A Modified Unequally Spaced Array Antenna Synthesis Method for Side Lobe Reduction

Soroush VEISEE, Shahrooz ASADI

Dept. of Electrical Engineering, University of Shahid Beheshti, Tehran, Iran
veisee@aut.ac.ir, sh_asadi@sbu.ac.ir

Submitted October 14, 2016 / Accepted February 7, 2017

Abstract. The aim of this paper is to demonstrate the application of Unequally Spaced Arrays (USAs) in decreasing side lobe level (SLL) in linear arrays. As well known, solving of a nonlinear equation is needed in USA antenna pattern synthesis. In this paper, an improved algorithm for USA antenna pattern synthesis is presented. This method is based on converting the array factor into a triangular system of equations capable to be solved using a recursive algorithm. This method has more accuracy and speed than reported similar analytical methods based on simulation results, which leads to lower SLL and simulation time. In addition, an improvement of 3dB beamwidth in comparison with uniform spaced array can be observed.

Keywords
Antenna arrays, unequal spacing, pattern synthesis, side lobe level

1. Introduction

Uniformly spaced array antenna synthesis techniques have been deeply investigated in past decades. Traditional synthesis techniques such as Dolph-Chebyshev [1], Taylor [2], and Fourier transform, can lead to design of a desired pattern. These synthesis procedures change the phase and magnitude of elements current to reduce the side lobe level (SLL). However, the amplitude/phase distribution can complicate the designing of the array antenna feed network. In addition, it cannot optimize the use of all elements in terms of maximum power. Non-uniformly spaced array antenna can solve this issue. This kind of array antenna has also other significant advantages such as smaller size and reduced number of elements. As a result, non-uniformly spaced array antennas are widely used in radar, sonar, and wireless communications.

Variety of numerical and analytical methods [7–20] have been introduced for unequally spaced array (USA) pattern synthesis. The first work on USA pattern synthesis was performed by Unz [3] by using a matrix formulation to obtain the distribution of elements current of an USA to form a desired pattern. Then, Harrington [4] presented a method for reducing SLL using non-uniform element spacing and showed that SLL can be reduced to the level of $2/N^{th}$ of the main lobe level. In [5], Ishimaru used Poisson’s formula to reduce SLL in comparison with a linear array with uniform amplitude distribution. Miller and Goodman [6] proposed a simple and non-iterative method for USA pattern synthesis using Prony’s method. In [7], [8], Kumar proposed a recursive algorithm for USA pattern synthesis based on Legendre conversion.

On the other side, stochastic methods such as Genetic Algorithm (GA) [11], Particle Swarm Optimization (PSO) and Differential Evolution Algorithm (DEA) [10] have been recently used in pattern synthesis. However, these optimization methods require large CPU time, which further increases with the increment of array element numbers.

In USA pattern synthesis, analytical methods are affordable in time because of the non-iterative property. In this paper, an analytical method for pattern synthesis of unequally spaced arrays is proposed. It is an enhanced version of the method described in [7], in terms of higher precision and speed. Compared with the method in [7], which is used for odd number of elements, the method presented in this paper with some modification, can be used also for even number of elements.

2. Theory

The geometry of a $(2N+1)$ element non-uniform linear array is shown in Fig. 1, where $I_i$ and $d_i$ state respectively for the element position and current.

Fig. 1. Geometry of a symmetrical non-uniform linear array.
This paper proposes a new synthesis pattern approach that adjusts the element position in a linear array with a uniform current distribution, in order to quickly and accurately improve the radiation characteristics such as lower peak side lobe level (PSLL). This method can indeed determine the element positions of the array in order to fit the array pattern to a pre-specified array pattern. The desired array pattern is defined as where, \(-1 \leq u \leq 1\) and \(u = \cos \theta\), \(0 \leq \theta \leq \pi\) radians. As known, the array pattern of a symmetric array with \((2N + 1)\) elements is defined as:

\[
E(u) = \sum_{n=0}^{N} I_n \cos(kd_n u). \quad (1)
\]

By sampling the above relation with a sampling space \(\Delta u = 1/(M-1)\), we obtain:

\[
E(u_m) = \sum_{n=0}^{N} I_n \cos(m \beta_n) \quad (2)
\]

where \(u_m = m \Delta u\) and \(\beta_n = k d_n \Delta u\).

Because of the symmetry of the pattern in the \(u\) space, this sampling is performed in \(0 \leq u \leq 1\) interval or \(m = 0,1,2,3,\ldots, N\). The element position must be chosen in such a way where:

\[
E(u_m) = A(u_m) \quad (3)
\]

or

\[
E_d(u_m) = \sum_{n=0}^{N} I_n \cos(m \beta_n), \quad m = 0,1,2,3,\ldots, M-1 \quad (4)
\]

where \(E_d(u)\) is the desired pattern.

Relation (1) is a nonlinear equation. To solve it, both sides of (4) are multiplied by a matrix \(A\) defined as:

\[
A(\alpha) = [a_s(\alpha) \quad a_s(\alpha) \quad a_2(\alpha) \quad \ldots \quad a_{M-1}(\alpha)] \quad (5)
\]

leading to

\[
\sum_{m=0}^{M-1} a_m(\alpha) E_d(u_m) = \sum_{n=0}^{N} I_n \sum_{m=0}^{M-1} a_m(\alpha) \cos(m \beta_n) d \quad (6)
\]

The values of \(a_m(\alpha)\) coefficients are selected as:

\[
f(\alpha, \beta) = \sum_{m=0}^{M-1} a_m(\alpha) \cos(m \beta) \quad (7)
\]

where [7]

\[
f(\alpha, \beta) = \begin{cases} 
\frac{2}{(\cos(\beta) - \cos(\alpha))^2} & 0 \leq \beta < \alpha \\
0 & \alpha \leq \beta \leq \pi
\end{cases} \quad (8)
\]

Also, by expressing the left side of (6) as:

\[
F(\alpha_p) = \sum_{m=0}^{M-1} a_m(\alpha_p) E_d(u_m) \quad . \quad (9)
\]

This later can be simplified to:

\[
F(\alpha_p) = \sum_{n=0}^{N} I_n f(\alpha_p, \beta_n). \quad (10)
\]

In [7], Legendre coefficients \(p_{m-0.5}(\cos \alpha)\) are used instead of \(a_m(\alpha)\) in (7), which are the Fourier series coefficients of \(f(\alpha, \beta)\). But in this paper the values of \(a_m\) coefficients are obtained by minimizing the error term stated below:

\[
\text{error} = \sum_{i=0}^{M-1} \left| f(\alpha, \beta_i) - \sum_{m=0}^{M-1} a_m(\alpha) \cos(m \beta_i) \right|^2, \quad (11)
\]

which leads to

\[
f(\alpha, \beta_i) - \sum_{m=0}^{M-1} a_m(\alpha) \cos(m \beta_i) = 0, \quad i = 0,1,\ldots, M-1 \quad (12)
\]

In fact, the left hand side of (7) is considered to be equal to the right hand side in \(\beta_i\) points, which \(i = 0,1,\ldots, M-1\).

After some manipulations, equation (12) can be stated as a matrix equation as below:

\[
B \times A(\alpha) = F. \quad (13)
\]

Therefore, matrix \(A(\alpha)\) can be obtained as:

\[
A(\alpha) = B^{-1} \times F \quad (14)
\]

which leads to obtain the values of \(a_m(\alpha)\). In order to avoid misunderstanding, in continue, instead of \(a_m(\alpha)\) which derived from (14), we use \(a'_m(\alpha)\).

In (14), the matrices \(B\) and \(F\) are defined as below:

\[
B = \begin{bmatrix}
1 & 1 & 1 & \ldots & 1 \\
1 & \cos(\beta_1) & \cos(2\beta_1) & \ldots & \cos((M-1)\beta_1) \\
1 & \cos(\beta_2) & \cos(2\beta_2) & \ldots & \cos((M-1)\beta_2) \\
\vdots & \vdots & \vdots & \ddots & \vdots \\
1 & \cos(\beta_{M-1}) & \cos(2\beta_{M-1}) & \ldots & \cos((M-1)\beta_{M-1})
\end{bmatrix}
\]

and

\[
F = \begin{bmatrix}
f(\alpha, \beta_1) \\
f(\alpha, \beta_1) \\
f(\alpha, \beta_2) \\
\vdots \\
f(\alpha, \beta_{M-1})
\end{bmatrix}
\]

The value of \(\alpha\) is defined as:

\[
a_p = \beta_p + V_p \quad (15)
\]
in which the value of $V_p$ is defined such that:

$$\alpha_p < \beta_{p+1}. \quad (16)$$

From (9) and (14), we obtain:

$$F(\alpha_p) = \sum_{n=0}^{\infty} I_n f(\alpha_p, \beta_n)$$

which can be reformatted as:

$$F(\alpha_n) = I_0 f(\alpha_n, \beta_0) \quad F(\alpha_1) = I_0 f(\alpha_1, \beta_0) + I_1 f(\alpha_1, \beta_1) \quad \ldots$$

$$F(\alpha_p) = \sum_{n=0}^{p-1} I_n f(\alpha_p, \beta_n) + I_p f(\alpha_p, \beta_p) \quad (17)$$

The locations of the array elements for a defined current distribution can be then obtained from the above equation using a recursive algorithm. In the proposed algorithm, the distance between elements was selected as:

$$0.5\lambda \leq d_p - d_{p-1} \leq 0.5\lambda + \Delta(p), \quad p = 1, 2, \ldots, N$$

where $\Delta(p)$ is the variable space broadening factor for the $p$th element. The lower limit of (17) is able to prevent mutual coupling effects and the upper limit is determined in order to prevent from grating lobes. It is noted that in comparison with [7] which used a constant value for $\alpha_p, \beta_p$, in this paper we obtained an optimized value for each value of $\Delta(p)$. Let us define some parameters such as:

- $d_0$, the place of the first element, assumed to be zero, which leads to $\beta_0 = 0$,
- $I_0$ the normalized value of the first element current as:

$$I_0 = \frac{F(\alpha_0)}{f(\alpha_0, \beta_0)},$$

$$\alpha$$ such as:

$$\alpha_0 = k(d_0 + \frac{\lambda}{2})\Delta u \quad (20)$$

and

$$\alpha_p = k(d_{p-1} + \frac{\lambda}{2} + \Delta(p))\Delta u. \quad (21)$$

Before starting the algorithm, the initial values of all $\Delta(p)$ are considered to be equal with $\Delta$, where $\Delta$ is a constant space broadening factor to be specified by the user. Then the values of $\Delta(p)$ should be optimized in order to minimize PSLL. This algorithm includes $N$ steps. In the $n$th step, the optimized value of $\Delta(n)$ is obtained and the other values of $\Delta(p)$, $p \neq n$ are considered as below:

For $p > n$, $\Delta(p) = \Delta$.

For $p < n$, $\Delta(p)$ is equal to the optimized value obtained in previous steps.

In order to get the optimized value of $\Delta(n)$, the value of $\Delta(n)$ varies from 0 to $0.5\lambda$ with 0.01$\lambda$ step size. For each value of $\Delta(n)$, by utilizing the following procedure, a set of element positions ($d_0, d_1, \ldots, d_N$) can be obtained.

Assuming $\beta_0 = 0$, the values of $\beta_p$ for $p = 1, 2, 3, \ldots, N$, are calculated as [7]:

$$\beta_p = \cos^{-1}\left(g(\alpha_p)\right),$$

$$g(\alpha_p) = \frac{1}{\left[F(\alpha_p) - \sum_{n=0}^{p-1} I_n f(\alpha_p, \beta_n)\right]^2 + \cos(\alpha_p)}. \quad (23)$$

Thus,

$$d_p = \beta_p / (k\Delta u).$$

For each value of $d_p$ from (23), the conditions below should be applied.

- If $d_p$ is a complex number, $d_p = d_{p-1} + 0.5\lambda$.
- If $d_p$ is larger than $\alpha_p, d_p = d_{p-1} + 0.5\lambda$.
- If the element spacing between $d_p$ and $d_{p-1}$ is less than $0.5\lambda, d_p = d_{p-1} + 0.5\lambda$.

Then, utilizing the obtained set of element positions from above procedure and using (1), the array pattern and PSLL of the related set of positions can be acquired. Therefore, in the $n$th step, for each value of $\Delta(n)$, there is a related value of PSLL. The optimized value of $\Delta(n)$ is the one which has the minimum related value of PSLL. This selected value of $\Delta(n)$ should be used for the $(n + 1)$th step.

In order to apply the mentioned method to an array of even number of elements ($2N$), $\alpha_0$ and $d_0$ should be selected as:

$$\alpha_0 = k(\Delta_0 + \frac{\lambda}{2})\Delta u, \quad d_0 = 0.25\lambda. \quad (24)$$

In the above equation, $\Delta_0$ is called the initial space broadening factor and its value is changed from 0 to $\lambda$. In this way, $\Delta_0$ and $\Delta$ should be initially defined by users.

### 3. Numerical Results

In this section, the presented method is demonstrated through examples. In all these examples, it is assumed that the current distribution amplitude is uniform and unequally spaced is used in order to decrease SLL. The desired pattern to design an unequally spaced array with $q$ elements is defined as:

$$E_q(u) = \begin{cases} 1 & 0 \leq u \leq u_m \\ E_{min} & u_m \leq u \leq 1 \end{cases} \quad (25)$$

The value of $E_{min}$ is typically selected as $1 \times 10^{-3}$. It is shown that this value should be decreased with the element number increment. $u_m$ is equal to the place of the first pattern null of an equally spaced array with the same number of elements $q$. 

As mentioned before, the proposed method in this paper is more accurate vs. similar existing methods [7]. It is noted that the right hand side of (7) is an estimation of \( f(\alpha, \beta) \) function defined in (8). Using the \( a'_m(\alpha) \) coefficients and also the Legendre coefficients \( p_{m, 0.5}(\cos \alpha) \) [7], instead of \( a_m(\alpha) \) in the right hand side of (7), two approximate functions can be obtained to estimate the function \( f(\alpha, \beta) \) defined in (8). In Fig. 2, these two approximate functions and \( f(\alpha, \beta) \) in (8) are plotted for \( M = 107 \) and \( \alpha = 1 \). As can be seen, the \( a'_m(\alpha) \) coefficients can better estimate \( f(\alpha, \beta) \) than the Legendre coefficients. Another advantage of this method is its speediness since the \( a'_m(\alpha) \) coefficients can be determined faster than \( p_{m, 0.5}(\cos \alpha) \) for \( M = 107 \) and \( \alpha = 1 \).

To validate the proposed synthesis method, a linear unequally spaced array of 39 elements was evaluated. In its pattern, shown in Fig. 3, the SLL has been decreased by 8.15 dB in comparison with a uniform array with the same number of elements. In addition, this value was improved by 1.75 dB while compared to the one obtained by the method described in [7]. Also as can be seen, the beamwidth is narrower than the case with uniform space. The 3-dB beamwidth of the uniform and non-uniform arrays are 2.52° and 2.25° respectively. In this algorithm, the values of \( \Delta \) and \( M \) were set to 0.33\( \lambda \) and 107, respectively. The obtained elements positions are reported in Tab. 1. As can be inferred from (23), in order to obtain the value of element positions \( (d_0, d_1, \ldots, d_N) \), we need to calculate the value of \( F(\alpha) \) function using (9) and then calculating the values of \( a_m(\alpha) \). As mentioned previously, since the \( a'_m(\alpha) \) coefficients can be determined faster than \( p_{m, 0.5}(\cos \alpha) \), therefore, using \( a'_m(\alpha) \) instead of \( p_{m, 0.5}(\cos \alpha) \) makes our proposed method faster than the Legendre method. For the linear unequally spaced array of 39 element discussed above, with \( \Delta = 0.33\lambda \) and \( M = 107 \), the measured simulation time shows that for this case, the simulation speed of our proposed method is 79 time faster in comparison with the Legendre method. As a result, the proposed method is an appropriate analytical method for synthesis of large unequally spaced array.

![Fig. 2. Estimation of \( f(\alpha, \beta) \) function using the proposed method and Legendre method [7].](image)

![Fig. 3. Synthesized pattern of a 39 elements nonuniform array.](image)

![Fig. 4. Synthesized pattern of a 200 elements nonuniform array.](image)

<table>
<thead>
<tr>
<th>Element number</th>
<th>Element position</th>
<th>Element number</th>
<th>Element position</th>
<th>Element number</th>
<th>Element position</th>
</tr>
</thead>
<tbody>
<tr>
<td>p</td>
<td>( d_0 / \lambda )</td>
<td>p</td>
<td>( d_1 / \lambda )</td>
<td>p</td>
<td>( d_2 / \lambda )</td>
</tr>
<tr>
<td>1</td>
<td>0.5</td>
<td>8</td>
<td>4.00</td>
<td>15</td>
<td>8.63</td>
</tr>
<tr>
<td>2</td>
<td>1.0</td>
<td>9</td>
<td>4.52</td>
<td>16</td>
<td>9.46</td>
</tr>
<tr>
<td>3</td>
<td>1.5</td>
<td>10</td>
<td>5.12</td>
<td>17</td>
<td>10.41</td>
</tr>
<tr>
<td>4</td>
<td>2.0</td>
<td>11</td>
<td>5.71</td>
<td>18</td>
<td>11.32</td>
</tr>
<tr>
<td>5</td>
<td>2.5</td>
<td>12</td>
<td>6.40</td>
<td>19</td>
<td>12.12</td>
</tr>
<tr>
<td>6</td>
<td>3.0</td>
<td>13</td>
<td>7.12</td>
<td></td>
<td></td>
</tr>
<tr>
<td>7</td>
<td>3.5</td>
<td>14</td>
<td>7.87</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

**Tab. 1.** Element position for 39 elements array.
4. Conclusion

In this paper, a fast and accurate analytic method for synthesizing of unequally spaced arrays with uniform amplitude distribution was proposed. It can be implemented in arrays with both even and odd number of elements. This approach allows obtaining a radiation pattern with a narrow beamwidth and lowest PSLL value. The simulation results show a significant reduction of simulation time. In fact, the obtained results demonstrated its strong ability to design low side lobe level phased array antennas with unequally spaced elements.

References


