Application of Artificial Neural Networks System for Synthesis of Phased Cylindrical Arc Antenna Arrays

N. Fadlallah and L. Moustafa

Equipe RADICOM, Institut Universitaire de Technologie de Saida
Université de Liban, P.O.Box #813, Liban
n.fadlallah@ul.edu.lb, n_fadlallah@yahoo.com, lina.moustapha@liu.edu.lb

Abstract

In this paper, we present a neural network approach to the problem of cylindrical antenna array synthesis. In modern satellite mobile communications systems and global positioning systems, both desired and interfering signals change their directions continuously. Therefore, a fast tracking system is needed to constantly track the users and then adapt the radiation pattern of the antenna with direct multiple narrow beams to desired users and nulls interfering sources. This paper proposes a neural network architecture that consists of 2 sub-systems, each one composed of 8 patches and 9 sub-neural networks, to solve the synthesis of a cylindrical antenna array. The network’s training database contains a finite number of samples of cylindrical targets at certain angles. A part of the database is used to train the network and the rest is used to test its performance for target identification and classification. Used neural networks are multi-layered perceptron (MLP) with a back-propagation training algorithm. The proposed synthesis approach provides important improvements in terms of performances, computational speed (convergence’s time) and software implementation.

Index Terms— Adaptive arrays, cylindrical antenna arrays, beamforming, neural network applications, phased excitations.

1- Introduction

Adaptive antenna arrays applications require scanning angles with multilobes [1], nulling interferences and large lobes. In a smart antenna array system, the beam is electronically positioned by adjusting, in a predetermined manner, the excitation phases and amplitudes of the array’s elements. This synthesis operation is essential for the conception of optimized antennas.
Most of the synthesis methods use complex excitation (amplitudes and phases excitation) [2]; this allows a better control of the quality of the beam shape and side-lobe level. In a linear phased antenna array, the beam cannot usually be steered more than about 60° - 70° from the normal of the array [3].

As a conformal array, the cylinder has a potential of 360° coverage. Today, the common solution is to use three separate antennas, each covering a 120° sector [3]. In this paper, a cylindrical array with 2 sub-systems is used, each covering a 90° sector in order to perform 180° coverage. The concept may be then generalized to cover the 360° space using a 16 elements phase shifters. A general approach based on phase adjustment is proposed, in order to synthesize beam and/or multibeam with steered zero in desired directions. The basic idea of this present approach is to first apply the two terms of the Taylor expansion to express a steering vector in a particular direction and to construct a set of linear equations that can be optimized with the minmax criterion [4]. The resulting linear system can be solved in the minmax sense by a method similar to the Madsen algorithm. The advantages of this method are its efficiency and its ability to control the performances of desired beams simply by simple adjustment of the weights in the angular positions.

This technique will use neural networks with back-propagation algorithm for synthesis by phase control of steered beams, adaptive nulls and multi-beams patterns in the directions of multiple users [5-6]. Nowadays, this method is considered as one of the most efficient nonlinear programming algorithms.

The proposed approach is represented as a strategy that could be re-used and implemented to solve any antenna array synthesis problem. The basics of this strategy are explained in details in the sections that follow.

2. SYNTHESIS PROBLEM FORMULATION
The radiation field of a cylindrical array (Fig. 1) is expressed as:

\[ E(\phi) = \sum_{n=1}^{N} I_n e^{j\beta_n} E_0(\theta, \phi_n) e^{jkR \cos(\theta - \phi_n)} \]  

(1)

Where:
- \( I_n e^{j\beta_n} \) is the complex excitation for the element \( n \).
- \( \phi_n \) is the angular position of the element \( n \).
- \( E_0 \) is the elementary field.

![Fig. 1- Geometry of a cylindrical array](Image)
For conformal arrays, the elementary pattern plays an important role in the array pattern, because each element is facing a different direction. Therefore, the element pattern expression has a different value in each summand of equation (1).

The desired field is defined in amplitude throughout template (Fig. 2), that indicates the direction and shape of main lobes, position and width of zeros.

![Fig. 2- Electrical field template.](image)

The optimization problem under consideration consists in the minimization of the error function.

\[ \text{ERR}(\beta) = W^{\text{MAX}} \sum_j |\text{ERR}_j(\beta)| \]

Where:

\[ \text{ERR}_j(\beta) = \text{ERR}_j(\beta_1, \beta_2, \ldots, \beta_N) \]

\( j = 1, \ldots, M \) (M is the number of sampled angular directions).

\( \text{ERR}_j \) is the deviation of an actual calculated field (1) \( \mathbf{E}_c(\beta, \theta_j) \) from a desired one \( \mathbf{E}_d(\theta_j) \).

\[ \text{ERR}_j(\beta) = \mathbf{E}_c(\beta, \theta_j) - \mathbf{E}_d(\theta_j) \]

\( \mathbf{W}_j \) is a weighting factor in the direction \( \theta_j \), it can be adjusted to control the beam in all directions and create a deep zero in the direction of interfering sources.

At the k\text{th} stage of the minimization algorithm, we solve a linearized system based on minimax criterion.

\[ \text{ERR}_j(\beta_k) + \sum_{i=1}^{N} \frac{\partial \text{ERR}_i(\beta_k)}{\partial \beta_i} h_{ki} = 0 \]

\( \beta_k \) is automatically adapted during the process to find the inequality:

\[ \|h_k\| \leq \delta_k \] to ensure a good linear approximation of the set of linear equations.

\( \delta_k \) is automatically adjusted during the process to find the inequality:

\[ \text{ERR}(\beta_k + h_k) < \text{ERR}(\beta_k) \]
Convergence of the method is ensured by adjusting the value of $\delta_k$ at each iteration, so the point $x_{k+1} = x_k + h_k$ will be good if the decrease in the function $\text{ERR}(\beta)$ exceeds a small multiple the decrease predicted by the linear approximation. Otherwise the linear approximation is insufficient, the value of $\delta_{k+1}$ will be reduced.

The iteration is stopped if one of the following criteria is met:
- Maximum of the error function is below a certain value.
- The maximum of $\|A_k\|$ is very small compared to $\beta$.

3. NEURAL BEAMFORMER
3.1 The neural beamformer architecture

The Beamforming Neural Network (BFNN) consists of an antenna measurement pre-processing input, an Artificial Neural Network (ANN) and a post-processing output.

The true power and advantage of neural networks lies in their ability to represent both linear and non-linear relationships and in their ability to learn these relationships directly from the data being modeled. Traditional linear models are simply inadequate when it comes to modeling data that contains non-linear characteristics. The most common neural network model is the MultiLayer Perceptron (MLP). This type of neural network is known as a supervised network because it requires a desired output in order to learn. The goal of this type of network is to create a model that correctly maps the input to the output using historical data so that the model can then be used to produce the output when the desired output is unknown. A graphical representation of an MLP is shown below (Fig. 2):

Fig. 3- A two hidden layer MLP.

The MLP and many other neural networks learn using an algorithm called back propagation. With back propagation, the input data is repeatedly represented to the neural network. With each presentation the output of the neural network is compared to the desired output and an error is computed. This error is then fed back (back propagated) to the $E_d$ as "training".

In our work we are divided the cylindrical arc antenna into two sub-systems, and each one is composed of 8 elements arrays. The first sub-system is responsible to cover the space between $90^\circ$ and $0^\circ$, and the other sub-system responsible to cover the space between $0^\circ$ and $-90^\circ$. 
The architecture of the beamforming neural network (BFNN) with a backpropagation consists of 18 MLP linked in parallel. Each MLP section comprises 9 binary inputs neurons and four outputs neurons (targets). Each MLP is responsible for a fixed main lobe and another scanning lobe.

In the problem of synthesis, the amplitude of the received signal is not a strong indicator of the angle of arrival DOA. The absolute phase of the received signal at each element also contains non-essential information. There is, however, a strong relationship between element phases and angle of arrival DOA (Fig. 3).

3.2 Training, Simulations and Results
Pre-processing and post-processing of the BFNN configure the network interfaces to perform particular functions. The neural network approximates the function that we model by adapting its internal structure to map the problem space.

This section briefly summarizes the purpose and interaction of these functional elements.
Pre-processing: Network pre-processing exploits antenna expertise to simplify and enhance neural network inputs. It removes redundant or irrelevant information, eliminates artificial discontinuities in the input function space, and reduces problem inputs to a small set of relevant information. In the pre-processing phase, four steps should be performed.
1- The space between two entering positions is firstly tested, if it is greater than 20°, only one steering lobe is obtained.

2- The real synthesis is secondly used to eliminate 4 outputs (targets) due to the symmetry with the other 4 outputs.

3- Thirdly, the space is divided into 9 sectors, repeated every 10° in the interval [0° - 90°] and [90° - 180°]. More accurate sectors division can be reached by increasing the number of array’s elements. The input vectors to the network are represented in a 9-bit binary code (one bit for each sector); all bits are set to zero except two (+1 and -1) or (+1 and +1). A binary input of +1 indicates a source exactly on (main lobe) in the sector, the bin location of 0 represents no source in the sector and the bin location of -1 indicates a null interfering in the sector. This step has the advantage of considerably decreasing the number of unknown variables. Convergence may then be achieved more rapidly.

4- All obtained excitation phases are between $-\pi$ and $+\pi$. The input and output of the MLP are then scaled according to the Tansig function. The Tansig function will have its input from $+\pi$ to $-\pi$ and its output between +1 and -1. To be able to achieve this condition, linearization is applied to the inputs and outputs by division or multiplication by certain factors. The resulting inputs and outputs of the MLP are shown in Fig. 3.

The MLP is managed to produce accurate data with an error of $1.74e^{-5}$ within 25 epochs. Obtained results match with target values. A learning rate of 0.2 is used. Performances of the MLP are shown in Fig. 5.

![Performance result of one sector MLP.](image)

**Fig. 5- Performance result of one sector MLP.**

**Post-processing:** In the last step, simulations are performed using the appropriate MLP and remained excitation phases are obtained by symmetry. Results of this post-processing show that the technique is properly functioning for steering lobe and steering lobe with null interfering in any desired direction.

Table 1 shows the simulation results of the proposed approach when it is used with prescribed steering lobes, and steering and null design. It can be observed in Fig. 6,7 that this technique is capable of determining the phases for arrays main lobes and nulls imposed in the directions of interferences.
TABLE 1: Excitations For Different Steering Lobes And Interfering Nulling.

<table>
<thead>
<tr>
<th>Element</th>
<th>Phase (Deg)</th>
<th>Phase (Deg)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>225</td>
<td>60</td>
</tr>
<tr>
<td>2</td>
<td>35</td>
<td>85</td>
</tr>
<tr>
<td>3</td>
<td>155</td>
<td>50</td>
</tr>
<tr>
<td>4</td>
<td>280</td>
<td>20</td>
</tr>
<tr>
<td>5</td>
<td>80</td>
<td>340</td>
</tr>
<tr>
<td>6</td>
<td>205</td>
<td>310</td>
</tr>
<tr>
<td>7</td>
<td>325</td>
<td>275</td>
</tr>
<tr>
<td>8</td>
<td>135</td>
<td>300</td>
</tr>
</tbody>
</table>

Fig. 6. Steering lobe (@-50°) and interference nulling (@-10°)  
Fig. 7. Steering lobe (@-10°) and interference nulling (@ -30°)

4. Measurement
To verify the performance achievable using the proposed approach, numerical results are obtained for several arrays. A 16-element circular patch array (band 2.45 GHz), has been realized (Fig. 2) and tested for three cases of steered beams with null control, and three cases of multibeam. We used a feeding circuit for each realized case. A 1/8 microstrip divider for each sub-system (Fig. 7) is connected to the antennas by eight pieces of 50 ohm cables with variable lengths corresponding to synthesized phases for each sub-system. The measurement of the coefficient of reflection for each realized case has confirmed our hypothesis before. The different measurements are realized in the H-plane (Fig. 8 and Fig. 9).
5. Conclusion
The presented method is very practical for neural network implementation. The convergence and the generalisation of results are efficiently reached and the obtained “not trained” solutions are very accurate. The neural approach based on a (FF) Neural Network with BP shows good simulation results and allows a real time synthesis of desired steering beam with nulling interference directions.

The proposed approach in the paper can be treated as a strategy to be followed for any other future work in the same domain. Mainly it can be implemented for cylindrical antenna array. It is useful to mention that, based on the proposed approach; new research has been started lately for applying this neural networks approach in adaptative antennas.
References


