

Antenna Array Pattern Synthesis via Coordinate Descent Method

Yuanhao Wang¹, Xiaoxi He¹, Jiangning Wang¹, Sergey Berezin², Wolfgang Mathis¹

¹Institut für Theoretische Elektrotechnik, Gottfried Wilhelm Leibniz Universität Hannover, Hannover, Germany

²Applied Mathematics Department, Saint-Petersburg State Polytechnical University, St. Petersburg, Russia

Email: ywang@tet.uni-hannover.de, servberezin@yandex.ru

Received 24 March 2015; accepted 16 May 2015; published 19 May 2015

Copyright © 2015 by authors and Scientific Research Publishing Inc.

This work is licensed under the Creative Commons Attribution-NonCommercial International License (CC BY-NC).

<http://creativecommons.org/licenses/by-nc/4.0/>



Open Access

Abstract

This paper presents an array pattern synthesis algorithm for arbitrary arrays based on coordinate descent method (CDM). With this algorithm, the complex element weights are found to minimize a weighted L_2 norm of the difference between desired and achieved pattern. Compared with traditional optimization techniques, CDM is easy to implement and efficient to reach the optimum solutions. Main advantage is the flexibility. CDM is suitable for linear and planar array with arbitrary array elements on arbitrary positions. With this method, we can configure arbitrary beam pattern, which gives it the ability to solve variety of beam forming problem, e.g. focused beam, shaped beam, nulls at arbitrary direction and with arbitrary beam width. CDM is applicable for phase-only and amplitude-only arrays as well, and furthermore, it is a suitable method to treat the problem of array with element failures.

Keywords

Antenna, Phased Array, Optimization, Pattern Synthesis, Coordinate Descent Method

1. Introduction

Comparing conventional antenna, antenna array provides advantages in flexibility and versatility. By changing the complex weights (including amplitude and phase) on each array element, the radiation pattern can be controlled and reconfigured. The antenna array pattern synthesis problem consists of finding weights that satisfy a set of specifications on the beam pattern [1]. Several most famous approaches can solve certain array pattern synthesis problems analytically, such as Schelkunoff method [2], the Dolph-Chebyshev synthesis [3], and the Taylor method [4]. Over the last several decades, many numerical techniques have been proposed. There are a few methods often used e.g. Convex Optimization, Genetic Algorithms (GA) [5], Differential Evolution Algo-

rithm (DE) [6], Particle Swarm Optimization (PSO) [7] and Taguchi's Method [8] [9].

As an advantage of convex optimization, once an array pattern synthesis problem is convex, the global optimum must exist [1]. The most difficult part is reforming of a nonconvex array pattern synthesis problem into a convex problem. So far many array pattern synthesis problems have already been solved with convex optimization. Focused beam pattern with fix nulls and upper bound constraints over all side lobes can be solved with convex optimization [1]. In [10], a shaped beam array pattern synthesis problem was solved with sequential convex optimizations. If the amplitude-response error is linearly approximatable, it can be solved with convex optimization as well [11]. Some array pattern synthesis problems e.g. radiation pattern with lower bound constraints or synthesis with phase-only array cannot be solved with convex optimization method [1].

While the usage of convex optimization is restricted to the class of problems that can be transformed to be convex, the metaheuristic algorithms, e.g. GA, DE and PSO, can provide much more flexibility. They are not restricted in terms of types of array pattern synthesis problem. Furthermore, these evolutionary algorithms are able to deal with a huge parameter set without getting trapped in the local minima [12]; thus such a stochastic process is better suitable for large size arrays. In [13] GA was used to optimize the large thinned arrays in order to obtain low SLL (Side Lobe Level). The GA has no need to calculate derivatives and search from many points instead of a single point [5]. Further articles [14]-[17] present GA used for array pattern synthesis with different fitness functions and different requirements of the pattern. Differential Evolution Algorithm (DE) belongs to evolutionary algorithm as well. Comparing with the GA, the DE has no need for coding of the parameters and used different cross-over strategies. DE uses very few control parameters, and is easy to implement with any sort of program language. Many modified versions of the standard DE have been published in [12] [18]-[20]. PSO is similar in some ways to GA and DE, but requires less calculation and generally fewer lines of code. An overview about PSO in electromagnetics has been offered in [21]. In [22] the results of GA and PSO solving array pattern synthesis problems are compared. Many variants of the PSO specialized for difference array pattern synthesis problems have been discussed in [23]-[25].

In this article, an array pattern synthesis algorithm for arbitrary arrays based on coordinate descent method (CDM) is presented. CDM has been first mentioned in [26] for solving smooth unconstrained minimization problems. As an efficient optimization algorithm, CDM is already used for many areas e.g. the autofocus system of SAR-Radar [27] [28], tomography [29] and digital image processing [30]. CDM minimizes a function with respect to a single parameter while holding the remaining parameters constant. Comparing with other direct search methods, the descent direction of CDM is predefined once the coordinate system is given. This property is meaningful for array pattern synthesis problems, because every complex amplitude of array element can be seen as a predefined search direction (or one dimension) of a N -dimension space, which describes the domain of a array pattern function.

This paper is an illustration of the utility of CDM for array pattern synthesis problems. In Section 2, a briefly analytical description of CDM is presented, along with the formulation of the array pattern synthesis problems and corresponded fitness function. It is shown in Section 2.4, how the applications of adaptive weights function accelerate the optimization process and help achieving better results. Example of the applications of the CDM in different array pattern synthesis problems with comparisons to known algorithms is shown in Section 3. Then there is the conclusion in Section 4.

2. Problem Formulation and Resolution

2.1. Coordinate Descent Method

Coordinate descent methods were among the first optimization schemes suggested for solving smooth unconstrained minimization problems. The main advantage of these methods is the simplicity of each iteration, both in generating the search direction and in performing the update of variables [31]. A coordinate descent method can be described as follow:

We have $\mathbf{a} = (a_1, a_2, \dots, a_N) \in \mathbb{C}^N$ and $f : \mathbb{C}^N \rightarrow \mathbb{R}$.

The optimization problem for $f(\mathbf{a})$ is

$$\hat{\mathbf{a}} = \arg \min_{\mathbf{a} \in \mathbb{C}^N} f(\mathbf{a}). \quad (1)$$

As the optimization in (1) has no closed-form solution, we resort to an iterative method based on coordinate

descent. In coordinate descent optimization, each parameter is optimized in turn, while holding all other parameters fixed. Because the parameters are interrelated, optimization over the entire set of parameters must be performed a number iterations. We start with arbitrary $\mathbf{a}^{(0)} = (a_1^{(0)}, \dots, a_N^{(0)}) \in \text{dom}f$ as initial value. Let $\hat{a}_n^{(i)}$ denote the optimum value of n th coordinate (the n th element) of the i th iteration; then, the coordinate descent estimate at the next iteration $i+1$ is defined as

$$\forall i \in [1, \infty) \subset \mathbb{N} : \hat{a}_n^{(i+1)} = \arg \min_{a \in \mathbb{C}} f(\hat{a}_1^{(i+1)}, \dots, \hat{a}_{n-1}^{(i+1)}, a, \hat{a}_{n+1}^{(i)}, \dots, \hat{a}_N^{(i)}). \quad (2)$$

Now compute the $f^{(i+1)} = f(\hat{a}_1^{(i+1)}, \dots, \hat{a}_N^{(i+1)})$ and compare it with $f^{(i)} = f(\hat{a}_1^{(i)}, \dots, \hat{a}_N^{(i)})$. While $\left\| \frac{f^{(i+1)} - f^{(i)}}{f^{(i+1)}} \right\| > \epsilon$, which is a predefined value as quality criterion, repeat the procedure until $\left\| \frac{f^{(i+1)} - f^{(i)}}{f^{(i+1)}} \right\| \leq \epsilon$.

2.2. The Linear Array Pattern

For the sake of clarity, our array pattern synthesis problem is described for a linear array. Consider an antenna array composed of N isotropic elements placed at arbitrary and known locations $x_1, \dots, x_N \in R$. The power pattern in far field is given by

$$G(\mathbf{a}, \theta) = \left\| \sum_{n=1}^N a_n e^{jkx_n \cos(\theta)} \right\|^2, \quad (3)$$

where $a_n \in \mathbb{C}$ is the complex amplitude at the n th element with $a_n = w_n e^{j\phi_n}$, where w_n is the real amplitude and ϕ_n is the phase at n th element. $k = \frac{2\pi}{\lambda}$ is the wave number for a wavelength λ and θ denotes the polar angle.

2.3. The Fitness Function

The goal of array pattern synthesis problem is to find optimal parameters including array weights so that the designed beam pattern satisfies a set of specifications [1]. In this paper we reach the desired beam pattern via minimization of a fitness function f , whose similar form has been introduced in [32].

For a power pattern we define $f : \mathbb{C}^N \rightarrow [0, \infty) \subset \mathbb{R}$

$$f(\mathbf{a}^{(i)}) := \int_0^\pi \left(G(\mathbf{a}^{(i)}, \theta) - S(\theta) \right)^2 d\theta. \quad (4)$$

$G(\mathbf{a}^{(i)}, \theta)$ converges to $S(\theta)$ in the senses of $f \approx 0$ by i passes the entire given interval. Now we rewrite a_n into its complex form $a_n = w_n e^{j\phi_n}$. According to (3) the fitness function (4) can be reformed to

$$f(\mathbf{w}, \boldsymbol{\phi}) = \int_0^\pi \left(\left\| \sum_{n=1}^N w_n e^{j(C_n(\theta) + \phi_n)} \right\|^2 - S(\theta) \right)^2 d\theta \quad (5)$$

$$\text{with } C_n(\theta) = kx_n \cos(\theta), \mathbf{w} \in \mathbb{R}^N, \boldsymbol{\phi} \in \mathbb{R}^N.$$

For the n -th element the fitness function is given by

$$f(w_n, \phi_n) = \int_0^\pi \left(\left\| w_n e^{j(C_n(\theta) + \phi_n)} + R(\theta) \right\|^2 - S(\theta) \right)^2 d\theta = \int_0^\pi (\xi)^2 d\theta \quad (6)$$

with

$$\xi = w_n^2 + w_n R(\theta) e^{-j(C_n(\theta) + \phi_n)} + w_n R(\theta)^* e^{j(C_n(\theta) + \phi_n)} + R(\theta) R(\theta)^* - S(\theta) \quad (7)$$

where $R(\theta)$ denotes sum of the remaining terms in $\sum_{n=1}^N w_n e^{j(C_n(\theta)+\phi_n)}$, which are not depending on w_n and ϕ_n and $()^*$ denotes complex conjugate.

Due to following important characters of $G(w_n, \phi_n, \theta)$

$$G(\theta)|_{w_n, \phi_n} = G(\theta)|_{-w_n, \phi_n \pm \pi} \quad (8)$$

$$G(\theta)|_{w_n, \phi_n} = G(\theta)|_{w_n, \phi_n \pm 2\pi} \quad (9)$$

it is easy to prove

$$f(w_n, \phi_n) = f(-w_n, \phi_n \pm \pi) \quad (10)$$

$$f(w_n, \phi_n) = f(w_n, \phi_n \pm 2\pi) \quad (11)$$

as shown in **Figure 1**. That means we can use a surjection

$$p: A \rightarrow B \quad (12)$$

with

$$A =: \{(w_n, \phi_n) | w_n, \phi_n \in \mathbb{R}\} \quad (13)$$

$$B =: \{(w_n, \phi_n) | w_n \in [0, +\infty), \phi_n \in [0, 2\pi)\}, \quad (14)$$

which do not change $G(\theta)$ and $f(w_n, \phi_n)$, to cover the entire $f(w_n, \phi_n)$. Thus the new domain of $f(w_n, \phi_n)$ is $\mathbf{dom} f := \{w_n \in [0, +\infty), \phi_n \in [0, 2\pi)\}$. Because $f(w_n)$ and $f(\phi_n)$ are independent of each other and f is a quadratic function, there is only one $f_{\min}(w_n, \phi_n)$ in the new $\mathbf{dom} f$ and $\hat{\mathbf{a}} = \mathbf{argmin} f(w_n, \phi_n)$ can be solved numerically.

2.4. The Weight of Fitness Function

A crucial advantage of CDM is that a weight function $P(\theta)$ can be used, or more precisely, multiplied to the fitness function (4), in order to save the needed time and steps of the optimization process or achieve better results:

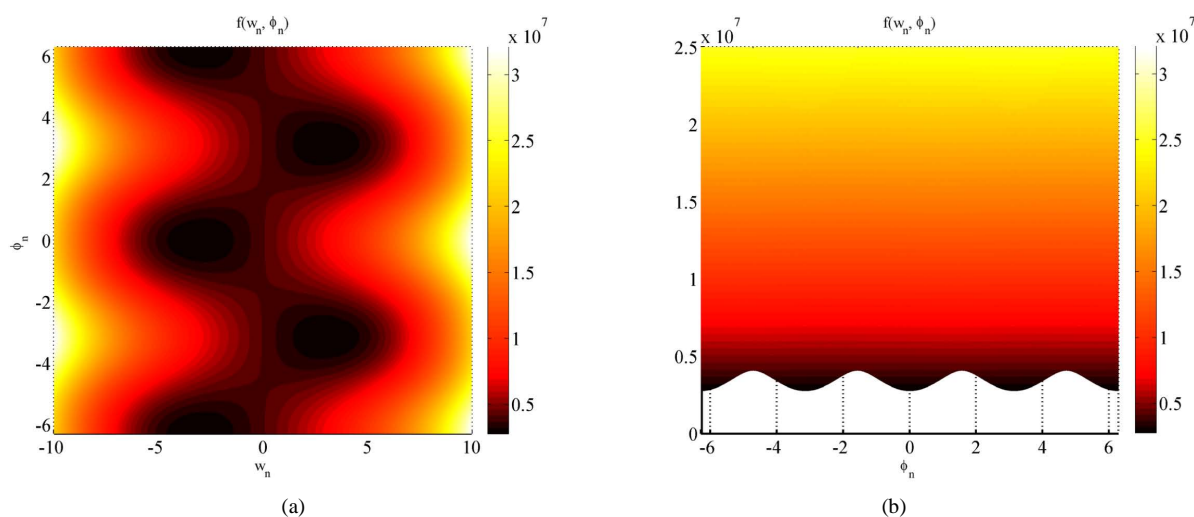


Figure 1. $f(w_n, \phi_n)$ by arbitrary iteration for a linear array (a) w_n, ϕ_n -view (horizontal); (b) $\phi_n, f(w_n, \phi_n)$ -view (vertical).

$$f(\mathbf{a}) := \int_0^\pi (G(\mathbf{a}, \theta) - S(\theta))^2 \cdot P(\theta) d\theta. \quad (15)$$

Because of the mathematical property of CDM, an arbitrary function $P(\theta) > 0$ can be multiplied to the fitness function, functioning as a weight function, without the need of changing any other part of the algorithm. Hence, an algorithm can be implemented to modify the fitness function during the optimization process by means of varying the weight function $P(\theta)$. With the application of different types of weight function $P(\theta)$ (or the algorithm to calculate the weight function $P(\theta)$), this array pattern synthesis algorithm can be exclusively used for different optimization problems, in order to acquire maximal speed and best results.

It should be emphasized that the effect of the usage of the weight function will be different according to the type of optimization problems. For problems which can obtain a perfect matching, e.g. Dolph-Chebyshev pattern (**Figure 4(a)**), the application of the weight function will only alter the speed of the optimization process, without changing the result, as the fitness function will always converge to zero. But when dealing with optimization problems which does not have a perfect matching, e.g. the problem of array with element failures (**Figure 8**), using the weight function can both change the speed of the process and the resulted beam pattern. This character of weight function can be quite useful when a improved result is required, but it also means that the weight function should be determined carefully when not expecting a huge change in the result.

A simple example is given here: when synthesizing the Dolph-Chebyshev pattern with CDM (shown in **Figure 4(a)**), the convergence is quite slow after the SLL started to descend. The reason can be considered as that at the side lobe area, the $G(\mathbf{a}, \theta)$ and $S(\theta)$ are both considerable small and thus their difference gives less feedback to the fitness function. Here weight functions like $P(\theta) = \frac{1}{G(\mathbf{a}, \theta)}$ or $P(\theta) = \frac{|G(\mathbf{a}, \theta) - S(\theta)|}{G(\mathbf{a}, \theta)}$ can

be employed, in order to enlarge the weights of the SLL in the fitness function. Thus the convergence is observably accelerated, which can be seen in **Figure 2(a)**, that the SLL descend much faster when using the weight function. It is shown in **Figure 2(b)**, how the SLL descend when matching a shaped beam using the same weight functions. In this example the optimization problem is from [33], which is a perfect matching optimization problem. The beam pattern of this example is shown in **Figure 6(a)**.

Another example for the optimization problems without perfect matching is given here with the desired function being a rectangular function (**Figure 7(a)**). The ripple of the shaped main beam should not exceed 0.1 dB, while the SLL less than -28 dB. The algorithm to calculate the adaptive weight function is described with a flow diagram in **Figure 3**.

3. Numerical Results

Examples of the array pattern synthesis problem using CDM are presented and compared with PSO, GA and DE. All simulations are carried out on Intel® Xeon® CPU E5-2687W v2 @ 3.40 GHz with 256 GB RAM computer. All simulations use the fitness function (4) with different weights. In the following we use

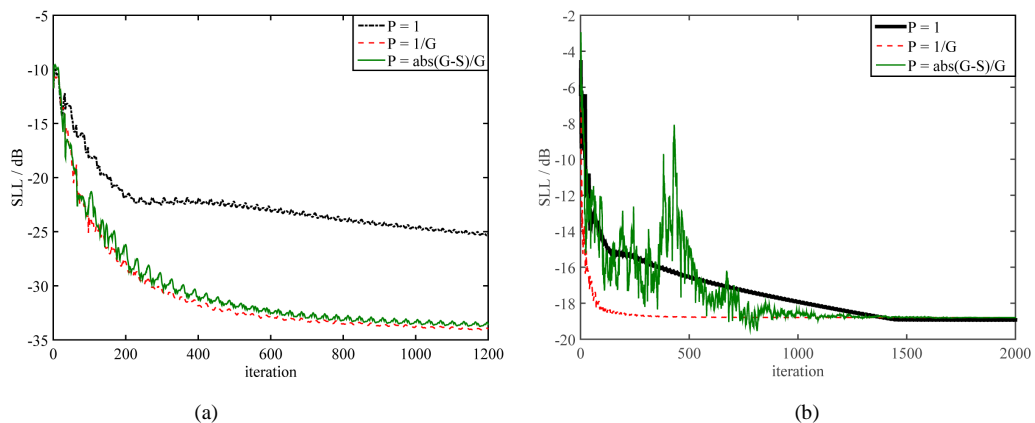


Figure 2. Investigations of relationships between SLL and weights. (a) Dolph-Chebyshev beam pattern; (b) Shaped beam pattern.

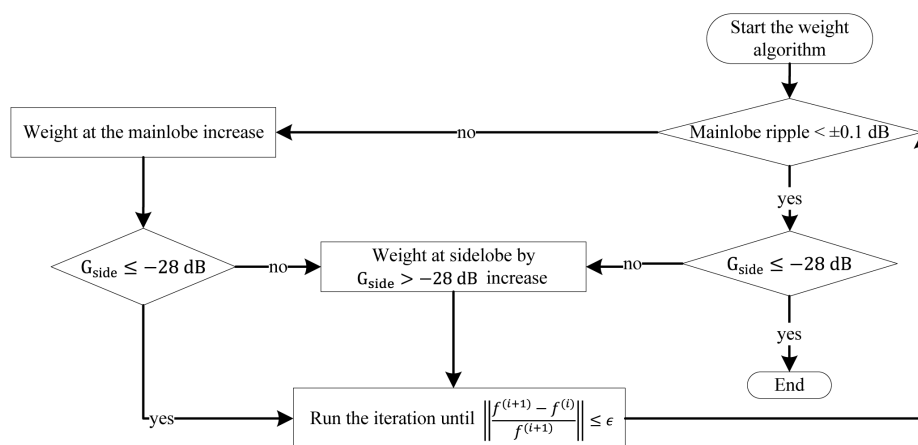


Figure 3. Flow process chart for a non-perfect matching (shaped beam pattern in **Figure 7(a)**).

$F(\mathbf{w}, \phi) = \frac{1}{1 + f(\mathbf{w}, \phi)}$ as quality criterion, because $F(\mathbf{w}, \phi) \in [0, 1]$ is more suited for representation in diagram as $f(\mathbf{w}, \phi) \in [0, \infty)$.

3.1. Dolph-Chebyshev Synthesis

In this example the desired pattern is a Dolph-Chebyshev pattern with 32 elements and SLL by -35 dB. Equal distance d between two elements is half wavelength. In **Figure 4(a)** original Dolph-Chebyshev pattern and three other patterns from CDM, PSO and DE are shown. The desired function $S(\theta)$ is an original Dolph-Chebyshev, thus it provides a perfect matching. The main lobes from all methods do not much differ from the original one. The pattern from CDM shows the best match of the SLL (perfect matching). This can be explained with the amplitude and phase distribution which is shown in **Figure 5**. **Figure 4(b)** represents the convergence of the fitness function over the time. $F(\mathbf{w}, \phi)$ by using CDM converges to 0.9999 in 5.8 seconds, while by using DE 1500 seconds are needed to achieve the same results. The optimization by using PSO, which converges to 0.9999 in 76 seconds, is neither get best results nor a short calculation time.

3.2. Shaped Beam Synthesis

The synthesis of shaped beams is a classical problem since it has many applications ranging from radar and remote sensing to communication systems [34]. Two examples are shown, one is a perfect matching, the other one is a non-perfect matching. The first example is shown in **Figure 6(a)**. This pattern has been represented in [33]. All three algorithms can synthesize a very similar beam pattern as the desired pattern. However, only the CDM succeed a perfect matching. The time of optimization by using CDM is 4.5 seconds for a $F(\mathbf{w}, \phi) \geq 0.9$, while the PSO 35 seconds and DE 1200 seconds needed for the same result.

The second example (**Figure 7(a)**) is a non-perfect matching, because the desired function $S(\theta)$ is a rectangular function, which means that it is not possible to achieve a perfect matching with finite elements. For this synthesis the weighting algorithm in **Figure 3** has been used to achieve the requirements which is described in subsection 3. The progress of convergence $F(\mathbf{w}, \phi)$ is shown in **Figure 7(b)**. It converges at first at high speed and then become stable after 50 seconds. Since the optimization is non-perfect matching, the $F(\mathbf{w}, \phi)$ is not possible to reach one.

3.3. The Array with Element Failures

Beam pattern reconfiguration of an antenna array with faulty elements has practical importance for antennas which cannot be repaired immediately. The possibility of reconfiguration of the antenna array with faulty elements has been considered by several authors over the years [35]-[39]. In this example we take the results from [36] as a reference. In this case the elements with Nr. 2, Nr. 5 and Nr. 6 of 32 elements Dolph-Chebyshev array

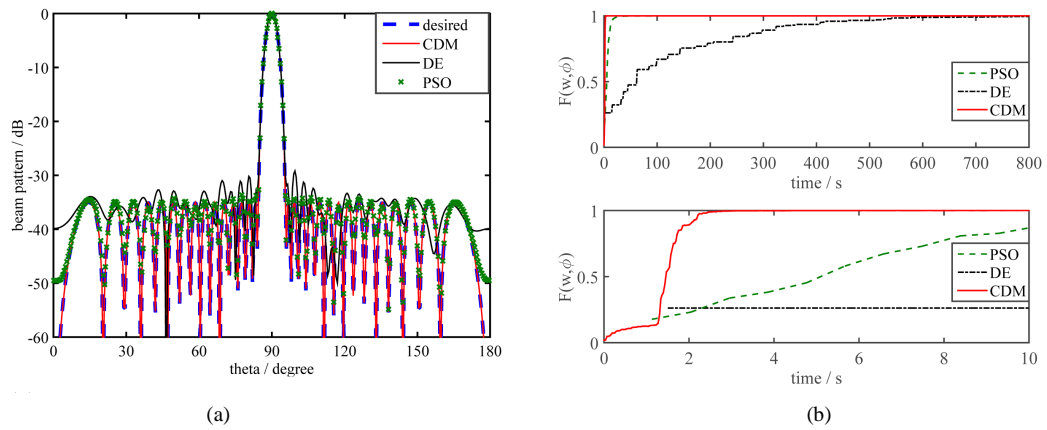


Figure 4. Synthesize a Dolph-Chebyshev beam pattern with three different optimization algorithms.

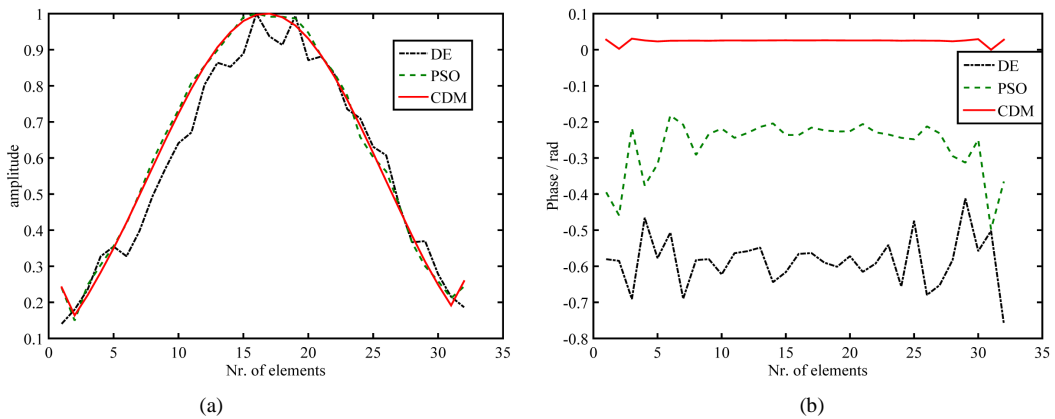


Figure 5. Comparison the results of optimizations of the Dolph-Chebyshev beam pattern (perfect matching).

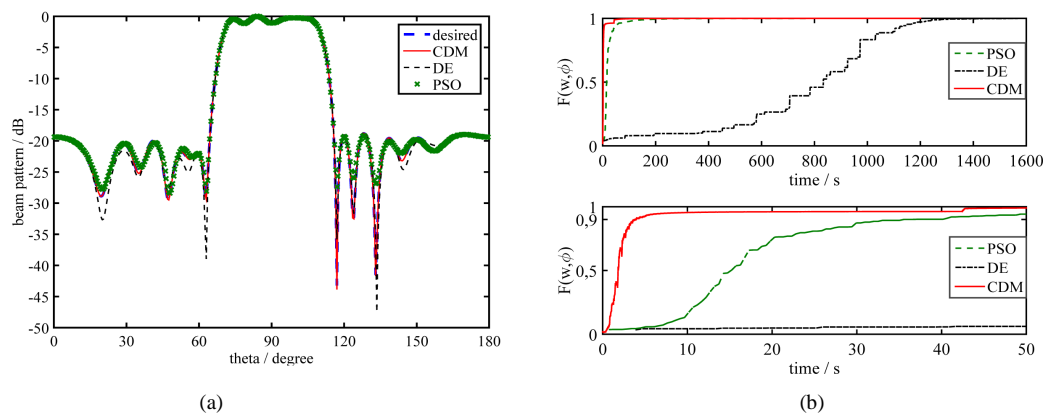


Figure 6. Synthesize a shaped beam pattern with three different optimization algorithms (perfect matching).

are failed, while the faulty elements do not radiate at all (“on-off” fault [40]). The result is shown in **Figure 8**. The half-power beam width of the results from CDM is 4.79° , which is 5.36° with GA. The SLL from CDM is -35.3 dB, which satisfies the original Dolph-Chebyshev condition as well.

3.4. The Planar Array

In this example the minimization of the SLL of a 8×8 array with uniform element distance half wavelength is

presented. In this case the half power beam width should keep 30° , while the SLL as low as possible. The result is shown in **Figure 9(b)**. The SLL has sunk from -12.5 dB to -21 dB.

4. Conclusion

An iterative method, which is based on the coordinate descent method to synthesize most of the current beam

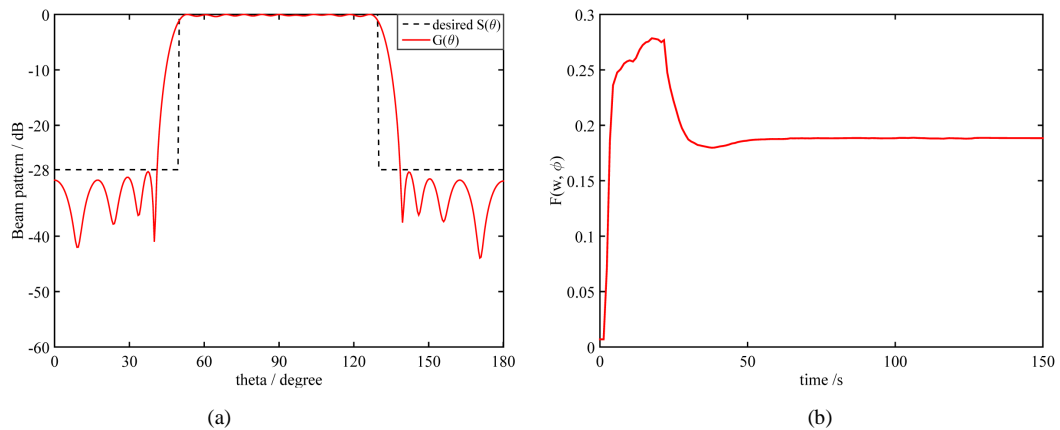


Figure 7. Synthesize a shaped beam pattern with CDM (non-perfect matching).

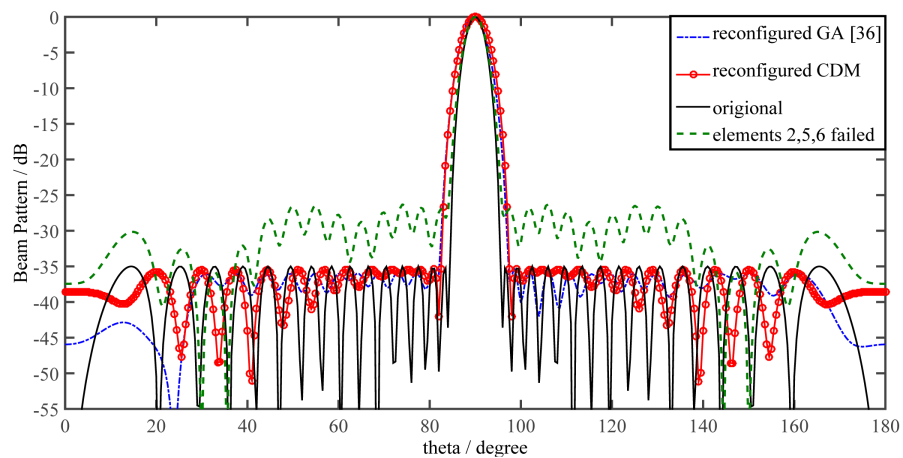


Figure 8. Dolph-Chebyshev array with 3rd, 5th and 6th elements failed.

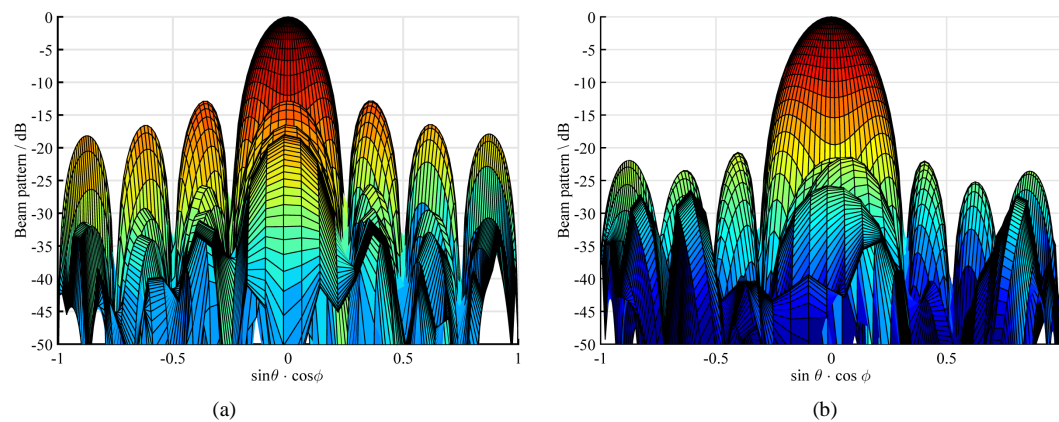


Figure 9. Beam pattern synthesis for planar array with CDM.

patterns from linear to planar array, is presented above. CDM allows changing the fitness function with a multiplication of a weight function $P(\theta)$. The function $P(\theta)$ can be changed any time during the optimization procedure depending on the requirement of the desired beam pattern. Therefore, we believe that it is worth investing more research in this area, in order to find more efficient and effective weight functions, and specialize the method for more array pattern synthesis problems. Due to the L_2 norm, the fitness function converges by every iteration, so the algorithm is very efficient to reach the optimum solution. Compared with other methods like PSO, GA and DE, CDM is accurater and faster, also easier to implement.

Acknowledgements

We thank the Editor and the referee for their comments.

References

- [1] Lebrete, H. and Boyd, S. (1997) Antenna Array Pattern Synthesis via Convex Optimization. *IEEE Transactions on Signal Processing*, **45**, 526-532. <http://dx.doi.org/10.1109/78.558465>
- [2] Schelkunoff, S.A. (1943) A Mathematical Theory of Linear Arrays. *Bell System Technical Journal*, **22**, 80-107. <http://dx.doi.org/10.1002/j.1538-7305.1943.tb01306.x>
- [3] Dolph, C.L. (1946) A Current Distribution for Broadside Arrays Which Optimizes the Relationship between Beam Width and Side-Lobe Level. *Proceedings of the IRE*, **34**, 335-348. <http://dx.doi.org/10.1109/JRPROC.1946.225956>
- [4] Taylor, T.T. (1955) Design of Line-Source Antennas for Narrow Beamwidth and Low Side Lobes. *Transactions of the IRE Professional Group on Antennas and Propagation*, **3**, 16-28. <http://dx.doi.org/10.1109/TPGAP.1955.5720407>
- [5] Goldberg, D.E. (1989) Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley Professional, Boston.
- [6] Storn, R. and Price, K. (1997) Differential Evolution—A Simple and Efficient Heuristic for Global Optimization over Continuous Spaces. *Journal of Global Optimization*, **11**, 341-359. <http://dx.doi.org/10.1023/A:1008202821328>
- [7] Kennedy, J. and Eberhart, R. (1995) Particle Swarm Optimization. *IEEE International Conference on Neural Networks*, **4**, 1942-1948. <http://dx.doi.org/10.1109/icnn.1995.488968>
- [8] Taguchi, G., Chowdhury, S. and Wu, Y. (2004) Taguchi's Quality Engineering Handbook. Wiley-Interscience, Hoboken.
- [9] Weng, W.-C., Yang, F. and Elsherbeni, A.Z. (2007) Linear Antenna Array Synthesis Using Taguchi's Method: A Novel Optimization Technique in Electromagnetics. *IEEE Transactions on Antennas and Propagation*, **55**, 723-730. <http://dx.doi.org/10.1109/TAP.2007.891548>
- [10] Fuchs, B., Skrivervik, A. and Mosig, J.R. (2013) Shaped Beam Synthesis of Arrays via Sequential Convex Optimizations. *IEEE Antennas and Wireless Propagation Letters*, **12**, 1049-1052. <http://dx.doi.org/10.1109/LAWP.2013.2280043>
- [11] Nongpiur, R.C. and Shpak, D.J. (2014) Synthesis of Linear and Planar Arrays with Minimum Element Selection. *IEEE Transactions on Signal Processing*, **62**, 5398-5410. <http://dx.doi.org/10.1109/TSP.2014.2350966>
- [12] Subhashini, K.R. and Praveen Kumar, A.T. (2014) Comparative Analysis of Linear and Nonlinear Pattern Synthesis of Hemispherical Antenna Array Using Adaptive Evolutionary Techniques? *International Journal of Antennas and Propagation*, **2014**, Article ID: 987140. <http://dx.doi.org/10.1155/2014/987140>
- [13] Haupt, R.L. (1994) Thinned Arrays Using Genetic Algorithms. *IEEE Transactions on Antennas and Propagation*, **42**, 993-999. <http://dx.doi.org/10.1109/8.299602>
- [14] Haupt, R.L. and Johnson, J.M. (1999) Dynamic Phase-Only Array Beam Control Using a Genetic Algorithm. *Proceedings of the 1st NASA/DoD Workshop on Evolvable Hardware*, Pasadena, 19-21 July 1999, 217-224. <http://dx.doi.org/10.1109/EH.1999.785456>
- [15] Marcano, D. and Duran, F. (2000) Synthesis of Antenna Arrays Using Genetic Algorithms. *IEEE Antennas and Propagation Magazine*, **42**, 12-20. <http://dx.doi.org/10.1109/74.848944>
- [16] Ares-Pena, F.J., Rodriguez-Gonzalez, J.A., Villanueva-Lopez, E. and Rengarajan, S.R. (1999) Genetic Algorithms in the Design and Optimization of Antenna Array Patterns. *IEEE Transactions on Antennas and Propagation*, **47**, 506-510. <http://dx.doi.org/10.1109/8.768786>
- [17] Elkamchouchi, H.M. and Hassan, M.M. (2014) Array Pattern Synthesis Approach Using a Genetic Algorithm. *IET Microwaves, Antennas & Propagation*, **8**, 1236-1240. <http://dx.doi.org/10.1049/iet-map.2013.0718>
- [18] Kurup, D.G., Himdi, M. and Rydberg, A. (2003) Synthesis of Uniform Amplitude Unequally Spaced Antenna Arrays Using the Differential Evolution Algorithm. *IEEE Transactions on Antennas and Propagation*, **51**, 2210-2217.

- <http://dx.doi.org/10.1109/TAP.2003.816361>
- [19] Goudos, S.K., Siakavara, K., Samaras, T., Vafiadis, E.E. and Sahalos, J.N. (2011) Self-Adaptive Differential Evolution Applied to Real-Valued Antenna and Microwave Design Problems. *IEEE Transactions on Antennas and Propagation*, **59**, 1286-1298. <http://dx.doi.org/10.1109/TAP.2011.2109678>
- [20] Guney, K. and Basbug, S. (2013) Null Synthesis of Time-Modulated Circular Antenna Arrays Using an Improved Differential Evolution Algorithm. *IEEE Antennas and Wireless Propagation Letters*, **12**, 817-820. <http://dx.doi.org/10.1109/LAWP.2013.2271273>
- [21] Robinson, J. and Rahmat-Samii, Y. (2004) Particle Swarm Optimization in Electromagnetics. *IEEE Transactions on Antennas and Propagation*, **52**, 397-407. <http://dx.doi.org/10.1109/TAP.2004.823969>
- [22] Boeringer, D.W. and Werner, D.H. (2004) Particle Swarm Optimization versus Genetic Algorithms for Phased Array Synthesis. *IEEE Transactions on Antennas and Propagation*, **52**, 771-779. <http://dx.doi.org/10.1109/TAP.2004.825102>
- [23] Khodier, M.M. and Christodoulou, C.G. (2005) Linear Array Geometry Synthesis with Minimum Sidelobe Level and Null Control Using Particle Swarm Optimization. *IEEE Transactions on Antennas and Propagation*, **53**, 2674-2679. <http://dx.doi.org/10.1109/TAP.2005.851762>
- [24] Donelli, M., Martini, A. and Massa, A. (2009) A Hybrid Approach Based on PSO and Hadamard Difference Sets for the Synthesis of Square Thinned Arrays. *IEEE Transactions on Antennas and Propagation*, **57**, 2491-2495. <http://dx.doi.org/10.1109/TAP.2009.2024570>
- [25] Roy, S., Martinez, S.Z., Coello Coello, C.A. and Sengupta, S. (2012) A Multi-Objective Evolutionary Approach for Linear Antenna Array Design and Synthesis. *Proceedings of the 2012 IEEE Congress on Evolutionary Computation (CEC)*, Brisbane, 10-15 June 2012, 1-8. <http://dx.doi.org/10.1109/CEC.2012.6252989>
- [26] Zangwill, W. (1969) *Nonlinear Programming: A Unified Approach*. Prentice-Hall, Englewood Cliffs.
- [27] Kragh, T.J. (2006) Monotonic Iterative Algorithm for Minimum-Entropy Autofocus? *Proceedings of the Adaptive Sensor Array Processing Workshop*, Lexington, MA, 6-7 June 2006.
- [28] Ash, J.N. (2012) An Autofocus Method for Backprojection Imagery in Synthetic Aperture Radar. *IEEE Geoscience and Remote Sensing Letters*, **9**, 104-108. <http://dx.doi.org/10.1109/LGRS.2011.2161456>
- [29] Bouman, C.A. and Sauer, K. (1996) A Unified Approach to Statistical Tomography Using Coordinate Descent Optimization. *IEEE Transactions on Image Processing*, **5**, 480-492. <http://dx.doi.org/10.1109/83.491321>
- [30] Shen, H.F., Zhang, L.P., Huang, B. and Li, P.X. (2007) A MAP Approach for Joint Motion Estimation, Segmentation, and Super Resolution. *IEEE Transactions on Image Processing*, **16**, 479-490. <http://dx.doi.org/10.1109/TIP.2006.888334>
- [31] Nesterov, Y. (2012) Efficiency of Coordinate Descent Methods on Huge-Scale Optimization Problems. *SIAM Journal on Optimization*, **22**, 341-362. <http://dx.doi.org/10.1137/100802001>
- [32] Perini, J. and Idselis, M. (1971) Note on Antenna Pattern Synthesis Using Numerical Iterative Methods. *IEEE Transactions on Antennas and Propagation*, **19**, 284-286. <http://dx.doi.org/10.1109/TAP.1971.1139919>
- [33] Bucci, O.M., Mazzarella, G. and Panariello, G. (1989) Reconfigurable Arrays by Phase-Only Control. *Proceedings of the IEEE Antennas and Propagation Society International Symposium*, San Jose, 26-30 June 1989, 142-145. <http://dx.doi.org/10.1109/APS.1989.134633>
- [34] Elliott, R.S. (1981) *Antenna Theory and Design*. Prentice-Hall, Englewood Cliffs.
- [35] Peters, T.J. (1991) A Conjugate Gradient-Based Algorithm to Minimize the Sidelobe Level of Planar Arrays with Element Failures. *IEEE Transactions on Antennas and Propagation*, **39**, 1497-1504. <http://dx.doi.org/10.1109/8.97381>
- [36] Yeo, B.-K. and Lu, Y. (1999) Array Failure Correction with a Genetic Algorithm. *IEEE Transactions on Antennas and Propagation*, **47**, 823-828. <http://dx.doi.org/10.1109/8.774136>
- [37] Zainud-Deen, S.H., Ibrahim, M.S., Sharshar, H.A. and Ibrahim, S.M.M. (2004) Array Failure Correction with Orthogonal Method. *Proceedings of the Twenty-First National Radio Science Conference*, Cairo, 16-18 March 2004, B7-1-9.
- [38] Mitilneos, S.A. and Capsalis, C.N. (2005) On Array Failure Mitigation Using Genetic Algorithms and a Priori Joint Optimization. *IEEE Antennas and Propagation Magazine*, **47**, 227-232. <http://dx.doi.org/10.1109/MAP.2005.1599213>
- [39] Keizer, W.P.M.N. (2007) Element Failure Correction for a Large Monopulse Phased Array Antenna with Active Amplitude Weighting. *IEEE Transactions on Antennas and Propagation*, **55**, 2211-2218. <http://dx.doi.org/10.1109/TAP.2007.902008>
- [40] Bucci, O.M., Capozzoli, A. and D'Elia, G. (2000) Diagnosis of Array Faults from Far-Field Amplitude-Only Data. *IEEE Transactions on Antennas and Propagation*, **48**, 647-652. <http://dx.doi.org/10.1109/8.855482>