Hindawi
International Journal of Antennas and Propagation
Volume 2022, Article ID 6987814, 9 pages
https://doi.org/10.1155/2022/6987814

Research Article

A Fast and Efficient Beamforming Algorithm Imitating Plant Growth Gene for Phased Array Antenna

Jinling Zhang,[1] Chen Han ,[1] Zhanqi Zheng,[2] Xiongzhi Zhu ,[3] and Xiaohui Li [1]

[1]*School of Electronic Engineering, Beijing University of Posts and Telecommunications, Beijing 100876, China*
[2]*CICT Mobile Equipment Co, Beijing 100029, China*
[3]*National Center for International Joint Research of Electronic Materials and Systems, School of Electrical and Information Engineering, Zhengzhou University, Zhengzhou 450001, China*

Correspondence should be addressed to Xiongzhi Zhu; zhuxiongzhi@zzu.edu.cn

Received 1 September 2022; Accepted 1 October 2022; Published 4 November 2022

Academic Editor: Chow-Yen-Desmond Sim

In this paper, a new beamforming algorithm for phased array antennas is proposed, the plant growth gene algorithm. The algorithm consists of three steps. Firstly, according to the excitation relation of the array unit before and after the local fine-tuning of the antenna radiation pattern, the model for solving the array unit excitation difference is established. Secondly, the Taylor series expansion is used to solve the model, and the growth model is established based on this, and the beam tuning network is designed to realize the growth model. Finally, based on the growth gene obtained by the neural network algorithm, the growth model is called multiple times for high-precision beamforming. This algorithm converts the complex optimization process of array antenna excitation by the classical optimization algorithm into a simple process of fine-tuning the gain at any angle on the beam to make it grow and approach the target pattern. The growth gene is used to weigh the target angle and gain to achieve beamforming, which greatly reduces the complexity of the algorithm and improves its accuracy of the algorithm. Taking a 1×16 linear array as an example, a cosecant square beam pattern with a coverage range of -31° to 31° and a maximum gain direction of 17° is designed using the algorithm proposed in this paper. The experimental results show that the proposed algorithm can easily fine-tune the gain of any angle to achieve precise beamforming. Importantly, the growth genes trained by the algorithm are universal to the phased array antenna with the same topology.

1. Introduction

Beamforming is an important technology for array antennas to realize long-distance high-quality communication and high-resolution target detection. A lot of research work on array antenna beamforming has been carried out at home and abroad [1–8]. Among them, based on the target beam pattern, there are many research studies on using algorithms to optimize the excitation amplitude and phase of each array unit or optimizing the arrangement of array units to realize beamforming, such as genetic algorithm [5], model order reduction [6], weed algorithm [7], and particle swarm optimization algorithm [8]. The method of using the algorithm for beamforming is to make the gains of all angles of the original beam approach the target pattern, so as to minimize the error between the original and target beam pattern. In these algorithms, the errors of all angles are counted, and the overall errors are aimed at minimizing. However, it is difficult to optimize the errors of a single-angle to the minimum, which leads to the failure of the antenna radiation pattern to accurately fit with the target pattern at a single angle, and it is difficult to achieve further improvement. In addition, for the antenna whose topology is determined, or without changing the antenna structure, when the target pattern needs to be changed or improved according to engineering applications or other reasons, even if only one angle gain needs to be changed, the optimization algorithm needs to be rerun to calculate the excitation corresponding to the new beam pattern. Therefore, the beamforming by the current algorithm has low efficiency and accuracy, high cost, poor generality, and no correction mechanism for local errors, so it is impossible to fine-tune the results of the beamforming.

With the development of deep learning technology, the neural network method is used to model the nonlinear relationship between the antenna radiation pattern and the excitation of the array units, and the coupling effect between the array units is considered in the beamforming process [9–11]. In [12], a convolutional neural network is used to model the beamforming problem of a phased array. The processed two-dimensional pattern image is used as input of the model and the output of the model is the phase of each array unit. In [13], the antenna is placed in a space with obstacles, the neural network is used to deal with the beamforming problem, and the interaction between array units and the existence of obstacles are considered, which is more in line with the practical application scenarios. Besides excitation, the position of each array unit also affects the beam pattern. In [14], the coupling effects of different array unit positions in beamforming are analyzed through the neural network, and the array unit position distribution and excitation are simultaneously optimized by combining optimization algorithms. The transfer learning method is used in [15] to reduce the amount of data required for training and obtains better results than traditional DNN methods.

In this paper, we propose a beamforming algorithm that imitates the plant growth gene (Growth Gene Algorithm (GGA)) and train a set of growth gene parameters by a neural network. Using the growth gene to weigh any target angle and target gain can easily control the gain of any angle on the known antenna radiation pattern and gradually increase the gain of that angle by a certain step size, just as a plant grows a new branch in a local area. The specific process is as follows: firstly, by analyzing the parameter information involved in the gain change of a certain angle of the phased array antenna, a growth model for beam fine-tuning of the single-angle gain of the antenna radiation pattern is proposed. Secondly, the Beam Tuning Network (BTNet) is designed based on the neural network and growth model structure. Finally, the data obtained by an improved genetic algorithm [16] and the growth gene of the phased array antenna are obtained. Accurate beamforming is realized based on the growth gene parameters. Experimental verification shows that we can fine-tune the beam gain at any angle based on the growth gene parameters and can be applied to the phased array antenna of any antenna unit structure. It has universality, greatly reduces the complexity of pattern configuration, and improves the efficiency and precision of array antenna synthesis.

2. The Theory of Gain Fine-Tuning

A method of expressing beamforming with excitation difference is proposed, which converts the array antenna beamforming into high-precision beamforming for any target pattern by gradually fine-tuning the gain at any angle in the antenna radiation pattern. Each of these fine-tuning of the gain at an angle is achieved by adding an excitation difference ∂W corresponding to the pattern difference to the known excitation W^{ori}.

Assuming that the excitation corresponding to the original pattern of the n-unit array antenna is $W^{ori} = \left[w^{ori}(1), w^{ori}(2), \ldots, w^{ori}(n), \ldots, w^{ori}(N) \right]^T$, and the excitation corresponding to the pattern after fine-tuning is $W = [w(1), w(2), \ldots, w(n), \ldots, w(N)]^T$, then the relationship between the corresponding excitation before and after pattern fine-tuning can be expressed as follows:

$$\begin{bmatrix} w(1) \\ w(2) \\ \cdots \\ w(n) \\ \cdots \\ w(N) \end{bmatrix} = \begin{bmatrix} w^{ori}(1) \\ w^{ori}(2) \\ \cdots \\ w^{ori}(n) \\ \cdots \\ w^{ori}(N) \end{bmatrix} + \begin{bmatrix} \partial w(1) \\ \partial w(2) \\ \cdots \\ \partial w(n) \\ \cdots \\ \partial w(N) \end{bmatrix}, \tag{1}$$

where $\partial w(n)$ represents the difference of excitation before and after beamforming of the nth array unit. The above expression is only related to the excitation of the antenna unit and does not involve the antenna array unit. Therefore, for an array antenna with any antenna unit structure, the relationship between the excitations of the array units before and after beamforming can be expressed by (1).

For any beamforming problem, there are corresponding W and $\partial W = [\partial w(1), \partial w(2), \ldots, \partial w(n), \ldots, \partial w(N)]^T$ on the basis of any original excitation W^{ori}. The key of traditional array antenna beamforming is to search for appropriate excitation W. The growth gene algorithm proposed in this paper is to solve ∂W instead of searching W and obtains the excitation corresponding to the target pattern indirectly by using the general model among W, W^{ori}, and ∂W.

3. Materials and Methods

3.1. Growth Gene Algorithm (GGA). As shown in Figure 1, the excitation difference $\partial w(n)$ corresponding to the gain change at a certain angle of the phased array antenna radiation pattern is only related to the following three parameters: the angle to be adjusted, θ (in radians), the original gain of the pattern at the angle θ, $G^{ori}(\theta)$ (in dBi), and the expected gain change, s (in dB). According to the three parameters $\theta, G^{ori}(\theta)$, and s to solve ∂W, the mathematical solution model of ∂W is assumed as follows:

$$\partial w(n) = f(s) g\left(G^{ori}(\theta)\right) h(\theta, n), \tag{2}$$

where $\partial w(n)$ represents the excitation difference of the nth array element before and after the pattern fine-tuning, $f(s)$ and $g(G^{ori}(\theta))$ are unknown functions about s and $G^{ori}(\theta)$ respectively, and $h(\theta, n)$ is the array factor of the target angle of the nth array element

$$h(\theta, n) = e^{-j2\pi d(n-1)\cos(\theta)}. \tag{3}$$

When beamforming the phased array antenna, since the fine-tuned pattern can be arbitrary compared to the original pattern, the gain $G^{ori}(\theta) + s$ of the fine-tuned pattern at angle θ is also arbitrary compared to the gain $G^{ori}(\theta)$ of the original pattern. That is, the expected gain change s at the angle θ is independent of the original gain $G^{ori}(\theta)$. Therefore, in (2), two unary functions $f(s), g(G^{ori}(\theta))$ instead of

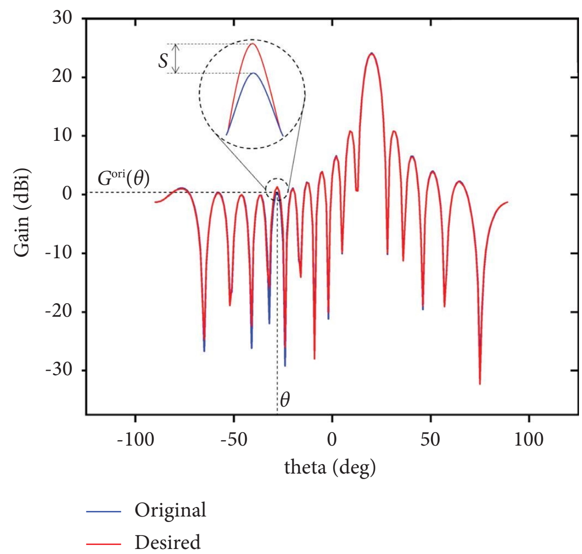

FIGURE 1: When there is only a local difference between the target pattern and the original pattern at the angle and the expected gain change is small, it is difficult to use the optimization algorithm for accurate beamforming.

binary functions $f(s, G^{ori}(\theta))$ are used to represent ∂W model.

Although $f(s)$ and $g(G^{ori}(\theta))$ are unknown in the ∂W model, according to the definition of Taylor series, $f(s)$ can be expressed as follows:

$$f(s) = \sum_{m=0}^{\infty} \frac{f^{(m)}(0)}{m!} s^m = f(0) + f'(0)s + \frac{f''(0)}{2!}s^2 + \ldots + \frac{f^{(m)}(0)}{m!}s^m + \ldots. \tag{4}$$

Let $a_0 = f(0), a_1 = f'(0), a_2 = (f''(0)/2!), \ldots, a_m = (f^{(m)}(0)/m!), \ldots$, then $f(s)$ can be expressed as follows:

$$f(s) = a_0(n) + a_1(n)s^1 + a_2(n)s^2 + \ldots + a_m(n)s^m + \ldots, \tag{5}$$

where n represents the nth array element. Similarly, $g(G^{ori}(\theta))$ can be expressed as follows:

$$g\left(G^{ori}(\theta)\right) = b_0(n) + b_1(n)G^{ori}(\theta) + b_2(n)G^{ori}(\theta)^2 + \ldots + b_m(n)G^{ori}(\theta)^m + \ldots. \tag{6}$$

The higher the order of the Taylor series, the higher the approximation of the function expression. The experimental simulation results show that the fifth-order Taylor series can already meet the accuracy requirements of beamforming. So in this paper, the functions $f(s)$ and $g(G^{ori}(\theta))$ are expanded into the fifth-order Taylor series. The ∂W model can be expressed as follows:

$$\begin{aligned}\partial w(n) = &\left(a_0(n) + a_1(n)s^1 + a_2(n)s^2 + a_3(n)s^3 + a_4(n)s^4 + a_5(n)s^5\right) \times \left(b_0(n) + b_1(n)G^{ori}(\theta)\right. \\ &\left. + b_2(n)G^{ori}(\theta)^2 + b_3(n)G^{ori}(\theta)^3 + b_4(n)G^{ori}(\theta)^4 + b_5(n)G^{ori}(\theta)^5\right) \times e^{-j2\pi d(n-1)\cos(\theta)}.\end{aligned} \tag{7}$$

Since the even and odd powers of negative numbers are opposite numbers, the same ∂W model cannot accurately express the two cases of expected gain change $s > 0$ and $s < 0$. However, the expansion of Taylor series shows that for both cases, the ∂W model has different weights but has similar forms. When $s > 0$, ∂W model can be expressed as follows:

$$\partial w(n) = \left(a_0^+(n) + a_1^+(n)s^1 + a_2^+(n)s^2 + a_3^+(n)s^3 + a_4^+(n)s^4 + a_5^+(n)s^5\right) \times \left(b_0^+(n) + b_1^+(n)G^{ori}(\theta) + b_2^+(n)G^{ori}(\theta)^2 + b_3^+(n)G^{ori}(\theta)^3 + b_4^+(n)G^{ori}(\theta)^4 + b_5^+(n)G^{ori}(\theta)^5\right) \times e^{-j2\pi d(n-1)\cos(\theta)}. \tag{8}$$

When $s < 0$, ∂W model can be expressed as follows:

$$\partial w(n) = \left(a_0^-(n) + a_1^-(n)s^1 + a_2^-(n)s^2 + a_3^-(n)s^3 + a_4^-(n)s^4 + a_5^-(n)s^5\right) \times \left(b_0^-(n) + b_1^-(n)G^{ori}(\theta) + b_2^-(n)G^{ori}(\theta)^2 + b_3^-(n)G^{ori}(\theta)^3 + b_4^-(n)G^{ori}(\theta)^4 + b_5^-(n)G^{ori}(\theta)^5\right) \times e^{-j2\pi d(n-1)\cos(\theta)}. \tag{9}$$

It can be seen from (8) and (9) that for the problem of fine-tuning the gain of the single-angle in beamforming, as long as the weights $(a_0^+(n), \ldots, a_5^+(n), b_0^+(n), \ldots, b_5^+(n); a_0^-(n), \ldots, a_5^-(n), b_0^-(n), \ldots, b_5^-(n))$ in the ∂W model are appropriate, the excitation corresponding to each array element after the gain fine-tuning can be calculated according to s and $G^{ori}(\theta)$. Since the positive fine-tuning of the gain at an angle $(s > 0)$ is similar to the plant gradually growing a new branch in a certain part, the weights $(a_0^+(n), \ldots, a_5^+(n), b_0^+(n), \ldots, b_5^+(n))$ that plays a key role in the process of positive fine-tuning of gain are called positive growth gene, and the weights $(a_0^-(n), \ldots, a_5^-(n), b_0^-(n), \ldots, b_5^-(n))$ that play a role in the process of negative fine-tuning $(s < 0)$ are called negative growth gene, and the gain fine-tuning model ∂W is named "Growth Model."

In this paper, the method of using the growth model to adjust the gain of any angle of the radiation pattern to realize the precise beamforming of the phased array antenna is named the Growth Gene Algorithm (GGA). The beamforming of GGA only involves simple matrix multiplication and addition operation, so the complexity is much lower than other beamforming algorithms. The solution of GGA is $\partial w(n)$ of a single array element after beamforming. Each array element corresponds to a growth model and two sets of gene parameters. Taking 1×16 linear array antenna as an example, a total of $2 \times 12 \times 16 = 384$ gene parameters need to be determined. Since the growth model does not involve the radiation pattern of the array element, for the 1×16 linear array antenna with the same topology, these 384 gene parameters are applicable to the antenna element of any structure.

In the next section, Beam Tuning Network (BTNet) is designed based on a growth gene algorithm and neural network, and the growth genes of 1×16 linear array antenna are trained to verify the effectiveness of the algorithm proposed in this paper for fine-tuning gain at any angle and high-precision beamforming.

3.2. Beam Tuning Network (BTNet). The neural network model designed to train the growth model of each array element is shown in Figure 2.

In BTNet, the weight training by the convolutional layer of sub network 1 corresponds to the gene parameter $a_1(n), a_2(n), a_3(n), a_4(n), a_5(n)$ related to $f(s)$, and the trained bias corresponds to the gene parameter $a_0(n)$. The activation function used by the convolutional layer is "identity mapping," that is, $y = \sigma(x) = x$. The architecture of subnetwork 2 is similar to that of subnetwork 1, in that the weight and bias trained by the convolutional layer correspond to the gene parameters related to $g(G^{ori}(\theta))$, and the activation function is also "identity mapping." In order to make the output of the neural network consistent with the growth model, sub network 3 uses the "multiplicator layer" designed for the structure of the growth model. Its input is $f(s)$ and $g(G^{ori}(\theta))$ learned by the network, and the output is $f(s) \times g(G^{ori}(\theta))$. According to the form of array factor in the growth model, the activation function in the sub network 3 is designed as $y = \sigma(x) = h(\theta, n)x$, so that the output of the activation function is the growth model $\partial w(n)$ of the nth array element.

To sum up, the process of beamforming by the GGA is shown in Figure 3. The gene parameters of the growth model corresponding to each array element are trained through BTNet, and the growth model obtained by training is used to calculate the excitation of each array element corresponding to the target pattern to complete the beamforming. The algorithm can fine-tune the gain at any angle on any radiation pattern, achieve high-precision beamforming, decouple the main lobe and side lobe, and have the characteristics of high computing efficiency and real-time data feedback.

4. Algorithms and Model Validation

4.1. Training Strategy. Taking the 1×16 linear array antenna as an example to verify the algorithm, BTNet is used to train the gene parameters of the growth model of each array element, and the Adam optimizer is used in the training [17]. In the experiment, the data pair (θ, s) composed of random angles and gain change values is used as the optimization target pattern, and the improved genetic algorithm [16] is used to generate 20,000 sets of data in the format of $(\theta, G(\theta), s, \partial W)$, in which the ratio of $s > 0$ to $s < 0$ is 1 : 1.

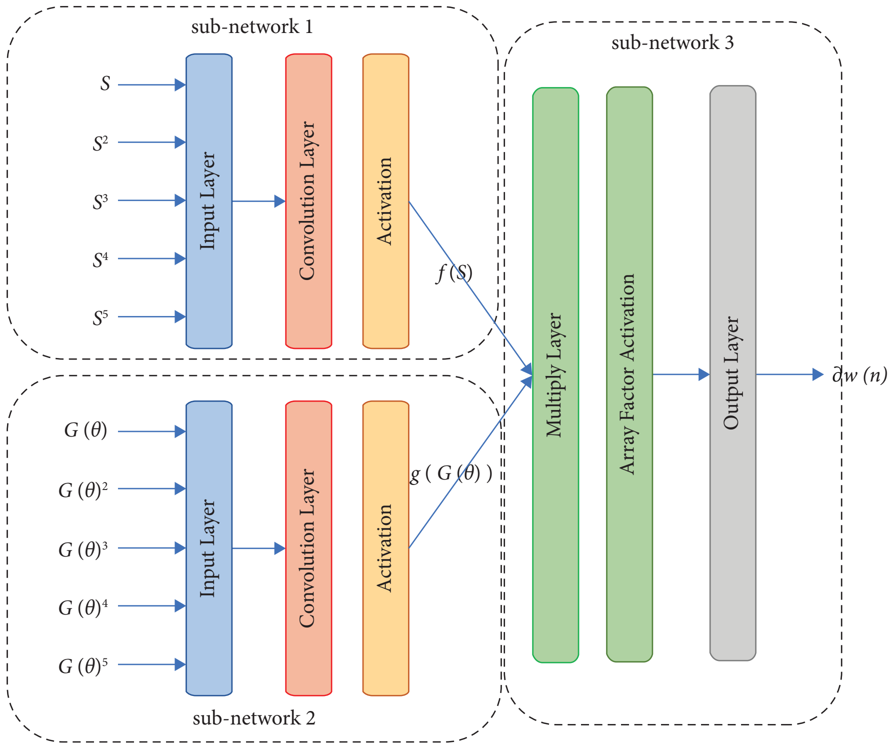

Figure 2: BTNet is a general model structure designed based on the neural network combined with the antenna gain fine-tuning problem. Different from the traditional neural network, it has two input layers and its network structure can be divided into three subnetworks.

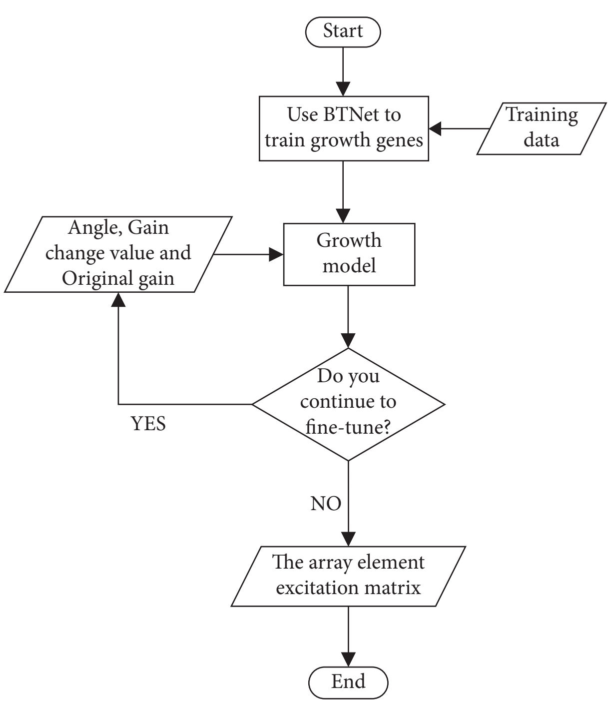

Figure 3: Flow chart of beamforming using growth model; in the process of beamforming, the gene parameters of the growth model corresponding to each array element were first trained by BTNet. Then, use the trained growth model to calculate the excitation of each array element corresponding to the target pattern and complete the beamforming.

BTNet is trained with 8000 sets of data, and 2000 sets of data are used for validation. Set the number of training times (epochs) to 400 and batch size to 100, and use the CPU version of TensorFlow 1.8.0 for simulation. In order to improve the efficiency of training, the transfer learning [18] strategy is adopted in training. Except that the initial weights are randomly generated in the first training, the weights trained in the previous training are used as the initial weights in the rest of the training. For the 1×16 linear array antenna, in order to obtain 2 sets of gene parameters of the growth model of each array element, BTNet was used for 32 times of training. Under the condition of Intel(R) Core(TM) I5-7300HQ CPU @ 2.50 GHz processor and 8 Gb memory, the first training takes about 7 minutes, and the rest training takes about 1 minute and 27 seconds on average due to the strategy of transfer learning.

4.2. Experimental Results of Growth Genes. Under the condition of 20000 sets of data, BTNet was used to train the growth model of 1×16 linear arrays, and the obtained 192 positive growth gene parameters and negative growth gene parameters are shown in Tables 1 and 2, respectively.

The gene parameters in this table are applicable to the 1×16 linear array antenna of any antenna element structure. By using the growth model to fine-tune the gain at any angle on the radiation pattern, the corresponding excitation of each array element can be calculated quickly.

Table 1: 1×16 linear array positive growth gene parameters.

	a0	a1	a2	a3	a4	a5	b0	b1	b2	b3	b4	b5
$n=1$	0	0.115134	0.006719	0.000341	0.000090	0.000010	1	0.007226	0.000444	0.000088	0.000014	0.000072
$n=2$	0	0.115141	0.006695	0.000299	0.000088	0.000072	1	0.007290	0.000499	0.000053	0.000070	0.000020
$n=3$	0	0.115133	0.006704	0.000300	0.000058	0.000090	1	0.007261	0.000472	0.000106	0.000081	0.000058
$n=4$	0	0.115148	0.006654	0.000339	0.000013	0.000049	1	0.007217	0.000508	0.000091	0.000050	0.000047
$n=5$	0	0.115136	0.006698	0.000254	0.000017	0.000052	1	0.007210	0.000492	0.000102	0.000072	0.000015
$n=6$	0	0.115196	0.006682	0.000347	0.000075	0.000080	1	0.007245	0.000453	0.000103	0.000008	0.000013
$n=7$	0	0.115147	0.006669	0.000333	0.000090	0.000006	1	0.007240	0.000463	0.000062	0.000066	0.000063
$n=8$	0	0.115159	0.006673	0.000252	0.000108	0.000017	1	0.007211	0.000447	0.000040	0.000049	0.000034
$n=9$	0	0.115225	0.006722	0.000255	0.000084	0.000027	1	0.007242	0.000465	0.000114	0.000042	0.000098
$n=10$	0	0.115160	0.006700	0.000317	0.000064	0.000070	1	0.007267	0.000428	0.000033	0.000100	0.000017
$n=11$	0	0.115133	0.006686	0.000338	0.000077	0.000019	1	0.007237	0.000456	0.000118	0.000016	0.000086
$n=12$	0	0.115194	0.006668	0.000269	0.000053	0.000048	1	0.007212	0.000469	0.000043	0.000038	0.000058
$n=13$	0	0.115155	0.006659	0.000312	0.000037	0.000082	1	0.007298	0.000483	0.000054	0.000058	0.000011
$n=14$	0	0.115221	0.006718	0.000332	0.000036	0.000059	1	0.007202	0.000453	0.000051	0.000016	0.000018
$n=15$	0	0.115172	0.006639	0.000310	0.000057	0.000070	1	0.007270	0.000474	0.000023	0.000007	0.000032
$n=16$	0	0.115183	0.006695	0.000291	0.000092	0.000072	1	0.007297	0.000463	0.000053	0.000011	0.000061

Table 2: 1×16 linear array negative growth gene parameters.

Gene	a0	a1	a2	a3	a4	a5	b0	b1	b2	b3	b4	b5
$n=1$	0	0.115199	−0.006574	0.000290	−0.000004	0.000078	1	0.007234	0.000471	0.000094	0.000010	0.000013
$n=2$	0	0.115185	−0.006581	0.000339	0.000070	0.000073	1	0.007205	0.000417	0.000029	0.000080	0.000094
$n=3$	0	0.115198	−0.006617	0.000322	0.000001	0.000012	1	0.007264	0.000443	0.000085	0.000075	0.000058
$n=4$	0	0.115204	−0.006607	0.000323	0.000087	0.000087	1	0.007209	0.000447	0.000057	0.000069	0.000060
$n=5$	0	0.115209	−0.006593	0.000271	−0.000001	0.000077	1	0.007221	0.000449	0.000075	0.000023	0.000064
$n=6$	0	0.115178	−0.006615	0.000328	0.000000	0.000029	1	0.007224	0.000463	0.000029	0.000041	0.000010
$n=7$	0	0.115141	−0.006552	0.000279	0.000050	0.000096	1	0.007243	0.000479	0.000096	0.000043	0.000066
$n=8$	0	0.115141	−0.006537	0.000269	0.000017	0.000080	1	0.007249	0.000487	0.000060	0.000027	0.000004
$n=9$	0	0.115197	−0.006587	0.000295	0.000051	0.000006	1	0.007232	0.000487	0.000090	0.000013	0.000013
$n=10$	0	0.115139	−0.006629	0.000292	0.000056	0.000072	1	0.007253	0.000421	0.000083	0.000013	0.000013
$n=11$	0	0.115140	−0.006616	0.000267	0.000010	0.000032	1	0.007232	0.000432	0.000045	0.000089	0.000070
$n=12$	0	0.115186	−0.006612	0.000271	−0.000002	0.000091	1	0.007271	0.000466	0.000051	0.000017	0.000062
$n=13$	0	0.115229	−0.006613	0.000276	0.000030	0.000007	1	0.007268	0.000450	0.000118	0.000040	0.000062
$n=14$	0	0.115145	−0.006592	0.000266	0.000066	0.000087	1	0.007235	0.000479	0.000049	0.000053	0.000083
$n=15$	0	0.115190	−0.006596	0.000280	0.000035	0.000042	1	0.007236	0.000466	0.000094	0.000042	0.000043
$n=16$	0	0.115142	−0.006628	0.000279	0.000022	0.000065	1	0.007296	0.000504	0.000066	0.000024	0.000076

4.3. Experimental Verification of Beam Fine-Tuning. The antenna structure and unit structure of the 1×16 linear arrays used to verify the growth gene algorithm are shown in Figure 4 and Figure 5. The antenna unit adopts a patch structure with a length of 37.26 mm and a width of 28 mm. The unit spacing is d, feeding from the coaxial line, and the feeding point is located 7 mm above the center of the patch antenna. The relative dielectric constant and thickness of substrate FR4 are 4.4 and 1.6 mm respectively. The working frequency of the antenna is 2.45 GHz, and the working air wavelength is $\lambda = 122$ mm. Figure 6(a) shows a flat-topped beam pattern of the 1×16 array antenna under the condition of array spacing $d = 0.5\lambda$. The amplitude and phase of the excitation corresponding to the flat-topped beam pattern are shown in Table 3.

As shown in Figure 6(a), the gain of the flat-topped beam pattern at $\theta = 18^\circ$ is 8dBi. In the experiment, the growth model trained by BTNet is used to fine-tune the gain at this angle by decreasing 3 dB and increasing 3 dB, respectively.

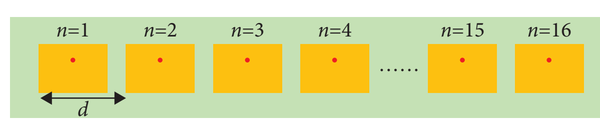

Figure 4: 1×16 linear array antenna.

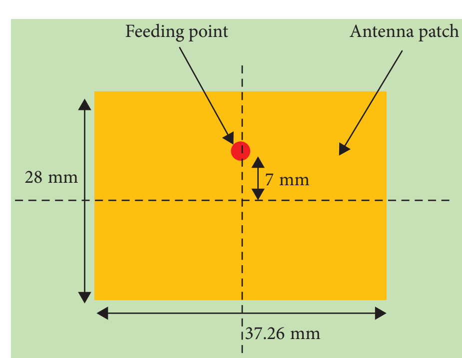

Figure 5: Patch antenna element.

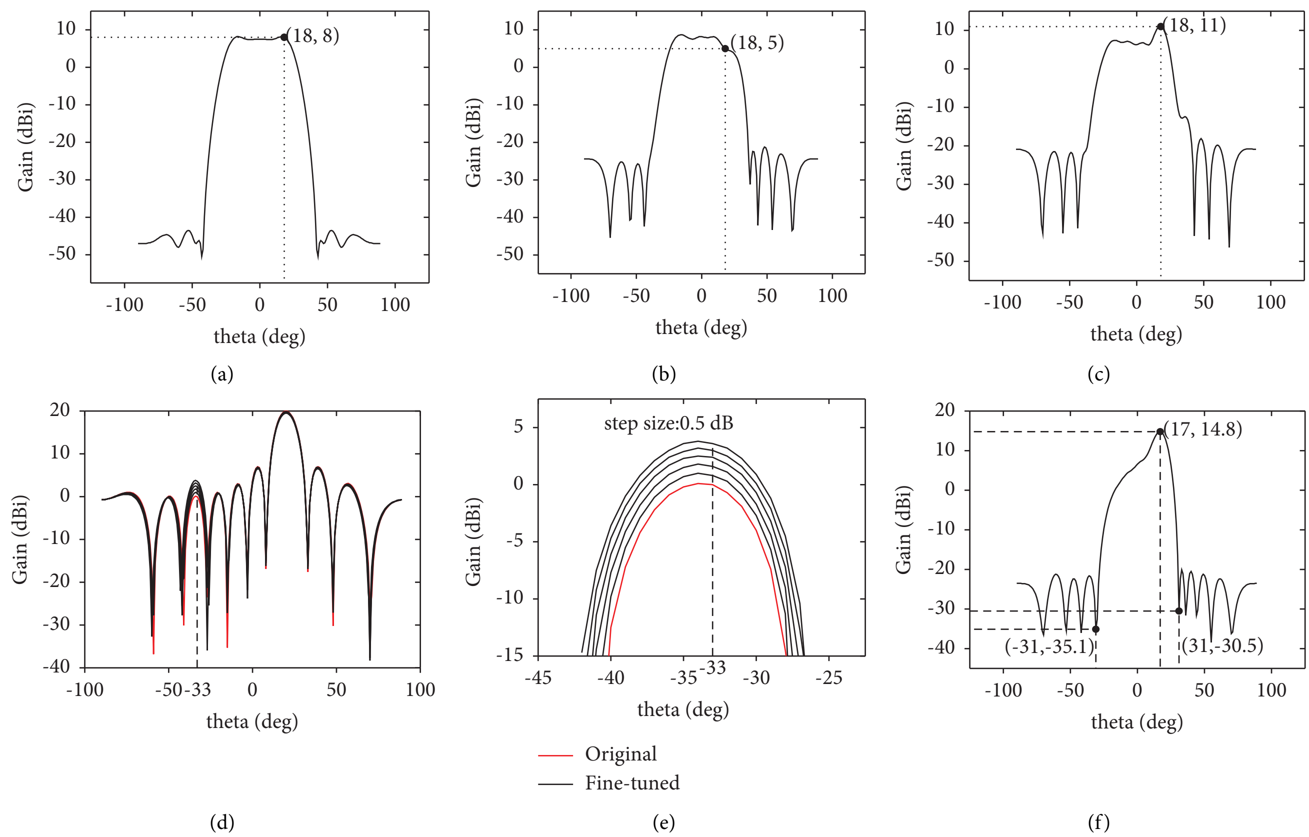

Figure 6: (a) The flat-topped beam pattern of the 1×16 antenna array under the condition of array spacing $d = 0.5\lambda$ and gain $G(\theta) = 8$ dBi at angle $\theta = 18^{\circ}$(b) the fine-tuned pattern has a gain of 5dBi at $\theta = 18^{\circ}$, which is 3 dB lower than the original flat-topped beam pattern; (c) the fine-tuned pattern has a gain of 11dBi at $\theta = 18^{\circ}$, which is 3 dB higher than the original flat-topped beam pattern. The random pattern of the 1×16 array antenna with array spacing $d = 0.5\lambda$ was taken as the initial pattern, and the growth model was used to carry out continuous fine-tuning of the side lobe gain at $\theta = -33^{\circ}$ with equal step size; (d) the general view of pattern tuning; (e) the detail view; (f) the cosecant squared pattern designed using the growth model, the coverage range of the formed pattern is -31°~31°, and the maximum gain direction is $\theta = 17^{\circ}$.

Table 3: The excitation corresponding to the flat-top beam.

Num	Amplitude (V)	Phase
1	0.80304	0deg
2	2.6395	0deg
3	3.6352	0deg
4	5.1516	−180deg
5	17.3215	180deg
6	6.1928	−180deg
7	28.7732	0deg
8	35.5739	0deg
9	19.9924	180deg
10	88.6058	−180deg
11	100	−180deg
12	57.2576	−180deg
13	14.716	180deg
14	0.68384	0deg
15	0.99359	0deg
16	0.14001	0deg

Table 4: The excitation after gain fine-tuning.

Num	Gain decreased by 3 dB		Gain increased by 3 dB	
	Amplitude	Phase	Amplitude	Phase
1	4.672 V	167.6122deg	8.4425 V	−9.5788deg
2	4.0511 V	−105.728deg	9.5885 V	34.7518deg
3	7.2696 V	−48.0397deg	7.8159 V	75.7985deg
4	2.2398 V	−100.0909deg	12.2195 V	165.3479deg
5	13.9121 V	167.903deg	22.9001 V	−169.7203deg
6	8.3639 V	138.906deg	9.7894 V	−128.0798deg
7	26.1478 V	7.2343deg	33.0671 V	−8.0232deg
8	32.3632 V	−3.1515deg	40.2185 V	3.5551deg
9	23.1728 V	−166.7736deg	18.0008 V	155.6163deg
10	90.0087 V	−177.3203deg	86.9729 V	176.1113deg
11	100V	179.685deg	100V	−179.5584deg
12	57.9555 V	175.2204deg	56.9624 V	−173.1756deg
13	18.6731 V	164.7761deg	12.2061V	−145.7311deg
14	4.7879 V	171.4933deg	8.3389 V	−6.8373deg
15	4.8673 V	−122.5324deg	8.3516 V	43.5287deg
16	5.5395 V	−74.6664deg	7.6825 V	102.9145deg

The corresponding excitation of each array element calculated by the growth model is shown in Table 4. The electromagnetic simulation software CST is used to simulate the excitation in Table 4, and the simulation results are shown in Figure 6(b) and Figure 6(c), respectively. It can be observed that the gains of the fine-tuned beam pattern at $\theta = 18^{\circ}$ are 5dBi and 11dBi, respectively, indicating that the growth model is very accurate for the fine-tuning pattern at any angle.

Table 5: Comparison of the growth model with different algorithms.

Year/Ref	Algorithm	Antenna array	Optimization objectives	Computational time
[2020]/[5]	Generic algorithm	10-circular	Beamforming	110 s
[2015]/[6]	Model order reduction	Rectangular	Beamforming	644 s
[2020]/[7]	Invasive weed optimization	10-circular	Beamforming	114.5 s
[2019]/[8]	Particle swarm optimization	8×1 Patch	Beamforming	31.7 s
[2019]/[8]	Firefly algorithm	8×1 Patch	Beamforming	55.4 s
[2019]/[8]	Taguchi's method	8×1 Patch	Beamforming	194.8 s
Proposed	Growth model	1×16-Linear	Gain fine-tuning	<10 ms

Figure 6(d) and Figure 6(e) show the results of continuously fine-tuning the side lobe gain of the 1×16 linear array antenna at $\theta = -33^{\circ}$ with 0.5 dB steps using the growth model. It can be found that using the growth model to fine-tune the side lobes of the antenna radiation pattern will not affect the gain of the main lobe of the pattern and vice versa. This conclusion is also verified in Figure 6(b) and Figure 6(c).

The fine-tuning of the gain at any angle by the above growth model takes an average of less than 10 ms, which verifies the high efficiency of the growth gene algorithm for fine-tuning patterns, which is convenient for real-time application in engineering. Table 5 shows the comparison of the computational times between the growth model and some beamforming algorithms in the literature. The growth model uses G (θ) (in dBi, the original gain at the angle θ) and s (in dB, the expected gain change at the angle θ) as inputs, where s is the step size of beam fine-tuning.

5. Experimental Verification of Cosecant Squared Beamforming

The beamforming of the array antenna can be achieved by fine-tuning the gain at multiple angles of the array radiation pattern using the growth model. In the experiment, the cosecant squared pattern with the pattern coverage range of -30° to 30° and the maximum gain direction of 16° to 18° is taken as the target, and the growth model is used to synthesize the beam pattern. The corresponding excitation of each element calculated by the growth model is shown in Table 6, and the radiation pattern simulated by using this excitation is shown in Figure 6(f).

As can be seen from Figure 6(f), the coverage range of the formed beam pattern is -31° to 31°, and the maximum gain direction is 17°, which well meets the design requirements of the cosecant squared pattern. Moreover, compared with the complex steps needed to be performed when using the optimization algorithm to perform the beamforming, the growth gene algorithm only needs to input the original gain $G^{ori}(\theta)$ and the expected gain change s at the corresponding angle θ into the growth model to calculate the excitation corresponding to the target pattern. It reduces the difficulty for antenna designers to implement high-precision beamforming.

Table 6: The excitation corresponding to the cosecant squared.

Num	Amplitude (V)	Phase
1	5.7913	126.8469deg
2	10.8134	−166.273deg
3	14.2734	−102.7381deg
4	22.9186	−40.2313deg
5	26.7167	14.9124deg
6	37.0708	87.1692deg
7	54.3422	138.6947deg
8	5.7913	126.8469deg
9	58.1583	−168.3989deg
10	75.653	−107.6462deg
11	100	−69.8895deg
12	90.6976	−45.7976deg
13	48.0395	−23.0731deg
14	8.2269	47.2888deg
15	16.8154	166.8038deg
16	13.9234	−158.1047deg

6. Conclusions

In this paper, a beamforming algorithm, called the growth gene algorithm, which imitates plant growth genes, is proposed. By establishing the growth model, the algorithm can fine-tune the gain of the antenna radiation pattern at any angle to achieve high efficiency and accurate beamforming of the target pattern. By fine-tuning multiple angles of the radiation pattern of 1×16 linear arrays, the cosecant square beamforming with a coverage range of -31° to 31° and maximum gain direction of 17° is realized. The process of fine-tuning the gain of the growth gene algorithm only involves simple matrix multiplication and addition operations, so the complexity of achieving precise beamforming is low. The growth model is only related to the gain value before and after the gain fine-tuning and the angle of the gain fine-tuning, but not to the array spacing and the array element radiation pattern. Therefore, these gene parameters can be applied to other 1×16 linear array antennae. More importantly, the growth gene algorithm proposed in this paper can be easily extended to the beamforming of area array antennas.

Data Availability

The data used to support the findings of this study are included in the article.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported by the Fundamental Research Funds for the Central Universities (No. 2019XD-A07), the National Natural Science Foundation of China (No. 62271063), the Director Fund of Beijing Key Laboratory of Space Ground Interconnection and Convergence, and the National Key Laboratory of Science and Technology on Vacuum Electronics.

References

[1] A. K. Pandey, "Phased Array Antenna with Beamforming Network for 5G mmWave Communication System," in *Proceedings of the 2020 50th European Microwave Conference (EuMC)*, pp. 364–367, Utrecht, Netherlands, January 2021.

[2] G. Zhou, "Energy efficiency beamforming design for UAV communications with broadband hybrid polarization antenna arrays," *IEEE Access*, vol. 7, pp. 34521–34532, 2019.

[3] A. H. Hussein, H. S. Fouda, H. H. Abdullah, and A. A. M. Khalaf, "A highly efficient spectrum sensing approach based on antenna arrays beamforming," *IEEE Access*, vol. 8, pp. 25184–25197, 2020.

[4] H. Li, Y. Chen, and S. Yang, "Harmonic beamforming in antenna array with time-modulated amplitude-phase weighting technique," *IEEE Transactions on Antennas and Propagation*, vol. 67, no. 10, pp. 6461–6472, 2019.

[5] A. Hamza and H. Attia, "Fast beam steering and null placement in an adaptive circular antenna array," *IEEE Antennas and Wireless Propagation Letters*, vol. 19, no. 9, pp. 1561–1565, 2020.

[6] Y. Sato, F. Campelo, and H. Igarashi, "Fast shape optimization of antennas using model order reduction," *IEEE Transactions on Magnetics*, vol. 51, no. 3, pp. 1–4, Article ID 7204304, 2015.

[7] G. G. Roy, S. Das, P. Chakraborty, and P. N. Suganthan, "Design of non-uniform circular antenna arrays using a modified invasive weed optimization algorithm," *IEEE Transactions on Antennas and Propagation*, vol. 59, no. 1, pp. 110–118, 2011.

[8] L. A. Greda, A. Winterstein, D. L. Lemes, and M. V. T. Heckler, "Beamsteering and beamshaping using a linear antenna array based on particle swarm optimization," *IEEE Access*, vol. 7, pp. 141562–141573, 2019.

[9] R. G. Ayestaran, F. Las-Heras, and L. F. Herran, "High-accuracy neural-network-based array synthesis including element coupling," *IEEE Antennas and Wireless Propagation Letters*, vol. 5, pp. 45–48, 2006.

[10] R. G. Ayestaran, F. Las-Heras, and L. F. Herran, "Neural modeling of mutual coupling for antenna array synthesis," *IEEE Transactions on Antennas and Propagation*, vol. 55, no. 3, pp. 832–840, 2007.

[11] J. H. Kim and S. W. Choi, "A deep learning-based approach for radiation pattern synthesis of an array antenna," *IEEE Access*, vol. 8, pp. 226059–226063, 2020.

[12] R. Lovato and X. Gong, "Phased antenna array beamforming using convolutional neural networks," in *Proceedings of the 2019 IEEE International Symposium on Antennas and Propagation and USNC-URSI Radio Science Meeting*, pp. 1247-1248, Atlanta, GA, USA, July 2019.

[13] R. G. Ayestaran and F. Las-Heras, "Obstacle modeling in array synthesis using neural networks," *IEEE Transactions on Antennas and Propagation*, vol. 54, no. 8, pp. 2420–2424, 2006.

[14] Y. Gong, S. Xiao, and B.-Z. Wang, "An ANN-based synthesis method for nonuniform linear arrays including mutual coupling effects," *IEEE Access*, vol. 8, pp. 144015–144026, 2020.

[15] H. Yang, J. Jee, G. Kwon, and H. Park, "Deep transfer learning-based adaptive beamforming for realistic communication channels," in *Proceedings of the 2020 International Conference on Information and Communication Technology Convergence*, pp. 2162–1233, Jeju, Korea, December 2020.

[16] J. L. Zhang, W. G. Wan, Z. Zhan-Qi, X. Gan, and X. Y. Zhu, "Research on X band extended cosecant squared beam synthesis of micro-strip antenna arrays using genetic algorithm," *Acta Physica Sinica*, vol. 64, no. 11, Article ID 110504, 2015.

[17] D. Kingma and J. Ba, "Adam: A Method for Stochastic Optimization," ICLR, San Diego, pp. 2162–1233, Article ID 1412.6980, 2015.

[18] S. J. Pan and Q. Yang, "A survey on transfer learning," *IEEE Transactions on Knowledge and Data Engineering*, vol. 22, no. 10, pp. 1345–1359, 2010.