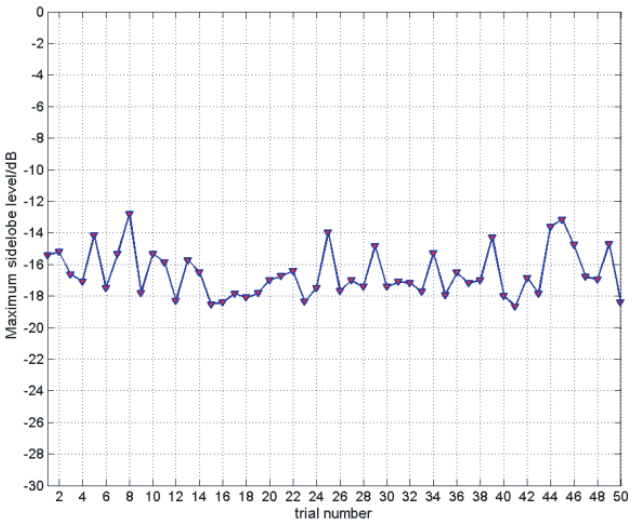

Figure 2. Convergence curve of the maximum sidelobe performance versus the trial number.

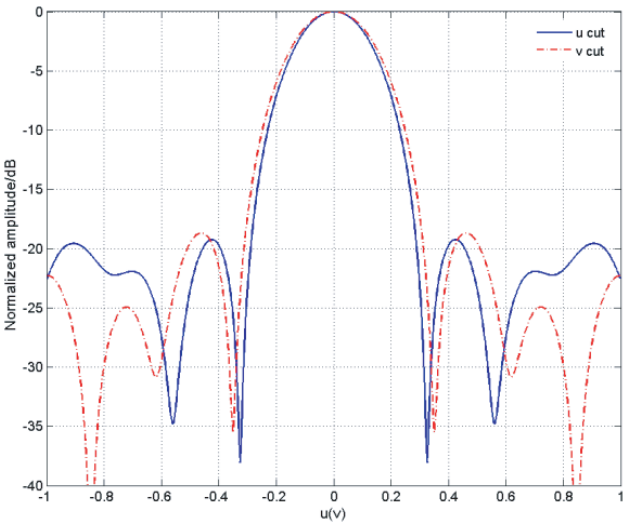

Figure 3. U-cut and v-cut of the far field pattern of the PSA (maximum aperture size: $3.5\lambda \times 3.5\lambda$).

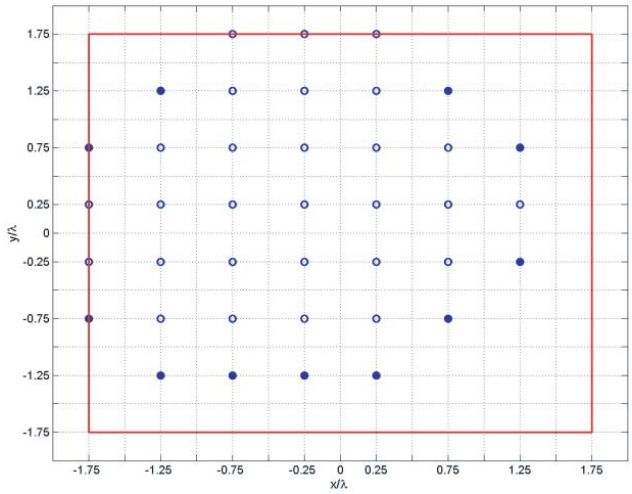

Figure 4. The element layout of the PTA obtained at trial #41 (maximum aperture size: $3.5\lambda \times 3.5\lambda$).

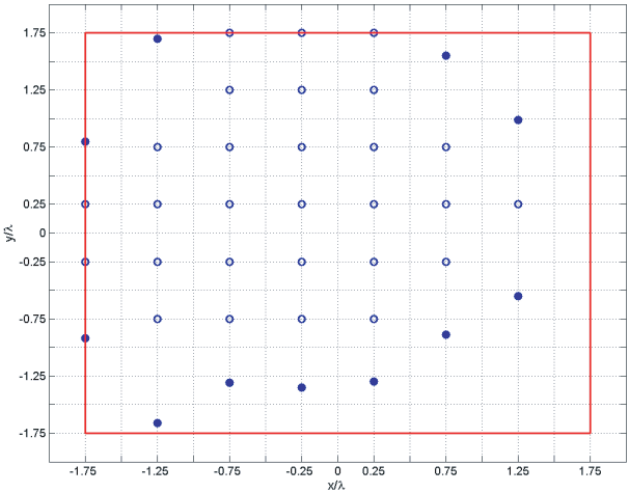

Figure 5. The element layout of the PSA obtained at trial #41 (maximum aperture size: $3.5\lambda \times 3.5\lambda$).

number of elements, and T_0 denotes the number of selected elements waiting for optimization. The synthesis results in Table 1 indicate that through performing the IFT-DE, the obtained PSAs show at least 1.4 dB, at best 3.79 dB sidelobe suppression compared to the PTA counterpart obtained by the IFT. Not only that, the value of T_0 accounts for less than $0.5T$ in most instances, which indicates that the obtained PSA is in fact a quasi-PTA. Furthermore, the u-cut pattern beamwidth (Δu) of both PTAs and PSAs are given in Table 1. Comparing the results we can see that although impressive sidelobe suppression can be arrived through relocating a small part of elements along the array column direction, it seems that just a little tradeoff at the beam broadening is obtained for the PSAs with aperture size ranging from $3.5\lambda \times 3.5\lambda$ to $5.5\lambda \times 5.5\lambda$. The reason can be partly explained by the expansion of the effective aperture of the array, as we compare the PSA to its PTA counterpart, respectively depicted in Fig. 5 and Fig. 4. Since there are 11 elements moved to the aperture boundary in different degrees along column direction, the effective aperture of PSA (Fig. 5) is indeed enlarged compared to that of

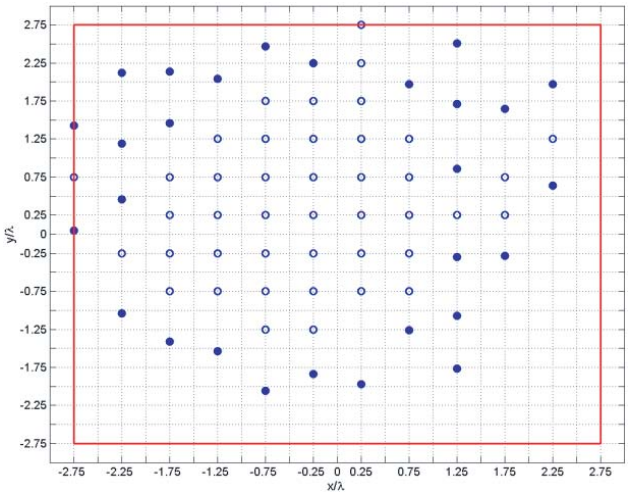

Figure 6. The element layout of the PSA (maximum aperture size: $5.5\lambda \times 5.5\lambda$).

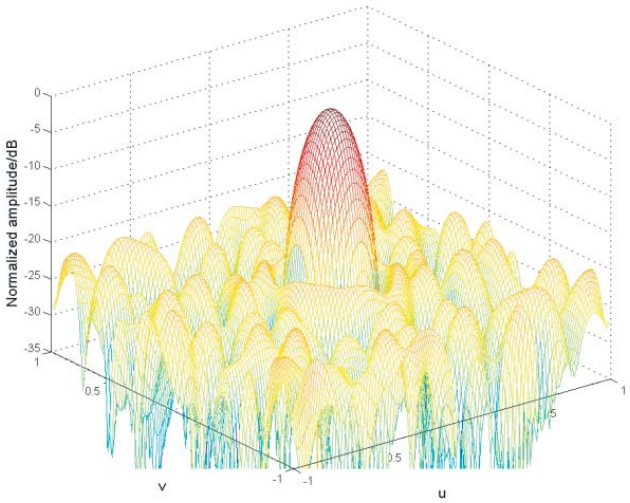

Figure 7. The 3D far field pattern of the PSA described in Fig. 6.

Table 1. Comparison of the synthesis performance between the rectangular PTA obtained by the IFT and the PSA obtained by the IFT-DE.

Aperture size	No. of lattices ($M \times N$)	T	The performance of PTA obtained by the IFT, the first step of the IFT-DE		T_0	The performance of PSA obtained by the IFT-DE	
			MSLL (dB)	Δu		MSLL (dB)	Δu
$2.5\lambda \times 2.5\lambda$	6×6	15	-12.31	0.418	6	-16.10	0.3828
		18	-13.78	0.5468	12	-16.65	0.4882
		22	-16.0	0.4492	11	-17.44	0.4102
$3.5\lambda \times 3.5\lambda$	8×8	26	-15.79	0.336	12	-17.93	0.3594
		32	-15.69	0.3632	16	-18.60	0.4062
		39	-15.69	0.2696	11	-18.65	0.293
$4.5\lambda \times 4.5\lambda$	10×10	40	-15.27	0.2656	16	-18.78	0.2774
		50	-16.87	0.2382	20	-19.15	0.246
		60	-17.44	0.2188	23	-19.50	0.2226
$5.5\lambda \times 5.5\lambda$	12×12	58	-17.93	0.2148	24	-19.39	0.2188
		70	-17.26 -19.4 by M-cGA [32]	0.211	28	-20.91	0.2266
		87	-18.04	0.1758	40	-21.21	0.1796

PTA (Fig. 4) counterpart. Therefore, although the MSLL performance of PSA is considerably improved compared to that of PTA, the beamwidth of u-cut pattern shows little broadening. Particularly, for the PSA with aperture size of $2.5\lambda \times 2.5\lambda$, the u-cut pattern is narrowed to some extent despite the reduced MSLL. The reason, however, may be explained by the fact that the contribution of aperture enlargement to the beamwidth exceeds the influence of the sidelobe reduction on beam broadening.

3.2. Planar Sparse Array with Circular Aperture

In order to compare with the published report in [9], a circular PSA with diameter of 10λ is considered, and the number of elements is equal to 201. According to the proposed method, there should be 305

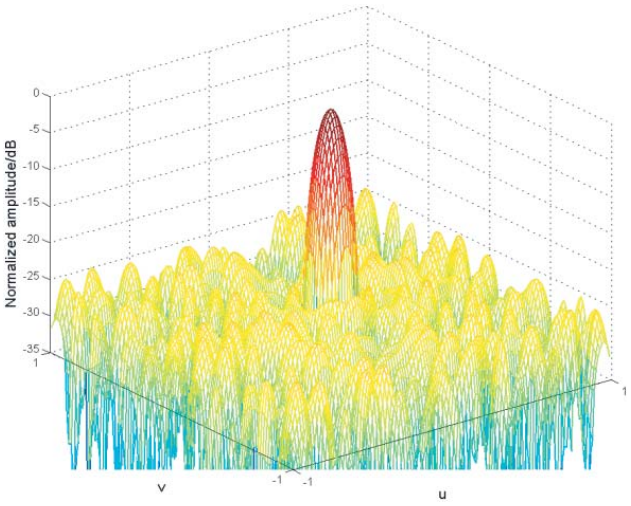

Figure 8. The 3D far field pattern of the circular PSA with diameter of 10λ.

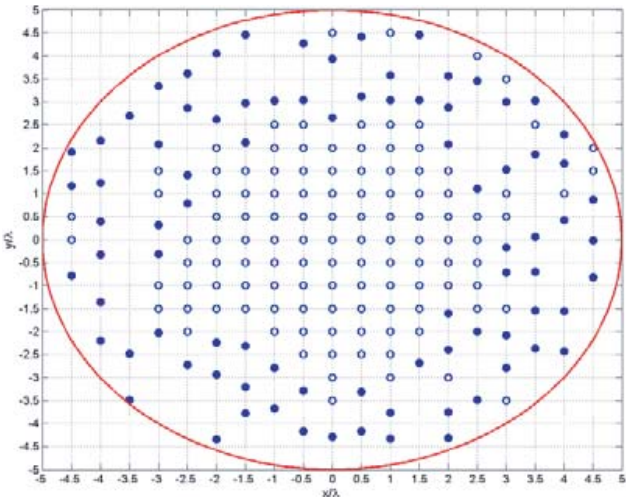

Figure 9. The element layout of the circular PSA (maximum diameter: 10λ).

lattices that are half wavelength spaced within the circular aperture. The best PSA after 50 trials of the IFT-DE has the MSLL of -24.2101 dB, which indicates about 1.45 dB sidelobe reduction compared to the PTA counterpart and 0.46 dB sidelobe reduction compared to the circular PSA reported in [9]. The 3D far field pattern is described in Fig. 8. Furthermore, Fig. 9 gives the element layout of the circular PSA, where we can see that 85 elements have been relocated among the total of 201 candidates. Therefore, the obtained circular PSA can still be deemed as a quasi-PTA.

All the numerical experiments mentioned above are conducted in a Windows 7 environment by using a PC equipped with an 8 GB RAM and the Intel I7-6700 processor that operates at 3.4 GHz. The computational cost for different PSAs ranges from 0.5 hours to 20 hours.

4. CONCLUSION

The synthesis of uniform amplitude moderate PSA always involves multiple constraints on many elements, thus the use of stochastic optimization algorithms would face huge computational burden. This method may ease the difficulty by the exploitation of the IFT. In more detail, the aperture of PSA is divided into a set of half wavelength spaced lattices firstly. Then the elements are assigned to locate at the lattices through performing the IFT, and thereby a PTA with improved sidelobe performance is obtained. Secondly, the elements of PTA spaced equal to or greater than a wavelength are selected as the candidates whose locations are optimized by the DE procedure. Consequently, since only a small part of elements that account for the total are optimized, it makes the convergence pressure of DE substantially alleviated. The numerical experiments indicate that the proposed method can be effectively applied to the synthesis of uniform amplitude PSAs with the apertures ranging from small to moderate size. However, the method shows no impressive sidelobe performance when dealing with large PSAs. To illustrate the case, we take a rectangular PSA with aperture size equal to $14.5\lambda \times 14.5\lambda$ that includes 360 elements as an example. According to the proposed method, the aperture can be divided into 30×30 lattices. Through performing the IFT-DE, the best PSA among 50 trials only shows 0.68 dB sidelobe improvement compared to its PTA counterpart. The reason can be explained as that with the enlarging of PSA aperture size, the number of elements needed for optimization (a total of 181 elements are optimized in this case) will be further increased accordingly. As a result, the solution space of the algorithm is considerably expanded, which makes the method apt to converge in local minima and cannot jump out of it for the potential huge computational burden.

ACKNOWLEDGMENT

This work are supported by the National Natural Science Foundation of China under grant No. 61601272 and No. 61772398, in part by the 111 Project (B17035), and the Key Research and Development Program Projects of Shaanxi Province under grant No. 2019SF-257, in part by the scientific research plan of Shaanxi Provincial Education Department under grant No. 18JK0144, and the scientific research plan of Shaanxi University of Technology under grant No. SLGQD14-05.

REFERENCES

1. Mailloux, R. J., *Phased Array Antenna Handbook*, 2nd Edition, Artech House, Norwood, MA, Boston, 2005.
2. Skolnik, M., G. Nemhauser, and J. Sherman, "Dynamic programming applied to unequally spaced elements," *IEEE Trans. Antennas Propag.*, Vol. 12, No. 1, 35–43, 1964.
3. Redlich, R. W., "Iterative least-squares synthesis of nonuniformly spaced linear arrays," *IEEE Trans. Antennas Propag.*, Vol. 21, No. 1, 106–108, 1973.
4. Leahy, R. M. and B. D. Jeffs, "On the design of maximally sparse beamforming arrays," *IEEE Trans. Antennas Propag.*, Vol. 39, No. 8, 1178–1187, 1991.
5. Holm, S., B. Elgetun, and G. Dahl, "Properties of the beam pattern of weight- and layout-optimized sparse arrays," *IEEE Trans. Ultrason. Ferroelectr. Freq. Control*, Vol. 44, No. 5, 983–991, 1997.
6. Murino, V., A. Trucco, and C. S. Regazzoni, "Synthesis of unequally spaced arrays by simulated annealing," *IEEE Transactions on Signal Processing*, Vol. 44, No. 1, 119–123, 1996.
7. Trucco, A. and V. Murino, "Stochastic optimization of linear sparse arrays," *IEEE J. Ocean. Eng.*, Vol. 24, No. 3, 291–299, 1999.
8. Chen, K., X. Yun, Z. He, et al., "Synthesis of sparse planar arrays using modified real genetic algorithm," *IEEE Trans. Antennas Propag.*, Vol. 55, No. 4, 1067–1073, 2007.
9. Chen, K., H. Chen, L. Wang, et al., "Modified real GA for the synthesis of sparse planar circular arrays," *IEEE Antennas and Wireless Propagation Letters*, Vol. 15, 274–277, 2016.
10. Kurup, D. G., M. Himdi, and A. Rydberg, "Synthesis of uniform amplitude unequally spaced antenna arrays using the differential evolution algorithm," *IEEE Trans. Antennas Propag.*, Vol. 51, No. 9, 2210–2217, 2003.
11. Lin, C., A. Qing, and Q. Feng, "Synthesis of unequally spaced antenna arrays by using differential evolution," *IEEE Trans. Antennas Propag.*, Vol. 58, No. 8, 2553–2561, 2010.
12. Goudos, S. K., K. Siakavara, T. Samaras, et al., "Sparse linear array synthesis with multiple constraints using differential evolution with strategy adaptation," *IEEE Antennas and Wireless Propagation Letters*, Vol. 10, 670–673, 2011.
13. Liu, D., Q. Feng, W.-B. Wang, and X. Yu, "Synthesis of unequally spaced antenna arrays by using inheritance learning particle swarm optimization," *Progress In Electromagnetics Research*, Vol. 118, 205–221, 2011.
14. Donelli, M., A. Martini, and A. Massa, "A hybrid approach based on PSO and hadamard difference sets for the synthesis of square thinned arrays," *IEEE Trans. Antennas Propag.*, Vol. 57, No. 8, 2491–2495, 2009.
15. Bucci, O. M., M. D'Urso, T. Isernia, et al., "Deterministic synthesis of uniform amplitude sparse arrays via new density taper techniques," *IEEE Trans. Antennas Propag.*, Vol. 58, No. 6, 1949–1958, 2010.
16. Bucci, O. M. and S. Perna, "A deterministic two dimensional density taper approach for fast design of uniform amplitude pencil beams arrays," *IEEE Trans. Antennas Propag.*, Vol. 59, No. 8, 2852–2861, 2011.
17. Bucci, O. M., T. Isernia, and A. F. Morabito, "An effective deterministic procedure for the synthesis of shaped beams by means of uniform-amplitude linear sparse arrays," *IEEE Trans. Antennas Propag.*, Vol. 61, No. 1, 169–175, 2013.

18. Morabito, A. F., T. Isernia, and L. Di Donato, "Optimal synthesis of phase-only reconfigurable linear sparse arrays having uniform-amplitude excitations," *Progress In Electromagnetics Research*, Vol. 124, 405–423, 2012.
19. Morabito, A. F., A. R. Laganà, and T. Isernia, "Isophoric array antennas with a low number of control points: A 'size tapered' solution," *Progress In Electromagnetics Research Letters*, Vol. 36, 121–131, 2013.
20. Morabito, A. F. and P. G. Nicolaci, "Optimal synthesis of shaped beams through concentric ring isophoric sparse arrays," *IEEE Antennas and Wireless Propagation Letters*, Vol. 16, 979–982, 2017.
21. Oliveri, G. and A. Massa, "Bayesian compressive sampling for pattern synthesis with maximally sparse non-uniform linear arrays," *IEEE Trans. Antennas Propag.*, Vol. 59, No. 2, 467–481, 2011.
22. Oliveri, G., M. Carlin, and A. Massa, "Complex-weight sparse linear array synthesis by bayesian compressive sampling," *IEEE Trans. Antennas Propag.*, Vol. 60, No. 5, 2309–2326, 2012.
23. Morabito, A. F. and P. Rocca, "Reducing the number of elements in phase-only reconfigurable arrays generating sum and difference patterns," *IEEE Antennas and Wireless Propagation Letters*, Vol. 14, 1338–1341, 2015.
24. Morabito, A. F., "Synthesis of maximum-efficiency beam arrays via convex programming and compressive sensing," *IEEE Antennas and Wireless Propagation Letters*, Vol. 16, 2404–2407, 2017.
25. Rocca, P., G. Oliveri, R. J. Mailloux, et al., "Unconventional phased array architectures and design methodologies — A review," *Proceedings of the IEEE*, Vol. 103, No. 3, 544–560, 2016.
26. Keizer, W. P. M. N., "Linear array thinning using iterative FFT techniques," *IEEE Trans. Antennas Propag.*, Vol. 56, No. 8, 2757–2760, 2008.
27. Keizer, W. P. M. N., "Low-sidelobe pattern synthesis using iterative Fourier techniques coded in MATLAB," *IEEE Antennas and Propagation Magazine*, Vol. 51, No. 2, 137–150, 2009.
28. Keizer, W. P. M. N., "Large planar array thinning using iterative FFT techniques," *IEEE Trans. Antennas Propag.*, Vol. 57, No. 10, 3359–3362, 2009.
29. Wang, X.-K. and G.-B. Wang, "A hybrid method based on the iterative fourier transform and the differential evolution for pattern synthesis of sparse linear arrays," *International Journal of Antennas and Propagation*, Vol. 2018, 1–7, 2018.
30. Price, K. V., R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*, Natural Computing Series Springer, 2005.
31. Rocca, P., G. Oliveri, and A. Massa, "Differential evolution as applied to electromagnetics," *IEEE Antennas and Propagation Magazine*, Vol. 53, No. 1, 38–49, 2011.
32. Ha, B. V., M. Mussetta, P. Pirinoli, et al., "Modified compact genetic algorithm for thinned array synthesis," *IEEE Antennas and Wireless Propagation Letters*, Vol. 15, 1105–1108, 2016.