

# 天线阵列方向图的一种数值综合算法

## A Numerical Pattern Synthesis Algorithm for Antenna Arrays

张志军 冯正和

(清华大学电子工程系微波与数字通信国家重点实验室,北京 100084)

**【摘要】** 本文提出了一种新型阵列综合算法;目标方向图迭代算法。这种算法与现有的阵列综合方法不同,它通过对目标方向图的迭代来调整实际方向图的形状,是一种纯数值的阵列综合算法。这种算法适用于任意结构阵列的方向图综合,计算效率高,可以满足实际工程的需要。作为验证,本文综合了一些具有代表性的天线阵列,给出了计算结果,并对结果进行了讨论。

**关键词:** 天线阵列, 方向图综合, 数值方法, 迭代算法

**Abstract:** The paper presents a new numerical synthesis algorithm, goal pattern iterative algorithm, for antenna arrays. This algorithm adjusts the actual pattern of arrays by modifying goal pattern in iteration. It can be applied to arbitrary array in which the individual elements have arbitrary radiation pattern. Some representative antenna arrays are synthesized to show the validity of the method, the calculated results are presented and discussed in detail.

**Key words:** Antenna arrays, Pattern synthesis, Numerical method, Iterative algorithm

### 一、引 言

由于天线阵列具有灵活的方向图形成能力及波束扫描能力,所以天线阵列自出现以来就一直受到广泛的重视,并得到了深入的研究。就已有的文献看,阵列方向图综合主要可以分为均匀直线阵和任意阵列两大类算法。

均匀直线阵由等间距共轴分布的一组天线单元构成,通常假设各单元方向图相同,并且假设各单元之间无互耦存在。经典的 Chebyshev 综合就属于这一类,它采用了纯解析的算法综合等副瓣阵列方向图。一些近期的文献<sup>[1,2]</sup>则采用了多项式极点移动等方法进行方向图综合,它们部分地应用了数值算法。所有针对均匀直线阵进行的研究,都充分利用了均匀直线阵所具有的结构简单性,进行了相当的数学变换,从而使综合问题更简单、综合算法更高效。

当然,这种简单和高效是以牺牲适用范围为代价换来的,所以这类方法都只能局限于理想均匀直线阵列综合问题。

在实际工程中,由于天线阵列的安装环境的影响及限制,天线阵列单元往往不是均匀直线分布的,天线阵列单元方向图亦非各向同性,且天线单元之间有互耦存在。针对这些要求,就出现了任意阵列的方向图综合算法。对于任意阵列的方向图综合,由于阵列分布的随意性以及各阵列单元方向图的随意性,很难利用解析算法进行计算,因此通常采用数值计算的方法。

Boguis<sup>[3]</sup>, Buckley<sup>[4]</sup>采用了最直观的优化方法,它们直接利用所需的目标方向图构造优化算法所需的目标函数,以各阵元的激励作为优化的变量,采用通用的优化算法进行寻优。由于这类方法使用的是通用优化算法,因此它们对初值的选取较为敏

• 1997年1月收到,1997年10月修改定稿,国家863项目

•• Zhang Zhijun, Feng Zhenghe (State Key Lab on Microwave and Digital Communications, Tsinghua University, Beijing 100084)

感。在通常情况下,它们的收敛速度较慢,并且很容易陷入局部极值而无法得到满足要求的解。

Versovoj<sup>[5]</sup>采用了投影的方法,它将所需的目标方向图投影到实际可能的阵列方向图的解空间上,从而获得与目标方向图的均方差积分最小的实际方向图,这种方法无需迭代,计算简单,但它无法对主瓣及副瓣形状进行调整,因此无法获得最佳方向图。

Ng<sup>[6]</sup>、Er<sup>[7,9]</sup>、Sim<sup>[8]</sup>采用了分段的方法,它们在方向图的某些角度范围内施加约束条件,计算使其余部分与目标方向图的均方差积分最小的阵列激励。其中文献[6~8]对阵列方向图副瓣辐射总功率进行约束,求解使主瓣方向图与目标方向图的均方差积分最小的阵列激励。文献[9]对实际主瓣方向图与目标方向图的均方差积分进行约束,求解使副瓣辐射总功率最小的阵列激励。这些方法<sup>[6~9]</sup>都可以通过调整约束条件对副瓣平均幅度进行调整,但這些方法都无法对阵列方向图副瓣进行独立的、精确的控制。

Tseng<sup>[10]</sup>提出了一种高效的阵列方向图综合算法,它可以精确地控制阵列方向图副瓣的形状及幅度。它的收敛速度快,稳定性好,但它必须计算包含阵列各单元方向图导数的 Lagrange 极值问题,对于一些实际的阵列方向图,例如包含互耦的阵列单元方向图,它们的导数很难用公式表达,因此利用 Tseng 方法就有一定的困难。Tseng 方法的另一个局限是它无法综合阵元平均间距超过半波长天线阵列。

Olien<sup>[11]</sup>、Swart<sup>[12]</sup>、Tennant<sup>[13]</sup>采用了一种基于自适应算法的阵列方向图综合算法,这是一种纯数

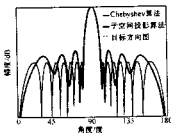


图1 Chebyshev和子空间投影算法的综合结果

与 Chebyshev 算法所得到的方向图相比,子空间投影算法综合出的方向图是很粗糙的,这主要是因为开始综合时,所能实现的阵列方向图还是未知

值算法,它通过迭代调整各副瓣方向的虚拟干扰度,然后通过自适应算法获得各阵列单元激励值。这种方法无需计算单元方向图的导数,可以计算任意单元类型及任意布局的天线阵列,具有较好收敛性。

本文提出了一种新型阵列综合算法:目标方向图迭代算法。这也是一种纯数值的阵列方向图综合算法。它与文献[11~13]的算法具有相似的优点,它无需计算单元方向图导数,可以计算任意单元类型及任意布局的天线阵列,如果采用文献[14]中的考虑互耦的分析方法,它还可以处理阵列单元之间存在互耦的圆形阵列综合问题。与自适应阵列方向图综合算法<sup>[12]</sup>相比,本文的收敛速度更快,所需迭代次数更少,并且显著地降低了每次迭代中的计算量。当阵列单元数  $N$  较大时,本文算法每次迭代所需的计算量仅为自适应阵列方向图综合算法的  $2/N$ 。

## 二、理论分析

为了清晰地阐述本文的阵列综合算法,我们首先以一个均匀直线阵为例进行讨论。众所周知利用 Chebyshev 算法是可以综合出等副瓣的阵列方向图的,假设已利用 Chebyshev 算法综合了一个满足要求的阵列方向图,如图1中的细实线所示。下面我们利用子空间投影法去综合同样的一个问题。显然,在开始综合时不可能知道最终实际方向图主瓣和副瓣的详细结构,因此通常会以所需的方向图副瓣峰值包络作为目标方向图,如图1中虚线所示。对这个目标方向图进行子空间投影运算,可以得到图1中粗实线所示的方向图。

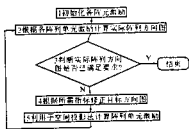


图2 目标方向图迭代算法的原理流程图

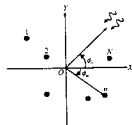


图3 任意阵列的位置分布示意图

的,因而无法准确地确定目标方向图,如果我们以 Chebyshev 算法所得到的方向图作为目标方向图进行子空间投影运算,那么可以得到与 Chebyshev 算法

完全一致的结果。这说明只要能够找到一种合理的算法,通过迭代来不断修正目标方向图,就可以综合出满足要求的实际阵列方向图。本文所提出的正是一种基于目标方向图迭代算法的阵列综合方法。

图 2 是目标方向图迭代算法的原理流程框图。目标方向图迭代算法的核心是目标方向图修正方案,目标方向图修正方案在每次迭代中对上一轮迭代所产生的实际方向图进行修正,从而生成本轮迭代所需的目标方向图。选择不同的目标方向图修正方案,将会影响基于目标方向图迭代算法的阵列综合方法的效率及稳定性。本文所介绍的是作者已构造的多种修正方案中性能最稳定、收敛速度最快的方案。不妨假设所要求的实际方向图副瓣峰值包络线为  $h(\varphi)$ ,下面分别具体介绍流程框图中的五个步骤。

### 1. 初始化各阵元激励

假设有图 3 所示的任意天线阵列,选择如图 3 所示的坐标系,坐标原点为  $o$ ,并对各阵列单元进行如图 3 所示的编号,阵列方向图可由式(1)给出。

$$f(\varphi) = \sum_{n=1}^N \omega_n \cdot g_n(\varphi) \\ = \sum_{n=1}^N \omega_n \cdot |g_n(\varphi)| \cdot e^{j \arg(g_n(\varphi))} \quad (1)$$

式中  $f(\varphi)$  为阵列矢量方向图(这里矢量指包含相位关系的量,并非空间矢量,以下同), $g_n(\varphi)$  ( $n=1, \dots, N$ ) 为各阵列单元以坐标原点  $o$  为参考点的单元方向图,式中  $||$  为求模运算,  $\arg(\cdot)$  表示提取相位的运算,  $\omega_n$  为各阵列单元的矢量激励。

按图 3 所示,阵列所需的主瓣方向为  $\varphi_0$ 。通常我们所希望选取的单元激励可以使各阵列单元方向图在主瓣方向同相叠加,因此可选取各阵列单元的初始化激励值为:

$$\omega_n = e^{-j \cdot \arg(g_n(\varphi_0))} \quad (n=1, \dots, N) \quad (2)$$

### 2. 根据各阵列单元激励计算实际阵列方向图

将在第 1 步或第 5 步中计算出的各阵列单元激励代入式(1),即可求出阵列方向图  $f(\varphi)$ 。

### 3. 判断实际阵列方向图是否已满足要求

针对不同的应用,判断实际阵列方向图是否已满足要求的判据可以不同,因为它只是判断是否完成阵列方向图的综合,并不影响目标方向图迭代算法的具体计算。上文已假设所要求的实际方向图的副瓣峰值包络线为  $h(\varphi)$ ,现假设  $\varphi_i$  ( $i=1, \dots, I$ ) 是

第 2 步求出的实际阵列方向图的各副瓣峰值的角度坐标,那么式(3)即为本文检验实际阵列方向图是否满足要求的判据。

$$|20 \cdot \log_{10} |f(\varphi_i)| - 20 \cdot \log_{10} h(\varphi_i)| \leq 1(\text{dB}), \\ (i=1, \dots, I) \quad (3)$$

式(3)要求实际阵列方向图各副瓣峰值与目标方向图副瓣峰值包络线的绝对误差小于 1dB,当式(3)条件满足时结束阵列方向图综合。

### 4. 根据所需指标修正目标方向图

这一步是目标方向图迭代算法的核心,不同的修正方案具有不同的性能,下面介绍的是本文作者已构造的多种修正方案中性能最稳定、收敛速度最快的一种算法。当然,由于目标方向图迭代算法是由本文首次提出的,作者进行的研究还比较有限,因此应该存在着性能更好的方案。

不妨假设所需的目标方向图副瓣峰值包络线为  $h(\varphi)$ ,图 4a 是由目标方向图迭代算法第 2 步计算出的实际阵列方向图  $|f(\varphi)|$ ,图 4a 中的垂直虚线为所需目标方向图的主瓣方向  $\varphi_0$ 。下面以它为例介绍修正目标方向图的三步变换。

第一步变换是对  $f(\varphi)$  进行归一化和坐标平移,假设所需目标方向图的主瓣方向为  $\varphi_0$ ,  $f(\varphi)$  主瓣方向为  $\varphi_{\max}$ ,则归一化和坐标平移变换由式(4)给出。

$$f_1(\varphi) = f(\varphi + \varphi_{\max} - \varphi_0) / |f(\varphi_{\max})| \quad (4)$$

图 4b 给出了完成第一步变换后得到方向图  $|f_1(\varphi)|$ 。

第二步变换是根据所要求的阵列方向图副瓣峰值包络  $h(\varphi)$  与实际阵列方向图  $f_1(\varphi)$  的副瓣峰值幅度之间的误差,对  $f_1(\varphi)$  进行角度域上的拉伸和压缩变换。假设  $\varphi_i$  ( $i=1, \dots, I$ ) 是  $f_1(\varphi)$  的  $I$  个副瓣峰值的角度坐标,  $\varphi_L, \varphi_R$  分别是  $f_1(\varphi)$  主瓣左右两侧凹点的角度坐标。定义误差为各副瓣峰值与目标方向图之间相对误差的代数平均数  $\epsilon$ 。

$$\epsilon = \frac{1}{I} \sum_{i=1}^I \frac{\log_{10} |f_1(\varphi_i)| - \log_{10} h(\varphi_i)}{|\log_{10} h(\varphi_i)|} \quad (5)$$

角度域上的拉伸和压缩变换可由式(6)给出。

$$f_2(\varphi) = \begin{cases} f_1(t_1(\varphi)), & 0 \leq \varphi \leq \varphi_L \\ f_1(t_2(\varphi)), & \varphi_L \leq \varphi \leq \varphi_0 \\ f_1(t_3(\varphi)), & \varphi_0 \leq \varphi \leq \varphi_R \\ f_1(t_4(\varphi)), & \varphi_R \leq \varphi \leq 2\pi \end{cases} \quad (6)$$

其中:

$$\varphi_L = \varphi_0 - (\varphi_0 - \varphi_L) \cdot (1 + \epsilon \cdot K) \quad (7)$$

$$\varphi_R = \varphi_0 + (\varphi_R - \varphi_0) \cdot (1 + \epsilon \cdot K) \quad (8)$$

$$t_1(\varphi) = \varphi \cdot \varphi_L / \varphi_L \quad (9)$$

$$t_2(\varphi) = \varphi_0 - (\varphi_0 - \varphi) / (1 + \epsilon \cdot K) \quad (10)$$

$$t_3(\varphi) = \varphi_0 + (\varphi - \varphi_0) / (1 + \epsilon \cdot K) \quad (11)$$

$$t_4(\varphi) = 2 \cdot \pi - \frac{(2 \cdot \pi - \varphi) \cdot (2 \cdot \pi - \varphi_R)}{2 \cdot \pi - \varphi_R} \quad (12)$$

式中  $\varphi_L$ ,  $\varphi_R$  是经过拉伸与压缩后的方向图  $f_2(\varphi)$  主瓣左右两侧凹点的角度坐标,  $K$  是一个表征角度域上拉伸与压缩程度的常量,  $K$  取的越小, 方向图变形程度就越小, 收敛速度就越慢,  $K$  取的越大, 方向图变形程度就越大, 这时对于某些阵列迭代将变得不稳定, 甚至无法收敛, 实际上方向图迭代算法对  $K$  的选取是不敏感的, 这一点将在下一节予以验证, 作者曾取  $K=0.5$  计算了数十个不同的阵列, 迭代算法显示出一致的稳定性和高效性。

当式(5)所计算出的  $\epsilon$  大于 0 时, 表明副瓣峰值高于目标方向图, 式(6)所进行的变换以  $\varphi_0$  为中心对主瓣区域进行拉伸, 对副瓣区域进行压缩, 反之, 当  $\epsilon$  小于 0 时, 表明副瓣峰值低于目标方向图, 式(6)所进行的变换以  $\varphi_0$  为中心对主瓣区域进行压缩, 对副瓣区域进行拉伸。

需要注意, 在综合直线阵列时, 由于阵列方向图相对于天线阵列轴线对称, 应将式(12)以及下文式(16)中的  $2 \cdot \pi$  替换为  $\pi$ 。

图 4c 给出了完成第二步变换后得到的方向图  $|f_2(\varphi)|$ 。

第三步变换是根据所要求的阵列方向图副瓣峰值包络  $h(\varphi)$  对方向图  $f_2(\varphi)$  进行幅度域上的拉伸和压缩, 假设  $\varphi_r$  ( $r=1, \dots, I$ ) 是  $f_2(\varphi)$  的  $I$  个副瓣峰值的角度坐标, 那么由式(13)可以得出新一轮迭代所需的目标方向图  $c(\varphi)$ 。

$$c(\varphi) = f_2(\varphi) \cdot h(\varphi) / l(\varphi), \varphi \in \text{mainlobe} \quad (13)$$

其中:

$$l(\varphi) = |f_2(\varphi_r)| + [|f_2(\varphi_{r+1})| - |f_2(\varphi_r)|] \cdot (\varphi - \varphi_r) / (\varphi_{r+1} - \varphi_r), \varphi_r \leq \varphi \leq \varphi_{r+1} \quad (14)$$

如图 4c 中所示,  $l(\varphi)$  是  $\varphi_r$ ,  $\varphi_{r+1}$  两个相邻副瓣峰值之间的连线, 图 4d 给出了完成第三步变换后得到的新一轮迭代所需的目标方向图  $|c(\varphi)|$ 。

### 5. 利用空间投影算法计算阵列单元激励

由于空间分解理论可知, 上一步求出的目标方向图  $c(\varphi)$  可以分解为式(15)右式所示的两部分。

$$c(\varphi) = \sum_{n=1}^N \omega_n \cdot g_n(\varphi) + O(\varphi) \quad (15)$$

其中  $g_n(\varphi)$  与式(1)中含义相同, 为各阵列单元以坐标原点  $O$  为参考点的阵列单元方向图,  $O(\varphi)$  是目标方向图  $c(\varphi)$  中与各阵列单元方向图  $g_n(\varphi)$  ( $n=1, \dots, N$ ) 内积均为 0 的余项, 在开始迭代时, 由于选取的  $c(\varphi)$  与实际阵列所能实现的方向图相差较大, 此时余项  $O(\varphi)$  较大; 随着迭代的进行, 余项  $O(\varphi)$  将越来越小。

取  $g_n(\varphi)$  为试验函数, 对式(15)两端在  $[0, 2\pi]$  范围内进行内积, 可得:

$$\langle c(\varphi), g_n(\varphi) \rangle = \left\langle \sum_{n=1}^N \omega_n \cdot g_n(\varphi), g_n(\varphi) \right\rangle + \langle O(\varphi), g_n(\varphi) \rangle \quad (16)$$

其中  $\langle \cdot, \cdot \rangle$  代表  $\int_0^{2\pi} d\varphi$ , 由  $O(\varphi)$  的定义可知, 式(16)中右式第二项为 0。

求解式(16)即得到各阵列单元的激励矢量, 实际编程时, 由于式(16)中右式矩阵在迭代过程中保持不变, 因此只需要在程序初始化时计算一次即可。

假设计算方向图时用  $M$  个离散采样点, 本文算法的计算量主要集中在第二、四、五步, 每次迭代过程中的总计算量为三步之和, 约为  $M^* (2N+3)$  个加减运算和  $M^* (2N+5)$  个乘除运算, 文献[12]算法的计算量主要包括三部分, 第一部分是方向图计算, 与本文算法相同; 第二部分是虚拟干扰迭代; 第三部分是单元激励计算, 该步所需计算量最多, 约需  $M^* N^* N$  个加减运算和  $M^* N^* N$  个乘除运算, 文献[12]在每次迭代过程中的总计算量为三部分之和, 约为  $M^* (N^* N + N + 2)$  个加减运算和  $M^* (N^* N + N + 2)$  个乘除运算。

由此可见, 当  $N$  较大时, 本文算法的计算量近似为文献[12]算法的  $2/N$ , 极大地减少了每次迭代过程中所需的计算量。

## 三、结果讨论

例 1 为了验证正确性, 首先综合了一个阵元间距半波长的 10 元均匀直线阵, 要求的方向图为: 主瓣指向  $115^\circ$ , 副瓣归一化功率为  $-35\text{dB}^{[11]}$ , 综合时  $K$  取 0.5, 综合结果如图 5 所示, 迭代次数为 6 次, 表 1 中对照列出了 Chebyshev 方法和本文算法求得各阵元激励, 由表 1 可知, 两种算法求得阵元激

励基本一致,从而验证了本文算法的正确性。

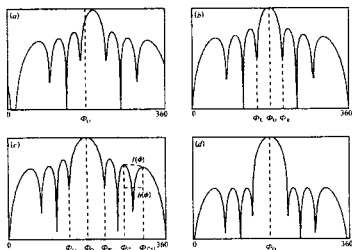


图 4 修正目标方向图的三步变换

表 1 10 元 Chebyshev 阵列单元激励

	Chebyshev 算法		本文算法	
	幅度	相位	幅度	相位
1	0.176	-0.3086	0.176	-0.3574
2	0.367	-1.636	0.367	-1.673
3	0.622	2.964	0.622	2.990
4	0.858	1.992	0.858	1.976
5	1.000	0.6638	1.000	0.6584
6	1.000	-0.6638	1.000	-0.6584
7	0.858	-1.992	0.858	-1.976
8	0.622	-2.964	0.622	-2.990
9	0.367	1.636	0.367	1.673
10	0.176	0.3086	0.176	0.3574

表 2 三种算法迭代所需的次数

阵元数目	旁瓣电平	文献[11]	文献[12]	本文算法
10	-30dB	10	7	5
10	-35dB	13	10	5
10	-40dB	32	10	6
20	-40dB	>40	15	7

例 2 表 2 中对列出了采用本文算法和采用自适应阵列综合算法<sup>[11,12]</sup>去综合一组阵列方向图<sup>[12]</sup>所需的迭代次数,综合时  $K$  取 0.5, 由表 2 可知,本文方法的迭代次数少于文献[11]和文献[12]的综合算法,考虑到本文每次迭代中的计算量约为自适应阵列综合算法的  $2/N$ , 因此利用本文方法可以极大地提高阵列方向图的综合效率。

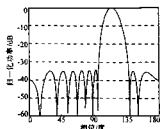


图 5 10 元直线天线阵列方向图

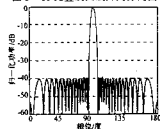


图 6 间距 0.8 波长 20 元均匀直线阵列方向图

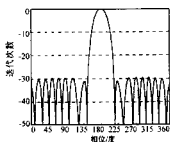


图 7 16 元圆形阵列方向图

例 3 图 6 是阵元间距 0.8 波长的 20 元均匀直线的功率方向图综合结果, 要求的方向图为: 主瓣指向  $90^\circ$ , 副瓣归一化功率为  $-40$  dB. 综合时  $K$  取 0.5, 迭代次数为 11 次. 在第一节中曾介绍过文献[10]的方法, 文献[10]方法虽然收敛速度较快, 但是它无法综合阵元平均间距超过半波长的阵列. 与之

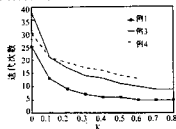


图 8  $K$  与迭代次数的关系图

相比,本文方法适用范围更广。

例 4 图 7 是一个 16 元圆形阵列,阵列半径为  $4\lambda/\pi$ , 要求的方向图为:主瓣指向  $180^\circ$ ,副瓣归一化功率为  $-30\text{dB}$ 。综合时  $K$  取 0.5,最终综合出的阵列方向图如图 7 所示。迭代次数为 14 次。

图 8 给出的是计算例 1、例 3、例 4 时,式(6)中的  $K$  取不同值所需要的迭代次数。由图 8 可知,目标方向图迭代算法对  $K$  是不敏感的,但如果  $K$  过大,迭代将会不收敛。

#### 四、结 论

本文提出了一种新型阵列综合算法:目标方向图迭代算法,这种算法通过目标方向图的迭代来调整实际方向图的形状,是一种纯数值的阵列综合算法。这种算法适用于任意结构的阵列综合,计算效率高,可以考虑阵元间互耦对阵列方向图的影响,从而满足实际工程的需要。由于这种算法是本文首次提出的,作者进行的研究还比较有限,在未来的工作中,作者将要目标方向图迭代算法中目标方向图修正方案以及其它一些细节进行更深入、细致的研究,充分发挥出目标方向图迭代算法的优点。

#### 参 考 文 献

- 1 S. E. Robert, G. J. Stern. A new technique for shaped beam synthesis of equispaced arrays. IEEE Trans.-AP, Oct., 1984; 1129~1133
- 2 D. J. Shpak. A method for the optimal pattern synthesis of linear arrays with prescribed nulls. IEEE Trans.-AP, Mar., 1996; 286~294
- 3 M. Boguasis. Antenna pattern synthesis using a relaxation method; Application to printed antennas. Electron. Letters. 1986, 7; 375~376
- 4 M. J. Buckley. Synthesis of shaped beam antenna patterns using implicitly constrained current elements. IEEE Trans.-AP, Feb., 1996; 192~197
- 5 R. Verscho. Constrained and unconstrained synthesis of array factor for circular arrays. IEEE Trans.-AP, Dec., 1995; 1405~1410
- 6 B. P. Ng. Array synthesis using a simple computer-aided approach. Electron. Letters. 5, 1990; 337~339
- 7 M. H. Er. Computer-Aided design of array pattern. Electron. Letters. 1990, 25; 2113~2115
- 8 S. L. Sim, M. H. Er. Constrained optimization technique for general array pattern synthesis. Electron. Letters. 1996, 10; 861~862
- 9 M. H. Er. Alternative approach to designing array pattern with controlled mean-square side-lobe level. Electron. Letters. 1991, 5; 435~437
- 10 C. Y. Tseng, L. J. Griffiths. A simple algorithm to achieve desired patterns for arbitrary arrays. IEEE Trans.-AP, Nov., 1992; 2737~2746
- 11 C. A. Olen, R. T. Compton. A numerical pattern synthesis algorithm for arrays. IEEE Trans.-AP, Oct., 1990; 1666~1676
- 12 W. A. Swart, J. C. Olivier. Numerical synthesis of arbitrary discrete arrays. IEEE Trans.-AP, Aug., 1993; 1171~1174
- 13 A. Tennant. Numerical pattern synthesis of difference beams in conformal arrays. Electron. Letters 1995, 8; 938~939
- 14 张志军,冯正和.考虑互耦的圆形阵列方向图综合.电波科学学报. 1997, 4; 361~368



张志军 1971 年生, 1992 年获电子科技大学学士学位, 1995 年获电子科技大学硕士学位, 现为清华大学电磁场与微波技术专业博士研究生。



冯正和 1945 年生, 清华大学电子工程系教授, 博士生导师。1970 年毕业于清华大学, 一直在该校任教, 主要从事电磁场数值方法, 微波毫米波技术及无线通信技术, 阵列天线及智能天线的教学和科学研究工作。