IEEE Access
Multidisciplinary | Rapid Review | Open Access Journal

Received July 13, 2020, accepted July 29, 2020, date of publication August 3, 2020, date of current version August 18, 2020.

Digital Object Identifier 10.1109/ACCESS.2020.3013880

An ANN-Based Synthesis Method for Nonuniform Linear Arrays Including Mutual Coupling Effects

YU GONG[1], SHAOQIU XIAO[1,2], (Member, IEEE), AND BING-ZHONG WANG[1], (Senior Member, IEEE)

[1]School of Physics, University of Electronic Science and Technology of China, Chengdu 610054, China
[2]School of Electronics and Information Technology, Sun Yat-sen University, Guangzhou 510006, China

Corresponding author: Shaoqiu Xiao (xiaoshaoqiu@uestc.edu.cn)

This work was supported in part by the National Natural Science Foundation of China under Grant 61731005, in part by the Pre-Research Foundation of National Defense under Grant 170441413086 and Grant 614240205070817, and in part by the Fundamental Research Funds for the Central Universities under Grant ZYGX2016Z008.

ABSTRACT This paper proposes an artificial neural network (ANN)-based synthesis method for nonuniform linear arrays with mutual coupling effects. The proposed method can simultaneously optimize the location distributions and excitations of the elements. As is well known, for nonuniform linear arrays, the mutual coupling effects on the active element patterns (AEPs) and passive S parameters vary with the location distribution of the elements. However, few papers have focused on how to describe the relationship between the effects of mutual coupling and the location distribution of the elements. In this paper, we use several parallel and independent ANNs to characterize the effects of mutual coupling on the AEPs and passive S parameters in nonuniform linear arrays with variable location distributions. At the same time, to relieve the curse of dimensionality, we make use of the idea of subarrays, which makes it possible for the proposed method to model nonuniform linear arrays with a large number of elements. After building accurate ANN models, we make use of the adaptive differential evolution algorithm JADE to search for the optimal location distribution and the excitations of the elements, to satisfy the requirements for the radiation pattern and the active S parameters of the antenna array. In view of the fact that the mutual coupling effects are accounted for in the synthesis process, the proposed method probably exhibits more practical value in antenna array designs than in other synthesis methods. The validity and efficiency of the proposed method are confirmed based on three examples.

INDEX TERMS Artificial neural networks, mutual coupling effects, nonuniform linear antenna arrays, subarray.

I. INTRODUCTION

With the development of radio technology, more and more requirements have been proposed for the performances of antenna arrays. As an important research direction of antenna arrays, the theory of array synthesis has attracted more and more attention because it can be applied to customizing the patterns and impedance matching. Therefore, many optimization methods based on analytical methods [1]–[6], stochastic algorithms [7]–[10], convex optimization [11]–[13] and hybrid methods [14]–[17] have been proposed for array synthesis problems in recent years. The above methods have shown excellent performances in many array synthesis problems; however, they all consider the elements in the arrays to be ideal models and the mutual coupling effects among the elements are ignored, which could cause two problems. The first is that under some circumstances, performance degradation of patterns could occur when the optimized results are applied to actual antenna arrays. Generally, this degradation will affect the side lobes close to the main lobe and become a primary disadvantage. The second is that in large-scale array antennas, the mutual coupling effects of the elements will enable the active S parameters to express certain differences from the passive S parameters, which would cause effects on the radiation characteristics of the antenna arrays [18].

Because of the importance of mutual coupling effects, some methods attempt to consider their influence in array synthesis problems. In [19], You *et al.* propose an efficient phase-only linear array synthesis method. In this method,

The associate editor coordinating the review of this manuscript and approving it for publication was Huanqing Wang.

the radiation pattern, including the mutual coupling effects, can be efficiently calculated by the fast Fourier transform. In [20], Liu *et al.* propose a virtual active element pattern (AEP) method to synthesize unequally spaced linear arrays with preset location distributions, and the proposed method can optimize the phases and amplitudes of excitations simultaneously. In [21], Chen *et al.* propose a numerical pattern synthesis algorithm that is integrated with a seeded region growing to synthesize the array in the presence of mutual coupling effects. All the above synthesis methods, which consider mutual coupling effects, are based on active element patterns (AEPs). In general, the shapes of the AEPs change with the number of surrounding elements and their positions. Therefore, the above methods can only synthesize the uniform arrays or nonuniform arrays with fixed positions of all elements. In [22], the authors propose an aperiodic array synthesis method with mutual coupling effects based on the calculation of the gradient of the field. However, the proposed method is suitable for only specific types of antenna elements and could be difficult to optimize the excitations of the elements, which greatly reduces the freedom of the optimization.

In recent years, nonuniform arrays have been widely applied because of their high target resolution, low sidelobe level (SLL) and low engineering cost [23]–[26]. Therefore, methods for the synthesis of nonuniform arrays have become one of the main research topics. However, as mentioned above, the methods that do not consider mutual coupling effects have obvious disadvantages, including the distortion of the active S parameters and the radiation pattern, while the methods that consider mutual coupling effects cannot optimize the location distribution and the excitations of the elements simultaneously. We are aware that the likelihood of adjusting the positions and excitations of the elements simultaneously provides more degrees of freedom than the cases in which the array geometry or array excitations are assigned [27].

Based on the above discussion, this paper aims at proposing a nonuniform linear array synthesis method that can synthesize nonuniform linear arrays with variable element positions while considering the mutual coupling effects, and the proposed method should be able to optimize the locations and excitations of the elements at the same time.

The core of the proposed synthesis method is to construct a model of the effects caused by mutual coupling on the AEPs and the passive S parameters under different location distributions of the elements. For nonuniform arrays, the AEPs of the elements are different due to their different array environment. In theory, we need to model the AEPs of the elements separately. However, this approach will make the model extremely complex and possibly extremely time-consuming. In fact, when the number of elements in the array is fixed, we can consider the AEPs of all elements as a whole. If the AEP of each element is represented by a row vector, then we can stitch the row vectors corresponding to all AEPs into a composite row vector. Then, we should find the nonlinear mapping relationship between the location distribution of the elements and the composite row vector. For the passive S parameters, similar to the AEPs, we can stitch all passive S parameters together to form a composite vector. Then, we should model the nonlinear relationship between the composite vector composed of all passive S parameters and the location distribution of the elements.

The artificial neural network (ANN) method is a type of nonlinear modeling technology, and it has been widely adopted to analyze and synthesize antennas in different ways [28]–[30]. Therefore, in our method, we make use of ANNs to model the relationship between the composite row vectors (constructed by the AEPs or passive S parameters) and the location distribution of the elements. It is well known that the number of samples required to train the ANNs will increase dramatically as the number of array elements increases. Therefore, the idea of subarrays is used to alleviate this problem. The details of the modeling process will be presented in section II.

Thus, through the ANNs, we can model the effects of mutual coupling on the AEPs and the passive S parameters in nonuniform linear arrays with variable location distributions of the elements. Then, if we know the location distribution and element excitations of the linear array, we can obtain the radiation pattern and the active S parameters of the antenna array, which are important performance parameters of the antenna array. Next, evolutionary algorithms (EAs) can be used to optimize the locations and excitations of the elements efficiently to make the radiation pattern and active S parameters of the array meet specific requirements. In our synthesis method, we take advantage of the adaptive differential evolution algorithm JADE, which is undoubtedly one of the most formidable stochastic parameter optimization algorithms at present to accomplish this task.

In terms of other parts of this paper, the detailed synthesis method for nonuniform linear arrays is described in section II. In addition, the proposed method is analyzed and assessed numerically in section III, which describes a series of experiments to further verify our method. In the end, there is a general conclusion of the whole paper in section IV.

II. METHOD DESCRIPTION

The proposed method is divided into two steps to apply based on the continuous sequences. In the first step, we model the nonuniform linear array using ANNs given the elements' structure, the number of elements and other constraints of the array. In the second step, we take advantage of JADE to perform an optimization of the locations and excitations of the array elements, to acquire the preset far-field pattern and active S parameters. In the following sections, we will elaborate on these two parts separately.

A. MODELING PROCESS

In view of the fact that ANNs have presented good performances in functional approximation and signal filtering operations as well as many other applications areas [31]–[33], they have attracted a large amount of attention from scholars,

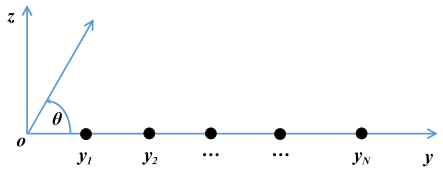

FIGURE 1. Geometry of an N-element nonuniform linear array.

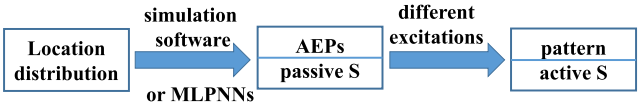

FIGURE 2. Relationship of antenna array parameters.

which resulted in many variants of the fundamental neural network [34]. In our method, we choose multi-layer perceptron neural networks (MLPNNs), which exhibit excellent and superior features, such as smaller training set demands, convenience in implementation and fast training speed, as our nonlinear mapping-modeling tool.

MLPNNs contain three types of layers: an input layer, an output layer and one or more hidden layers. Therefore, to model the linear arrays using MLPNNs, we must first define the input and output parameters of the networks. As mentioned in the introduction, we are mainly concerned with the far-field pattern and the active S parameters in the process of nonuniform linear array synthesis. Therefore, it is necessary for us to determine which variables are associated with these two types of parameters.

As shown in Fig. 1, given a Cartesian system *O(x, y,z)*, we consider a nonuniform linear array of N elements located at the positions $y_1, y_2, \cdots, y_N$ on the y-axis. The far-field pattern and active S parameters of this array can be written as in (1) and (2) [35].

$$f(u) = \sum_{n=1}^{N} w_n g_n(u) e^{j\beta y_n u} \tag{1}$$

$$S_n = \sum_{m=1}^{N} S_{nm} \frac{w_m}{w_n}, \quad n = 1, \cdots, N \tag{2}$$

where $\mathbf{w} = [w_1, \cdots, w_N]$ means the vector of complex excitations, and in the expression of $u = sin\theta$, θ denotes the angle from the y-axis to the z-axis. $\beta = 2\pi/\lambda$ uses the wavelength of λ, and $g_n(u)$ indicates the AEP of the nth element at the operating frequency (the location of the coordinate system's origin can be found at each element). The AEP of the nth element is the array pattern through exciting the nth element and connecting all other elements with the generator impedance. S_{nm} represents the passive S parameters, which can be written as

$$S_{nm} = V_n^- / V_m^+ \big|_{V_k^+ = 0 \; for \; k \neq m} \tag{3}$$

where V_m^+ and V_m^- indicate the incident and reflected voltage wave amplitude at the mth element. The relationship between the above parameters is shown in Fig. 2.

As shown in Fig. 2, there are two ways to acquire the AEPs and passive S parameters. In the process of obtaining training data for the ANNs, we can acquire the AEPs and passive S parameters directly through simulation software, such as HFSS and CST. Once we have obtained accurate models using the training data, we will obtain the AEPs and passive S parameters through the trained MLPNNs, which can greatly improve the calculation speed.

From Fig. 2, it can still be seen that the location distribution of the antenna array is the input parameter of the MLPNNs; the AEPs and passive S parameters are the output parameters of the MLPNNs. In view of the fact that these two types of output parameters are complex, we should model the real part and the imaginary part of these parameters separately. Therefore, we must train four parallel and independent MLPNNs in total. All MLPNNs have the same input parameter vector, as shown in (4).

$$\boldsymbol{y} = \begin{bmatrix} y_1 & y_2 & \cdots & y_N \end{bmatrix} \tag{4}$$

Assuming that the number of sampling points for all AEPs is M, then the AEP that corresponds to the nth element can be expressed as

$$AEP_n = \begin{bmatrix} r \cdot E_n^1 & r \cdot E_n^2 & \cdots & r \cdot E_n^M \end{bmatrix} \tag{5}$$

where r is a selected far-field radiation radius, and $E_n^m (m = 1, 2, \cdots, M)$ is the radiated electric field at the mth sampling point. Then, we can stitch together all AEPs to form a composite row vector:

$$\begin{bmatrix} AEP_1 & AEP_2 & \cdots & AEP_N \end{bmatrix} \tag{6}$$

The real part and the imaginary part of (6) are used to be the output parameter vectors of the MLPNNs that describe the real part and the imaginary part of the AEPs, respectively. Similar to the AEPs, the output parameter vectors of the MLPNNs correspond to the real part and the imaginary part of the passive S parameters are the real part and the imaginary part of the row vector shown in (7), respectively.

$$\begin{bmatrix} S_1 & S_2 & \cdots & S_N \end{bmatrix} \tag{7}$$

where $S_n = \begin{bmatrix} S_{n1} & S_{n2} & \cdots & S_{nN} \end{bmatrix}$, $n = 1, \cdots, N$.

As is known, with an increase in the number of array elements, the number of samples required to train the MLPNNs will increase sharply, which could make it impossible to construct the models. Therefore, for large arrays, we divide the arrays into several subarrays with an equal number of array elements, and all subarrays share the same MLPNNs, which indicates that the mutual coupling effects between the elements in each subarray are considered rigorously. However, the mutual coupling effects between the subarrays are not considered. It is widely known that a decrease in the mutual coupling effects between the elements is accompanied by an increase in the distances. Therefore, we can force the distances between the subarrays to be long enough that the mutual coupling effects between the subarrays can be neglected. In this way, regardless of how many elements are in the linear array, we can utilize the MLPNNs to model them.

Fig. 3 presents the form of the subarray partition, where the circle and rectangle represent an array element and

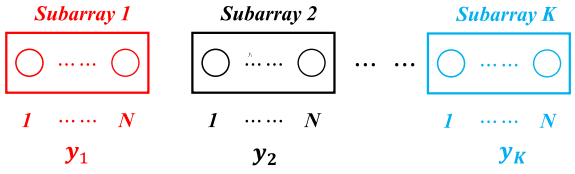

FIGURE 3. Large array partition diagram.

a subarray, respectively. In Fig. 3, $\boldsymbol{y}_k, (k = 1, \ldots, K)$ means the location distribution of the kth subarray, and K is the number of subarrays. The required distances between the subarrays are mainly related to the mutual coupling strengths between the elements and the magnitudes of the excitations. For most antenna arrays, this choice is regarded as an appropriate choice to establish the distances between the subarrays to $1 \sim 2$ wavelengths.

Although the distances between the subarrays appear to be large, the length of the whole array might be not larger than the results synthesized by the other methods. In other methods, with the purpose of reducing the influences of the mutual coupling effects, they usually set the distances between the elements to be not less than 0.5λ [36], [37]. Although the distances of the elements are preset long enough, for many of the actual arrays, the effects of mutual coupling remain significant, which will be shown in section III. However, in our method, we do not need to presuppose longer distances of the adjacent elements in the subarray, except for the geometric constraints. Therefore, although the distances between the subarrays are long, the distances between the elements in the subarrays can be shortened, which possibly makes the total length of the array be no longer than the others. In addition, for arbitrary array elements, the results synthesized by our method can be directly applied to a practical array, which is an advantage that many other methods do not have.

After acquiring accurate models for the AEPs and passive S parameters of the antenna arrays through the approach described above, we can use the trained MLPNNs to replace the full-wave simulation software to obtain the far-field patterns and active S parameters with different excitations and location distributions of the array elements, which will greatly accelerate the speed of the optimization process.

B. OPTIMIZATION PROCESS

In this section, we will elaborate on the process of optimization. To clarify this process, we first need to introduce JADE, which was chosen by us as the optimization tool because of its superior features, such as ease of use, parameter self-adaptation and fast convergence speed [38], [39].

Similar to nearly all evolutionary algorithms, JADE consists of four stages, namely, initialization, mutation, crossover and selection. Suppose that JADE searches for a global optimum point in a D-dimensional real parameter space R^D and we adopt the following notations for representing the ith parameter vectors of the population P_g at the gth generation:

$$\boldsymbol{x}_{g,i} = (x^1_{g,i}, x^2_{g,i}, \cdots, x^D_{g,i}) \tag{8}$$

To initialize the population, we sample NP parameter vectors uniformly in the initial search space, constrained by the preset minimum bound vector $\boldsymbol{x}_L$ and maximum bound vector $\boldsymbol{x}_U$.

For the mutation stage, we use the "DE/current-to-$p$best/1 with optional archive" mutation strategy [40] in this paper. We denote A_g as the set of archived inferior parameter vectors at the gth generation, and then, the mutation expression can be written as

$$\boldsymbol{v}_{g,i} = \boldsymbol{x}_{g,i} + F_{g,i} \cdot \left(\boldsymbol{x}^p_{g,best} - \boldsymbol{x}_{g,i}\right) + F_{g,i} \cdot (\boldsymbol{x}_{g,r1} - \tilde{\boldsymbol{x}}_{g,r2}) \tag{9}$$

where $\boldsymbol{x}^p_{g,best}$ is randomly chosen as one of the top 100p% parameter vectors at the current generation P_g with $p \in [0, 1]$, and $\boldsymbol{x}_{g,i}$ and $\boldsymbol{x}_{g,r1}$ are randomly selected from P_g, while $\tilde{\boldsymbol{x}}_{g,r2}$ is randomly chosen from the union $P_g \cup A_g$ of the current population P_g and the archive A_g. $F_{g,i}$ is the scale factor that controls the range at which the population evolves. The value of $F_{g,i}$ is independently generated according to a Cauchy distribution with the local parameter $\mu_{F,g}$ and scale parameter 0.1. If the generated $F_{g,i} \leq 0$ *or* $F_{g,i} > 1$, we should regenerate a new $F_{g,i}$. The location parameter $\mu_{F,0}$ is initialized to be 0.5 and then updated after each generation as

$$\mu_{F,g} = (1 - c) \cdot \mu_{F,g-1} + c \cdot \frac{\sum_{F_{g-1,i} \in S_{F_{g-1}}} {F_{g-1,i}}^2}{\sum_{F_{g-1,i} \in S_{F_{g-1}}} F_{g-1,i}} \tag{10}$$

where c is a positive constant between 0 and 1, and $S_{F_{g-1}}$ is the set of all successful $F_{g-1,i}$ in the selection process of (13). The operation of the archive is very simple. First, the archive A_0 is initiated to be empty. Then, the parent parameter vector that failed in the selection process of (13) is added to the archive after each generation. Once the archive size exceeds NP, some parameter vectors are randomly removed from the archive to keep the size of the archive at NP.

In the crossover stage, we cross each parameter vector $\boldsymbol{x}_{g,i}$ with the mutant vector $\boldsymbol{v}_{g,i}$ to form the trial vector $\boldsymbol{u}_{g,i}$:

$$u^j_{g,i} = \begin{cases} v^j_{g,i} & if\ (rand_{i,j}[0, 1] \leq CR_{g,i}\ or\ j = j_{rand}) \\ x^j_{g,i} & otherwise \end{cases} \tag{11}$$

where the superscript ^{j} means the jth component of the corresponding vector, and the j_{rand} is the randomly chosen index to ensure that the trial vector $\boldsymbol{u}_{g,i}$ does not duplicate $\boldsymbol{x}_{g,i}$. The crossover rate $CR_{g,i}$ controls the fraction of parameter vectors that are copied from the mutant, and it is independently generated according to a Gaussian distribution of mean $\mu_{CR,g}$ and standard deviation 0.1; and, it is truncated to [0,1]. The $\mu_{CR,0}$ is initialized to be 0.5 and then updated after each generation, as follows:

$$\mu_{CR,g} = (1 - c) \cdot \mu_{CR,g-1} + c \cdot mean(S_{CR_{g-1}}) \tag{12}$$

where $S_{CR_{g-1}}$ is the set of all successful $CR_{g-1,i}$ in the selection process of (13), and $mean(\cdot)$ is the usual arithmetic mean.

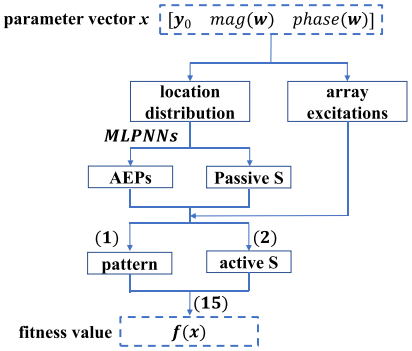

FIGURE 4. Process of computing the fitness value.

After the crossover stage, we will use (13) to acquire the next generation of parameter vectors:

$$\boldsymbol{x}_{g+1,i} = \begin{cases} \boldsymbol{u}_{g,i} & if\ f(\boldsymbol{u}_{g,i}) \le f(\boldsymbol{x}_{g,i}) \\ \boldsymbol{x}_{g,i} & otherwise \end{cases} \tag{13}$$

which means that if the trial vector $\boldsymbol{u}_{g,i}$ acquires an equal or lower fitness value, it will replace the target vector $\boldsymbol{x}_{g,i}$ in the next generation; otherwise, the target vector $\boldsymbol{x}_{g,i}$ retains its place in the population. There are two termination conditions of the optimization process. The first condition is that once the fitness value of the best individual is no greater than a preset value ϵ, the optimization process will be terminated. The other condition sets a maximum on the number of population iterations. Once the number of iterations exceeds G, the optimization process is terminated. The specific pseudo code of JADE can be found in [40].

To solve the array parameter optimization problem through JADE, we need to define only the optimization parameter vector $\boldsymbol{x}$ and fitness function $f(\boldsymbol{x})$. As noted in the introduction, the optimization variables include the locations of the array elements as well as the magnitudes and phases of the array element excitations. Thus, the optimization parameter vector can be written as

$$\boldsymbol{x} = [\boldsymbol{y}_0 \quad mag(\boldsymbol{w}) \quad phase(\boldsymbol{w})] \tag{14}$$

where $\boldsymbol{y}_0 = [y_1 y_2 y_3 \cdots y_N]$, and mag($\boldsymbol{w}$) and phase($\boldsymbol{w}$) mean the magnitudes and phases of the array excitations, respectively. Then, the loss function can be written as

$$f(\boldsymbol{x}) = f_1(\boldsymbol{x}) + \xi \cdot f_2(\boldsymbol{x}) \tag{15}$$

where $f_1(\boldsymbol{x})$ and $f_2(\boldsymbol{x})$ characterize the loss function of the active S parameters and the radiation pattern, respectively, and ξ is a positive number that is used to weigh the relationship between the two loss functions. The calculation process of $f(\boldsymbol{x})$ is shown in Fig. 4. For a given parameter vector $\boldsymbol{x}$, we can use the location distribution $\boldsymbol{y}_0$ of $\boldsymbol{x}$ and the trained MLPNNs to calculate the AEPs and passive S parameters of the array. Then, we can make use of (1) and (2) to compute the radiation pattern and active S parameters for the given array excitations of $\boldsymbol{x}$. After obtaining the two antenna properties that we are concerned with, we can easily calculate the cost value according to the specific definitions of the loss function $f(\boldsymbol{x})$, which will be shown in section III.

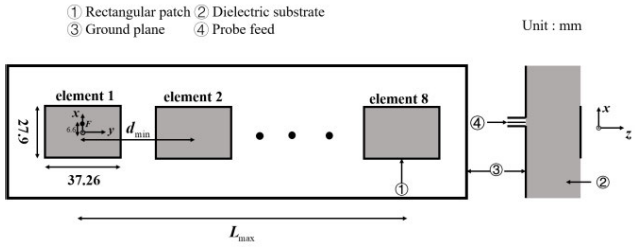

FIGURE 5. Geometry of an 8-element rectangular patch antenna array. (The inner and outer radius of the coaxial line are 0.6mm and 1.5mm.)

III. EXAMPLES

A. TAYLOR DISTRIBUTION PATTERN

For the first example, we synthesize an eight nonuniform linear array with an expected pattern that conforms to the Taylor distribution with a = 1.75λ, SLL = −20 dB and $\bar{n} = 8$ [41], and we constrain the active S parameters to be less than −10 dB. In this example, we assume that each element is the rectangular patch antenna and the material of the dielectric substrate is FR4 epoxy. The element is pre-designed with a working frequency of 2.45GHz, and we group the array in the H plane, as shown in Fig. 5.

1) TRAINING MODEL

As described in section II, it is necessary for us to train an accurate model of the array. Since the array should not be infinitely long, the length of the array is limited to less than $L_{max} = 3.5\lambda$, which means that the longest distance between the same points of the first element and the last element in Fig. 5 is 3.5λ. In addition, due to the physical size limitations, the minimum distance between the elements is established to be $d_{min} = 40\ mm$, which is slightly larger than the length of the element, 37.26 mm. The resolution of the pattern is set to 1 degree, and we merely model the co-polarization pattern of the H plane in the upper half space, i.e., the upper half of the y-o-z plane, as shown in Fig. 5. Therefore, the number of sample points of the AEP for each element is 181. Since the pattern to be synthesized is symmetric, it can be assumed that the elements are symmetric about the origin and the symmetrical elements have the same excitations.

We make use of the ANSYS HFSS to acquire 1000 samples; 60% of the samples are adopted to train the MLPNNs, and the remaining samples are used for validating and testing of the model. In addition, we apply the Adam optimizer and exponential attenuation learning rate to accelerate the convergence and improve the prediction accuracy. Considering the zeros of the parameters, we define (16) as our loss function.

$$\mathrm{LF} = \frac{1}{K}\sum_{k=1}^{K} \frac{\|\boldsymbol{y}_k - \widehat{\boldsymbol{y}}_k\|_2}{\|\boldsymbol{y}_k\|_2} \times 100\% \tag{16}$$

where $\boldsymbol{y}_k$ and $\widehat{\boldsymbol{y}}_k$ denote the simulation and forecasted values, respectively, of the kth sample, K is the total number of the samples and $\|\boldsymbol{y}\|_2$ is the Euclidian norm of vector $\boldsymbol{y}$. If we replace $\widehat{\boldsymbol{y}}_k$ in (16) with the values that correspond to the isolated element, LF will represent the strength of the

TABLE 1. Modeling Results of the 8-Element Patch Antenna Array.

AEP	Train	Validation	Test	Isolated
Imaginary	2.05%	2.39%	2.33%	21.31%
Real	2.23%	2.59%	2.49%	21.54%
S-Parameter	Train	Validation	Test	
Imaginary	3.77%	4.59%	4.69%	
Real	4.32%	5.03%	5.09%	

mutual coupling in the array environment for the plane that we care about. In extreme cases, if there is no mutual coupling between elements, the strength of the mutual coupling will be 0. In theory, both the passive S parameters and the AEPs can represent the strength of the mutual coupling; however, the passive S parameters can easily cause large errors because of their small values. Therefore, the mutual coupling strength that corresponds to the AEPs is taken as the criterion in this paper.

The prediction accuracy of the MLPNNs for different parameters and the mutual coupling strength are shown in Table 1. In fact, the mutual coupling strength also represents the error between the AEPs of the isolated elements and the AEPs in the antenna array environment. Therefore, the error between the real AEPs and predicted AEPS can be reduced by an order of magnitude when we use the results of the MLPNNs to replace the results of the isolated elements for this example. To visually display the above effects, we randomly select a sample from the test set to indicate the modeling results of the AEPs, shown in Fig. 6. This figure shows that the phase of the first element changes more dramatically than that of the fourth element. The reason for this phenomenon is that the phase center is chosen at the center of the linear array.

2) OPTIMIZATION

To optimize with JADE, we must define the optimization parameter vector and fitness function first. In this example, the location optimization variables can be written as

$$\boldsymbol{y}_0 = [y_1\ y_2\ y_3\ y_4] \tag{17}$$

The excitation optimization variables can be written as

$$[mag\ (\boldsymbol{w}) \quad phase(\boldsymbol{w})] \tag{18}$$

where

$$\mathbf{w} = [w_1, \cdots, w_4] \tag{19}$$

and the maximum amplitude ratio of w_n is set to be 5. The loss function of the active S parameters is defined as

$$f_1\ (\boldsymbol{x}) = \sum_{n=1}^{4} g(S_n - 0.25) \tag{20}$$

where

$$g\ (a) = \begin{cases} a, & a > 0 \\ 0, & a \le 0 \end{cases} \tag{21}$$

Note that in (20), we choose 0.25 as the upper bound for the active S parameters, which is more stringent than -10 dB. The reason for this selection is to ensure that the optimized results can meet the requirements of the active S parameters. We denote the target pattern and the predicted pattern as $T(\theta)$ and $P(\theta)$, respectively. Then, the loss function of the radiation pattern can be written as

$$f_2\ (\boldsymbol{x}) = \sum_{\theta=-90^\circ}^{0^\circ} |T\ (\theta) - P(\theta)| \tag{22}$$

In this paper, the parameter ξ, which is used to balance the two loss functions, is set to 2. The parameter settings of JADE in the optimization process are similar to those in [40]. We set $NP = 100$, $p = 0.05$ and $c = 0.1$ in this example. The parameters of the termination conditions are set to be $\epsilon = 10^{-8}$ and $G = 500$. To reduce the influence of randomness, each example is repeated 20 times independently. Fig. 7 shows the average, the worst and the optimal convergence curves of this problem. Table 2 presents the optimized values for the weights and locations of the elements.

In [41], each antenna element is supposed to be an isotropic ideal point source and the optimal solution is obtained by analytic methods. If we want to acquire the Taylor pattern described above, the total length of the array should be 3.5λ through the formula of [41].

Fig. 8(a) shows that the predicted pattern by our method, the simulation result acquired by HFSS and the target pattern

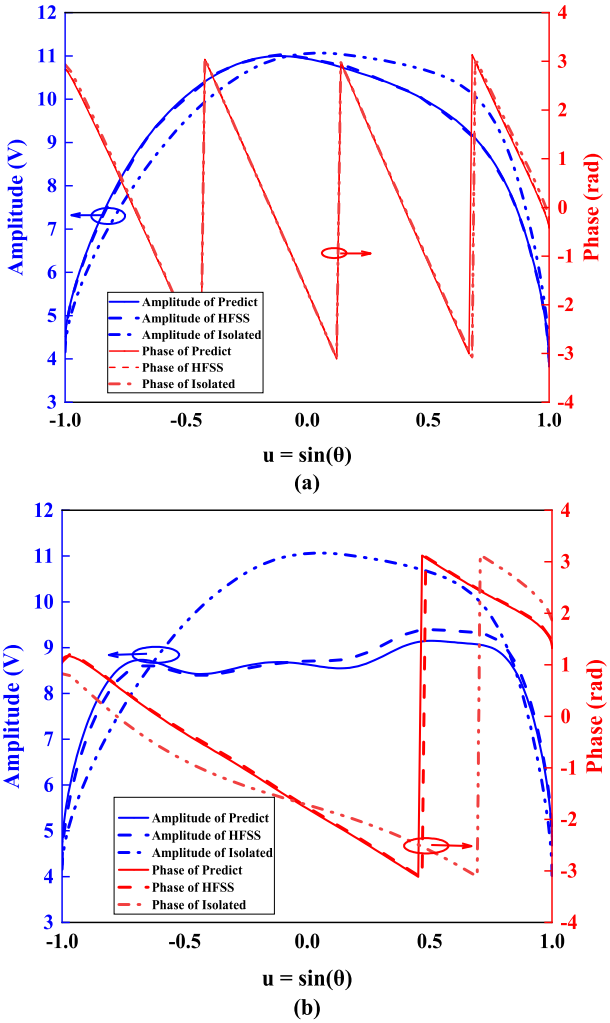

FIGURE 6. AEPs of the 8-element patch antenna array for the example A. (a) The first element; (b) The fourth element.

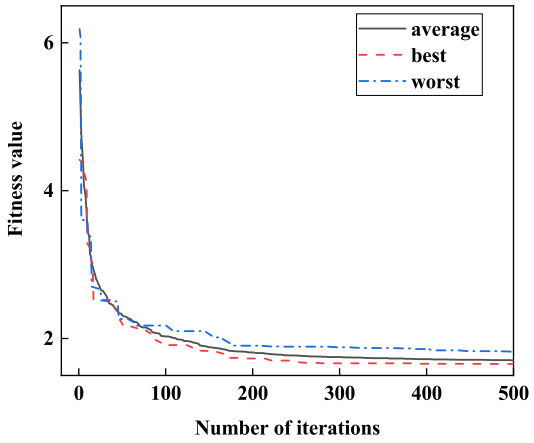

FIGURE 7. Convergence curves for example A.

TABLE 2. Optimized Weights and Locations for The Taylor Pattern.

Element No.	Location (λ)	Amplitude	Phase (rad)
1	-1.48	1.00	-0.28
2	-1.10	1.15	0.37
3	-0.61	1.35	-0.15
4	-0.23	1.54	0.13

are in good agreement. However, when the results synthesized in [41] are applied to the practical array, the pattern is obviously deformed. In addition, through our method, the total length of the array can be reduced to 2.96λ. However, we must point out that the gain of the array also decreases from 12.07 dB to 11.42 dB.

Fig. 8(b) compares the predicted active S parameters with the simulation results. All active S parameters are less than −10 dB, which satisfies the optimization objectives preset above.

B. SQUARE-COSECANT BEAM PATTERN

To verify the ability of the proposed method to address asymmetric patterns, the square-cosecant beam pattern mask proposed in [42] is applied as our second example. In addition, we also constrain the active S parameters to be less than −10 dB, which meets the requirements of many applications. We propose an assumption that the implementation of this array is performed under the circumstance of 13 patch elements working at 10 GHz to realize the demands above. Fig. 9 presents the whole structural frame of the array.

1) TRAINING MODEL

Similar to example A, the length of the array is limited to less than $L_{max} = 7.5\lambda$, and the minimum distance between the same points of adjacent elements is limited to $d_{min} = 0.5\lambda$, which is slightly larger than the width of the element, $\frac{13}{30}\lambda$. We define the location of the first element as 0, and thus, the input layer of the MLPNNs is $[y_2, y_3, \cdots y_{13}]$, where y_n is the location of the nth element. Therefore, the number of input parameters is 12. The resolution of the pattern is set to 1 degree, and we merely model the co-polarization pattern of the upper half of the H plane, i.e., the upper half of the

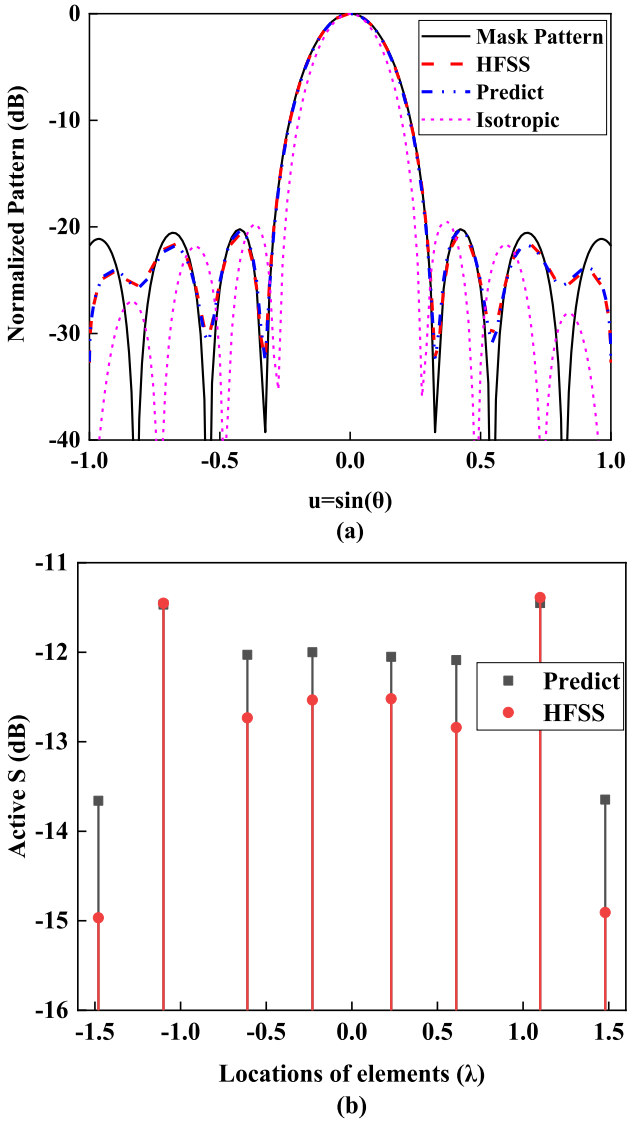

FIGURE 8. Optimized results of square-cosecant beam pattern synthesis. (a) Normalized patterns; (b) Active S parameters.

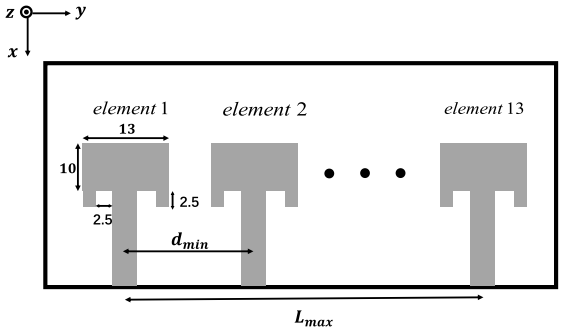

FIGURE 9. Geometry of a 13-element microstrip antenna array for example B. (A substrate with $\varepsilon_r = 2.2$ and a thickness of 1.57 mm is used).

y-o-z plane, as shown in Fig. 9. Therefore, the number of sample points of the AEP for each element is 181.

We make use of the ANSYS HFSS to acquire 1200 samples, and three types of data sets are utilized,

TABLE 3. Modeling Results of the 13-Element Patch Antenna Array.

AEP	Train	Validation	Test	Isolated
Imaginary	1.05%	1.34%	1.25%	11.64%
Real	1.36%	0.96%	1.21%	12.51%
S-Parameter	Train	Validation	Test	
Imaginary	4.69%	5.82%	5.71%	
Real	4.59%	5.43%	5.75%	

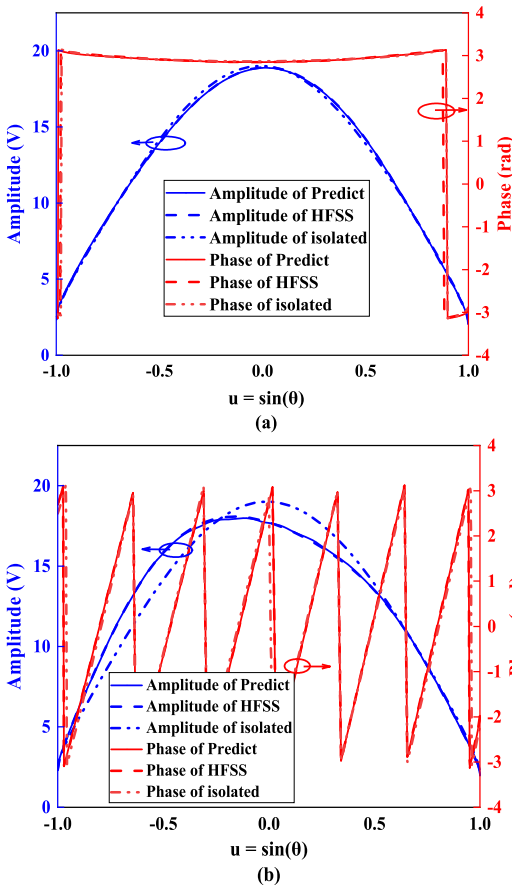

FIGURE 10. AEPs of the 13-element patch antenna array for example B. (a) The first element; (b) The sixth element.

namely, the training set, validation set and test set. A total of 800 samples are adopted to train the MLPNNs and the remaining samples are used for validating and testing the MLPNNs.

The modeling results of the different parameters and the mutual coupling strength are shown in Table 3. From Table 3, it can be seen that the mutual coupling strengths that correspond to the real part and the imaginary part of the AEPs are 12.51% and 11.64%, respectively. With MLPNNs, we can reduce the prediction errors of the AEPs by almost an order of magnitude, to 1.21% and 1.25%. To visualize the impact of the error, we randomly select a sample in the test set to present the modeling results of the AEPs, as shown in Fig. 10. The origin is selected at the first antenna element for both Fig. 10 (a) and (b), therefore, the phase curves in Fig.10 (a) and (b) are quite different. Although the deviation in the AEPs caused by mutual coupling appear to be very small, it will be shown afterward that the far-field pattern and the active S parameters are obviously distorted if we ignore the mutual coupling effects.

2) OPTIMIZATION

Since the position of the first element is fixed to be 0, the location optimization variables in this example can be written as

$$\boldsymbol{y}_0 = [y_2\ y_3\ \cdots\ y_{13}] \tag{23}$$

The excitation optimization variables can be written as

$$[mag\ (\boldsymbol{w}) \quad phase(\boldsymbol{w})] \tag{24}$$

where

$$\mathbf{w} = [w_1, \cdots, w_{13}] \tag{25}$$

The loss function of the active S parameters can be defined as

$$f_1\ (\boldsymbol{x}) = \sum\nolimits_{n=1}^{13} g(S_n - 0.25) \tag{26}$$

The target power pattern has a cosec-squared main lobe $T\ (\theta) = csc^2\ (\theta) \cos\ (\theta)$ for $\theta \in [10°, 50°]$, and the sidelobe level (SLL) is less than -20 dB and -30 dB for $\theta \in [-90°, -24°] \cup [53°, 90°]$ and $[-23°, -1°]$, respectively. Therefore, the upper and lower bound $T^u(\theta)$ and $T^l(\theta)$ of the pattern can be written as

$$T^u\ (\theta) = \begin{cases} 1.05 * 10log10\ (T\ (\theta))\ dB, \\ \qquad\qquad \theta \in [10°, 50°] \\ -20dB, \quad \theta \in [-90°, -24°] \cup [53°, 90°] \\ -30dB, \quad \theta \in [-23°, -1°] \\ 0dB. \qquad others \end{cases} \tag{27}$$

$$T^l\ (\theta) = \begin{cases} 0.95 * 10log10\ (T\ (\theta))\ dB, & \theta \in [10°, 50°] \\ -\infty, & others \end{cases} \tag{28}$$

Then, the loss function of radiation pattern can be written as

$$f_2\ (\boldsymbol{x}) = \sum\nolimits_{\theta=-90°}^{90°} g(\mathrm{P}\ (\theta) - T^u\ (\theta) + g(T^l\ (\theta) - P(\theta)) \tag{29}$$

Since there are more optimization parameters in this example than in example A, the population size *NP* of JADE is changed to 200, and the other parameter settings of JADE are the same as in example A. Fig. 11 shows the average, the worst and the optimal convergence curves of example B. Table 4 presents the optimization results for the weights and locations of the elements.

To show the effects of mutual coupling on the radiation pattern and active S parameters, we compare the results of our method with those of [42]. In [42], they use hybrid methods to synthesize 13 isotropic elements, and the mutual coupling effects are not considered. When the result is applied to the actual array, the pattern is distorted, as shown in Fig. 12(a). In contrast, our method achieves the target pattern very well. In addition, the worst active S parameter of [42] is -5.99 dB,

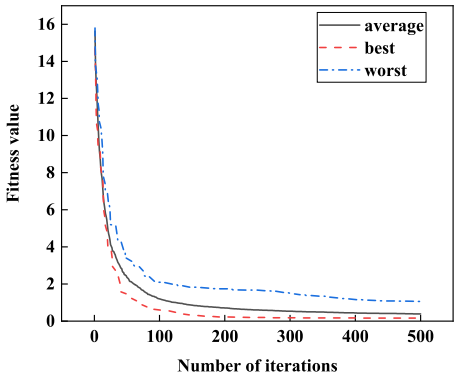

FIGURE 11. Convergence curves for example B.

TABLE 4. Optimized Weights and Locations for the Square-Cosecant Pattern.

Element No.	Location (λ)	Amplitude	Phase (rad)
1	0	2.65	2.55
2	0.52	3.53	1.04
3	1.13	4.74	0.36
4	1.63	4.72	-0.60
5	2.16	3.81	-1.32
6	2.67	2.57	-1.87
7	3.55	3.88	-1.83
8	4.07	2.79	3.11
9	4.79	1.12	-2.73
10	5.33	1.61	-2.88
11	5.84	1.85	2.14
12	6.97	1	2.61
13	7.47	1.01	1.05

which is clearly unacceptable in many applications. However, Fig. 12(b) reflects that through our method, all the active S parameters are below -10 dB, which achieves the desired goal.

C. FLAT-TOP PATTERN

As the last example, the synthesis of a 26 nonuniform linear array is performed to realize a flat-top pattern for presenting the practical effects for the idea of subarrays. The considered pattern mask exhibits a maximum ripple that is equal to $\pm 0.5\ dB$ for $|\theta| \leq 7°$ and a -25 dB peak sidelobe level for $|\theta| \geq 11°$. In addition, we constrain the active S parameters to be less than -10 dB for all elements.

We assume that the array element used in this example is the same as the element in example B. Since the number of elements in this example is large, the number of samples required to model it directly is very large, and the time for each sample to be simulated is also long. Thus, it is unreasonable to model this array directly. Therefore, we divide the array into 2 subarrays with the same number of elements to model it, which means that the trained MLPNNs in example B can be applied directly to this example. To reduce the mutual coupling effects between the subarrays, the distance between subarrays is set to 1λ.

Since the pattern to be synthesized in this example is symmetric about the origin, it can be assumed that the position distributions in the two subarrays are symmetric about the origin, and the symmetric elements are equally excited.

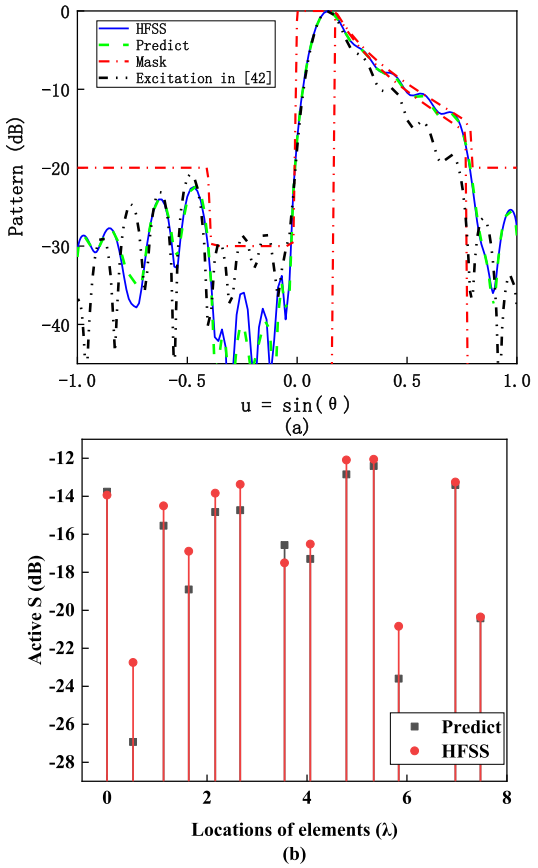

FIGURE 12. Optimization results of the square-cosecant beam pattern synthesis. (a) Normalized patterns; (b) Active S parameters.

Then, the optimization parameter vectors can be written as

$$\boldsymbol{x} = [\boldsymbol{y_0} \quad mag(\boldsymbol{w}) \quad phase(\boldsymbol{w})] \tag{30}$$

where

$$\boldsymbol{y_0} = [y_1\ y_2\ \cdots\ y_{13}] \tag{31}$$

$$\boldsymbol{w} = [w_1\ w_2\ \cdots\ w_{13}] \tag{32}$$

Therefore, the loss function of the active S parameters can be defined as

$$f_1(\boldsymbol{x}) = \sum_{n=1}^{13} g(S_n - 0.25) \tag{33}$$

The upper bound and lower bound of the pattern are

$$T^u(\theta) = \begin{cases} -25dB, & |\theta| \geq 11° \\ 0, & others \end{cases} \tag{34}$$

$$T^l(\theta) = \begin{cases} -1dB, & |\theta| \leq 7° \\ -\infty, & others \end{cases} \tag{35}$$

Then, the loss function of the target radiation pattern can be written as

$$f_2(\boldsymbol{x}) = \sum_{\theta=-90°}^{0°} g(\mathrm{P}(\theta) - T^u(\theta) + g(T^l(\theta) - P(\theta)) \tag{36}$$

Since the number of optimization variables in this example is similar to that of example B, the parameter settings are

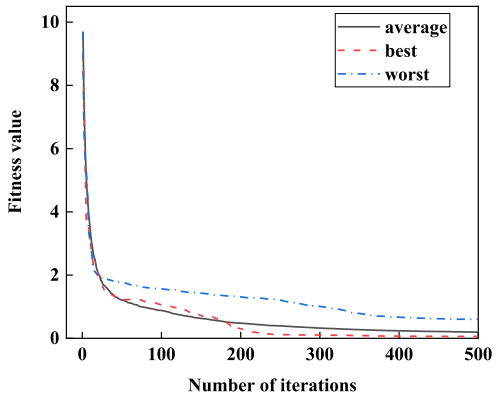

FIGURE 13. Convergence curves for example C.

TABLE 5. Optimized Weights and Locations for the Flat-Top Pattern.

Element No.	Location (λ)	Amplitude	Phase (rad)
1	-7.98	1.00	-0.76
2	-7.34	1.00	-1.30
3	-6.73	1.14	-1.71
4	-6.22	1.32	-1.92
5	-5.64	2.10	-2.44
6	-5.03	2.38	-2.66
7	-4.53	2.82	-2.86
8	-3.95	3.31	-3.11
9	-3.45	2.72	-3.14
10	-2.90	3.38	2.72
11	-2.16	3.28	2.16
12	-1.34	3.79	1.53
13	-0.50	4.99	1.19

the same as those of example B. Fig. 13 shows the average, the worst and the optimal convergence curves of example C, and Table 5 shows the optimization results for the weights and locations of the elements.

The optimized pattern and active S parameters are presented in Fig. 14. From Fig. 14(a), it can be concluded that both the predicted pattern and the simulation results are in good agreement with each other, which means that mutual coupling between subarrays has little effect on the radiation pattern when there is a slightly increase in the distance between the subarrays. In addition, Fig. 14(b) shows a comparison between the predicted active S parameters and the simulation results. Based on this figure, it can be known that all active S parameters are less than −10 dB, which satisfies the preset optimization objective.

D. DISCUSSION ON THE RUNNING TIME

In our method, the time required for the synthesis process mainly comes from three parts. The first part is the time needed to obtain the training samples by the simulation software, such as HFSS. For the three examples in this paper, it takes approximately 15 minutes to acquire a sample. For each example, we need approximately 1000 samples, which is likely to take a long time. However, all samples for each example can be obtained in parallel, which will greatly reduce the time required to obtain the samples. We can divide the training samples into several groups. For all groups, we can use several HFSS terminals to compute them in parallel.

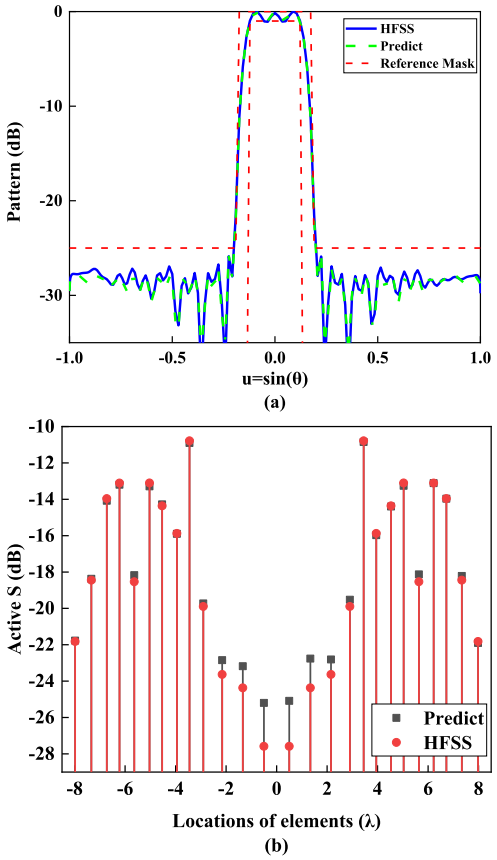

FIGURE 14. Optimization results for flat-top beam pattern synthesis. (a) Normalized patterns; (b) Active S parameters.

The second part of the running time comes from training the MLPNNs. In this paper, we make use of the tensorflow-gpu to train all MLPNNs. It takes approximately 30 minutes to train each MLPNN. If using the CPU to train the MLPNNs, the training time usually increases by 5 to 10 times compared with the GPU. The third part of the required time comes from the optimization process of the DE algorithm. We use tensorflow-gpu to calculate the fitness values for all individuals of each generation in parallel, and the time required for the optimization process is approximately 10 minutes.

E. MEASUREMENT

To verify the correctness of the simulation, we processed and tested the array in example A. The physical picture of the antenna is shown in Fig. 15. The prototype was tested in the anechoic chamber at the University of Electronic Science and Technology of China. Measurements were made according to the unit excitation active element pattern method [35], where individual ports are measured sequentially, with all others terminated at a matched load. Through this method, the normalized array pattern and active S parameters can be readily synthesized in a post-processing step. Fig. 16 shows the normalized array pattern and active S parameters that are obtained from the measurement.

From Fig. 16(a), it can be seen that the test pattern and the simulation pattern agree well in the main lobe.

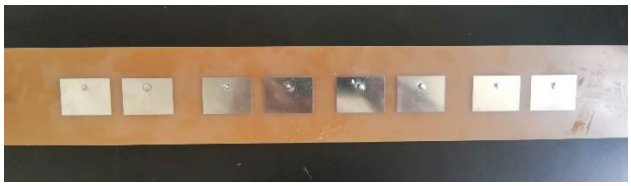

FIGURE 15. Physical picture of the antenna in example A.

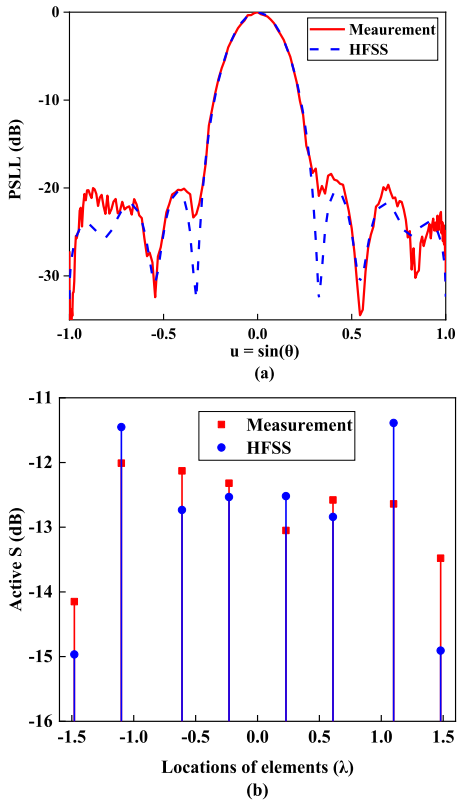

FIGURE 16. Comparison of simulation and test results of the array in the example A. (a) Normalized patterns; (b) Active S parameters.

However, there is some deviation near the side lobe. There are three main reasons for this error. The first is the error caused by the machining accuracy. The second is the error caused by the noise of the test environment. The third error source is the numerical error of the simulation process. The deviations in the active S parameters shown in Fig. 16(b) are also mainly caused by the above three factors.

IV. CONCLUSION

In this paper, MLPNNs are used to model the effects of mutual coupling on the AEPs and passive S parameters in nonuniform linear arrays with variable location distributions. Then, the MLPNN models are combined with the JADE algorithm to synthesize the nonuniform linear arrays. In theory, the method can synthesize nonuniform linear arrays with an arbitrary number of elements. However, for large arrays, the method of subarray division can result in a longer array length. The applications of three numerical examples lie in manifesting the effectiveness of this method. In addition, one example is machined and tested to prove the correctness of the simulation.

We should emphasize that the method proposed in this paper is computationally expensive. Therefore, the proposed method is more suitable for the scene that requires high accuracy of synthesized pattern. In addition, the research in this paper is limited to the linear array working on a single frequency point. In the future, we will attempt to extend the proposed method to the problems of planar array synthesis and broadband array synthesis.

REFERENCES

[1] B. P. Kumar and G. R. Branner, "Generalized analytical technique for the synthesis of unequally spaced arrays with linear, planar, cylindrical or spherical geometry," *IEEE Trans. Antennas Propag.*, vol. 53, no. 2, pp. 621–634, Feb. 2005.

[2] D. Caratelli and M. C. Viganó, "Analytical synthesis technique for linear uniform-amplitude sparse arrays," *Radio Sci.*, vol. 46, no. 4, pp. 1–6, Aug. 2011.

[3] Y. Liu, Z. Nie, and Q. H. Liu, "Reducing the number of elements in a linear antenna array by the matrix pencil method," *IEEE Trans. Antennas Propag.*, vol. 56, no. 9, pp. 2955–2962, Sep. 2008.

[4] Y. Liu, Q. H. Liu, and Z. Nie, "Reducing the number of elements in multiple-pattern linear arrays by the extended matrix pencil methods," *IEEE Trans. Antennas Propag.*, vol. 62, no. 2, pp. 652–660, Feb. 2014.

[5] M. Y. Zheng, K. S. Chen, H. G. Wu, and X. P. Liu, "Sparse planar array synthesis using matrix enhancement and matrix pencil," *Int. J. Antenn. Propog.*, vol. 2013, no. 2, pp. 245–253. 2013.

[6] P. Gu, G. Wang, Z. Fan, and R. Chen, "An efficient approach for the synthesis of large sparse planar array," *IEEE Trans. Antennas Propag.*, vol. 67, no. 12, pp. 7320–7330, Dec. 2019.

[7] S. K. Goudos, K. Siakavara, T. Samaras, E. E. Vafiadis, and J. N. Sahalos, "Sparse linear array synthesis with multiple constraints using differential evolution with strategy adaptation," *IEEE Antennas Wireless Propag. Lett.*, vol. 10, pp. 670–673, Jun. 2011.

[8] K. Chen, Z. He, and C. Han, "A modified real GA for the sparse linear array synthesis with multiple constraints," *IEEE Trans. Antennas Propag.*, vol. 54, no. 7, pp. 2169–2173, Jul. 2006.

[9] S. K. Goudos, V. Moysiadou, T. Samaras, K. Siakavara, and J. N. Sahalos, "Application of a comprehensive learning particle swarm optimizer to unequally spaced linear array synthesis with sidelobe level suppression and null control," *IEEE Antennas Wireless Propag. Lett.*, vol. 9, pp. 125–129, Mar. 2010.

[10] Y. Chen, S. Yang, and Z. Nie, "The application of a modified differential evolution strategy to some array pattern synthesis problems," *IEEE Trans. Antennas Propag.*, vol. 56, no. 7, pp. 1919–1927, Jul. 2008.

[11] S. E. Nai, W. Ser, Z. L. Yu, and H. Chen, "Beam pattern synthesis for linear and planar arrays with antenna selection by convex optimization," *IEEE Trans. Antennas Propag.*, vol. 58, no. 12, pp. 3923–3930, Dec. 2010.

[12] M. Li, Y. Li, X. Wang, J. Dong, and Y. Chang, "Optimal polarised pattern synthesis of wideband arrays via convex optimisation," *IET Microw., Antennas Propag.*, vol. 7, no. 15, pp. 1228–1237, Dec. 2013.

[13] Y. Liu, L. Zhang, L. Ye, Z. Nie, and Q. Huo Liu, "Synthesis of sparse arrays with Frequency-Invariant-Focused beam patterns under accurate sidelobe control by iterative second-order cone programming," *IEEE Trans. Antennas Propag.*, vol. 63, no. 12, pp. 5826–5832, Dec. 2015.

[14] S. K. Mahto and A. Choubey, "A novel hybrid IWO/WDO algorithm for interference minimization of uniformly excited linear sparse array by position-only control," *IEEE Antennas Wireless Propag. Lett.*, vol. 15, pp. 250–254, Feb. 2016.

[15] A. H. Hussein, H. H. Abdullah, A. M. Salem, S. Khamis, and M. Nasr, "Optimum design of linear antenna arrays using a hybrid MoM/GA algorithm," *IEEE Antennas Wireless Propag. Lett.*, vol. 10, pp. 1232–1235, Oct. 2011.

[16] W. T. Li, X. W. Shi, Y. Q. Hei, S. F. Liu, and J. Zhu, "A hybrid optimization method and its application for conformal array pattern synthesis," *IEEE Trans. Antennas Propag.*, vol. 58, no. 10, pp. 3401–3406, Oct. 2010.

[17] C. Cui, W. T. Li, X. T. Ye, and X. W. Shi, "Hybrid genetic algorithm and modified iterative Fourier transform algorithm for large thinned array synthesis," *IEEE Antennas Wireless Propag. Lett.*, vol. 16, pp. 2150–2154, May 2017.

[18] C. Zhang, Q. Lai, and C. Gao, "Measurement of active S-parameters on array antenna using directional couplers," in *Proc. IEEE APMC*, Kuala Lumpar, Malaysia, Nov. 2017, pp. 1167–1170.
[19] P. You, Q. H. Liu, L. Zhang, X. Huang, and Y. Liu, "Efficient phase-only linear array synthesis including coupling effect by GA-FFT based on least-square active element pattern expansion method," *Electron. Lett.*, vol. 51, no. 10, pp. 791–792, May 2015.
[20] Y. Liu, X. Huang, K. D. Xu, Z. Song, S. Yang, and Q. H. Liu, "Pattern synthesis of unequally spaced linear arrays including mutual coupling using iterative FFT via virtual active element pattern expansion," *IEEE Trans. Antennas Propag.*, vol. 65, no. 8, pp. 3950–3958, Aug. 2017.
[21] S. Chen, P. Qin, Y. J. Guo, Y. Liu, and P. You, "Array pattern synthesis using numerical pattern synthesis method including mutual coupling," in *Proc. APS/URSI*, San Diego, CA, USA, Jul. 2017, pp. 2295–2296.
[22] J. I. Echeveste, M. A. G. de Aza, J. Rubio, and C. Craeye, "Gradient-based aperiodic array synthesis of real arrays with uniform amplitude excitation including mutual coupling," *IEEE Trans. Antennas Propag.*, vol. 65, no. 2, pp. 541–551, Feb. 2017.
[23] C. Lin, A. Qing, and Q. Feng, "Synthesis of unequally spaced antenna arrays by using differential evolution," *IEEE Trans. Antennas Propag.*, vol. 58, no. 8, pp. 2553–2561, Aug. 2010.
[24] V. S. Gangwar, A. K. Singh, and S. P. Singh, "An effective approach for the synthesis of unequally spaced antenna array by estimating optimum elements density on the aperture," *IEEE Antennas Wireless Propag. Lett.*, vol. 16, pp. 2278–2282, Aug. 2017.
[25] Y. Suo, W. Li, Y. Liu, and S. Yin, "A PSO based on SA for the sparse linear array synthesis with multiple constraints," in *Proc. APS/URSI*, Boston, MA, USA, Jul. 2018, pp. 545–546.
[26] G. Buttazzoni and R. Vescovo, "Pencil beam constrained synthesis of linear sparse arrays in presence of coupling effects," in *Proc. APS/URSI*, Boston, MA, USA, Jul. 2018, pp. 2197–2198.
[27] G. Buttazzoni and R. Vescovo, "An efficient and versatile technique for the synthesis of 3D copolar and crosspolar patterns of phase-only reconfigurable conformal arrays with DRR and near-field control," *IEEE Trans. Antennas Propag.*, vol. 62, no. 4, pp. 1640–1651, Apr. 2014.
[28] R. G. Ayestaran, F. Las-Heras, and L. F. Herran, "High-accuracy neural-network-based array synthesis including element coupling," *IEEE Antennas Wireless Propag. Lett.*, vol. 5, pp. 45–48, Mar. 2006.
[29] Z. Wang, S. Fang, Q. Wang, and H. Liu, "An ANN-based synthesis model for the single-feed circularly-polarized square microstrip antenna with truncated corners," *IEEE Trans. Antennas Propag.*, vol. 60, no. 12, pp. 5989–5992, Dec. 2012.
[30] L.-Y. Xiao, W. Shao, F.-L. Jin, and B.-Z. Wang, "Multiparameter modeling with ANN for antenna design," *IEEE Trans. Antennas Propag.*, vol. 66, no. 7, pp. 3718–3723, Jul. 2018.
[31] Y. Wang, X. Yang, and H. Yan, "Reliable fuzzy tracking control of near-space hypersonic vehicle using aperiodic measurement information," *IEEE Trans. Ind. Electron.*, vol. 66, no. 12, pp. 9439–9447, Dec. 2019.
[32] Y. Wang, H. R. Karimi, H.-K. Lam, and H. Yan, "Fuzzy output tracking control and filtering for nonlinear discrete-time descriptor systems under unreliable communication links," *IEEE Trans. Cybern.*, vol. 50, no. 6, pp. 2369–2379, Jun. 2020.
[33] Y. Wang, W. Zhou, J. Luo, H. Yan, H. Pu, and Y. Peng, "Reliable intelligent path following control for a robotic airship against sensor faults," *IEEE/ASME Trans. Mechatronics*, vol. 24, no. 6, pp. 2572–2582, Dec. 2019.
[34] E. D. Übeylı and İ. Güler, "Multilayer perceptron neural networks to compute quasistatic parameters of asymmetric coplanar waveguides," *Neurocomputing*, vol. 62, nos. 1–4, pp. 349–365, Dec. 2004.
[35] D. M. Pozar, "The active element pattern," *IEEE Trans. Antennas Propag.*, vol. 42, no. 8, pp. 1176–1178, Aug. 1994.
[36] R. Q. Wang and Y. C. Jiao, "Multiple-constraint synthesis of rotationally symmetric sparse circular arrays using a hybrid algorithm," *Prog. Electromagn. Res. M*, vol. 79, pp. 33–40, Feb. 2019.
[37] S. X. Rong and L. X. Ming, "The optimization of sparse concentric ring array using differential evolution algorithm," in *Proc. EIIS*, Harbin, China, Jun. 2017, pp. 1–6.
[38] F. Neri and V. Tirronen, "Recent advances in differential evolution: A survey and experimental analysis," *Artif. Intell. Rev.*, vol. 33, nos. 1–2, pp. 61–106, Feb. 2010.
[39] K. Price, R. M. Storn, and J. A. Lampinen, *Differential Evolution: A Practical Approach to Global Optimization*, 1st ed. New York, NY, USA: Springer-Verlag, 2005.
[40] J. Zhang and A. C. Sanderson, "JADE: Adaptive differential evolution with optional external archive," *IEEE Trans. Evol. Comput.*, vol. 13, no. 5, pp. 945–958, Oct. 2009.
[41] T. T. Taylor, "Design of line-source antennas for narrow beamwidth and low side lobes," *IRE Trans. Antennas Propag*, vol. 3, no. 1, pp. 16–28, Jan. 1955.
[42] Y. Liu, Z.-P. Nie, and Q. H. Liu, "A new method for the synthesis of non-uniform linear arrays with shaped power patterns (invited paper)," *Prog. Electromagn. Res.*, vol. 107, pp. 349–363, 2010.

YU GONG was born in Jilin, China. He received the B.S. degree in electronic information science and technology from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2016, where he is currently pursuing the Ph.D. degree in radio physics.

His main research interests include array synthesis, signal processing, optimization methods, and machine learning.

SHAOQIU XIAO (Member, IEEE) received the Ph.D. degree in electromagnetic field and microwave technology from the University of Electronic Science and Technology of China (UESTC), Chengdu, China, in 2003.

From January 2004 to June 2004, he joined UESTC as an Assistant Professor. From July 2004 to March 2006, he worked for the Wireless Communications Laboratory, National Institute of Information and Communications Technology of Japan (NICT), Singapore, as a Research Fellow with the focus on the planar antenna and smart antenna design and optimization. From July 2006 to June 2010, he worked for UESTC as an Associate Professor. He visited the Ecole Normale Superieure de Cachan, Paris, France, as a Senior Research Scholar, from July 2015 to August 2015. He is currently working for UESTC as a Professor. He has authored/coauthored more than 300 technical journals, conference papers, books, and book chapters. His current research interests include planar phased array, frequency selective surface and radar absorber, computational electromagnetics, and microwave circuits and systems.

BING-ZHONG WANG (Senior Member, IEEE) received the Ph.D. degree in electronic engineering from the University of Electronic Science and Technology of China (UESTC), Chengdu, in 1988.

He joined UESTC, in 1984. He has been a Visiting Scholar with the University of Wisconsin-Milwaukee, Milwaukee, WI, USA, a Research Fellow with the City University of Hong Kong, Hong Kong, and a Visiting Professor with the Electromagnetic Communication Laboratory, Pennsylvania State University, University Park, PA, USA. He is currently a Professor with UESTC. His current research interests include computational electromagnetics, antenna theory and techniques, and time reversed electromagnetics.