

# 基于波束空间的 DBF 协方差矩阵估计方法

曾浩 庞宇 杨力生

(重庆大学通信工程学院 重庆 400030)

**摘要:** 该文对波束空间数字波束合成(Digital Beamforming, DBF)的前后向平滑协方差矩阵估计算法进行了改进,利用对称阵列特殊结构,把复数运算转化为实数运算,从而减少了运算量。文章介绍了算法的原理和实现过程,并分析了运算量。而仿真表明,改进算法在减少运算量的同时,保持了阵列抑制干扰性能。

**关键词:** 数字波束合成; 协方差矩阵; 前后向平滑; 波束空间

中图分类号: TN911.7

文献标识码: A

文章编号: 1009-5896(2008)05-1100-04

## Covariance Matrix Estimation in Beamspace DBF

Zeng Hao Pang Yu Yang Li-sheng

(College of Communication Engineering, Chongqing University, Chongqing 400030, China)

**Abstract:** A modified forward-backward averaging algorithm to estimate the covariance matrix in beamspace DBF is proposed. The computation is greatly reduced because the complex number operation is replaced with real number operation with the special structure of symmetrical array. Simulation results show that the proposed algorithm can preserve its interference suppression ability with the reduced computation.

**Key words:** Digital beamforming; Covariance matrix; Forward-backward averaging; Beamspace

### 1 引言

波束合成技术,通过对接收阵列信号的复数加权,达到方向图主瓣对准期望,零陷对准干扰的效果,从而提高系统性能,在通信、雷达、声纳中广泛采用。系统中信号处理,通常都是以信号协方差矩阵为基础,完成各种形式的自适应滤波<sup>[1-3]</sup>。协方差矩阵只能利用有限快拍进行估计,而文献[4]介绍的前后向平滑估计算法相对传统前向算法,具有较小误差。基于波束空间的 DBF,通过减少矩阵维数,减少系统复杂度,一直得到不断发展<sup>[5-7]</sup>。波束空间 DBF 同样首先需要进行协方差矩阵估计,而且,如果采用前后向平滑的估计算法,可以利用文献[4]介绍的前后向平滑算法的数学特性,用实数运算代替复数运算,从而在减少误差同时,减少运算量。

### 2 波束空间协方差矩阵

波束合成方法通常是基于阵元空间的,其协方差矩阵  $\mathbf{R}$  为  $N \times N$  维。为了减少信号处理复杂度,当期望信号和干扰信号位于波束扇覆盖范围时,可以对  $\mathbf{R}$  进行降维,得到  $N_{BS} \times N_{BS}$  维波束空间协方差矩阵  $\mathbf{R}_{BS}$ , 其中  $N_{BS} < N$ 。从阵元空间到波束空间的变换,由一个  $N \times N_{BS}$  维波束空间矩阵  $\mathbf{B}$  完成,即

$$\mathbf{R}_{BS} = \mathbf{B}^H \mathbf{R} \mathbf{B} \quad (1)$$

$$\mathbf{B} = [\mathbf{b}_0 \ \mathbf{b}_1 \ \dots \ \mathbf{b}_{N_{BS}-1}] \quad (2)$$

其中波束空间矩阵  $\mathbf{B}$  由用户选择的波束扇确定,每个列矢量  $\mathbf{b}_m$  则代表了波束扇的每个子波束。波束扇选择是要求各个子波束相互正交,此时

$$\mathbf{B}^H \mathbf{B} = \mathbf{I} \quad (3)$$

其中  $\mathbf{I}$  是单位矩阵。

为了具体说明  $\mathbf{B}$  的性质,以如图 1 所示均匀线阵为例。对于线阵,只能分辨信号俯仰角  $\theta$ ,而对于水平角  $\phi$  是无法分辨的。所以,阵列的方向矢量  $\mathbf{v}(\theta)$  将只含有  $\theta$  一个变量,并且方向矢量可以用

$$u = \cos \theta \quad (4)$$

表示。此时,由于阵元间距为信号半个波长,方向矢量为

$$\mathbf{v}(u) = \begin{bmatrix} e^{j\left(0-\frac{N-1}{2}\right)\pi u} \\ \dots \\ e^{j\left(N-1-\frac{N-1}{2}\right)\pi u} \end{bmatrix} \quad (5)$$

从而,波束扇可以选择为

$$\mathbf{b}_m = \frac{1}{\sqrt{N}} v \left[ \left( \frac{N_{BS}-1}{2} - m \right) \frac{2}{N} \right] \quad (6)$$

其中  $m = 0, \dots, N_{BS}-1$ 。以 32 阵元的标准线阵为例,如果选择  $N_{BS} = 9$ ,则图 2 为波束扇示意图。图中的每一个子波束,对应于以  $\mathbf{b}_m$  为权矢量的阵列方向图。

### 3 前后向平滑协方差矩阵估计

阵列信号处理的各种算法,往往都是对协方差矩阵的处理,比如协方差矩阵的各种分解等等。但协方差矩阵只能通过有限次快拍  $\mathbf{x}(k)$  估计得到,其中  $\mathbf{x}(k)$  是一个  $N$  维列矢

2007-04-02 收到, 2007-09-24 改回  
重庆市自然科学基金(2005BB2195)和重庆大学自然科学基金(2007CD056)资助课题

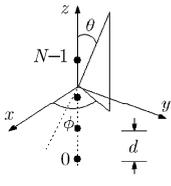


图1 均匀线阵

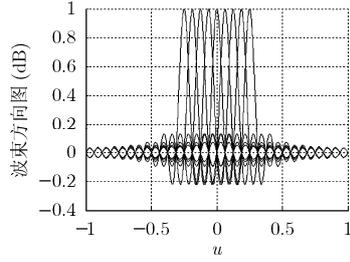


图2 波束扇示意图

量。传统前向协方差矩阵估计为

$$\hat{\mathbf{R}} = \frac{1}{K} \sum_{k=0}^{K-1} \mathbf{x}(k)\mathbf{x}^H(k) = \mathbf{X}\mathbf{X}^H \quad (7)$$

数据矩阵

$$\mathbf{X} = \frac{1}{\sqrt{K}} [\mathbf{x}(0) \cdots \mathbf{x}(K-1)] \quad (8)$$

而如果阵列几何上满足对称结构,即阵列任意一个阵元位置都同另一个阵元位置关于原点对称,通常均匀线阵,均匀圆阵,均匀矩形阵都满足该条件,则协方差矩阵是中心赫密特矩阵<sup>[8]</sup>。

$$\mathbf{R} = \mathbf{J}\mathbf{R}^*\mathbf{J} \quad (9)$$

其中,矩阵 $\mathbf{J}$ 是置换矩阵,其次对角线元素为1,其余为0。

$$\mathbf{J} = \begin{bmatrix} 0 & 0 & 1 \\ 0 & \dots & 0 \\ 1 & 0 & 0 \end{bmatrix} \quad (10)$$

此时,可以采用前后向平滑估计算法

$$\begin{aligned} \hat{\mathbf{R}} &= \frac{1}{2K} \left[ \sum_{k=0}^{K-1} \mathbf{x}(k)\mathbf{x}^H(k) + \sum_{k=0}^{K-1} \mathbf{J}\mathbf{x}^*(k)\mathbf{x}^T(k)\mathbf{J} \right] \\ &= \frac{1}{2} (\mathbf{X}\mathbf{X}^H + \mathbf{J}\mathbf{X}^*\mathbf{X}^T\mathbf{J}) \end{aligned} \quad (11)$$

前后向平滑算法较前向算法,具有更小误差,但它的实现,通常并不是通过式(11)直接计算,而是采用如下步骤进行,这样可以减少运算量,只是此时 $N$ 应为偶数<sup>[9]</sup>。

第1步 把数据矩阵 $\mathbf{X}$ 表示为两个子矩阵,即

$$\mathbf{X} = \begin{bmatrix} \mathbf{X}_1 \\ \mathbf{X}_2 \end{bmatrix} \quad (12)$$

第2步 计算矩阵 $\mathbf{Z}$

$$\mathbf{Z} = \begin{bmatrix} (\mathbf{X}_1 + \mathbf{J}\mathbf{X}_2)_R & -(\mathbf{X}_1 + \mathbf{J}\mathbf{X}_2)_I \\ (\mathbf{X}_1 - \mathbf{J}\mathbf{X}_2)_I & (\mathbf{X}_1 - \mathbf{J}\mathbf{X}_2)_R \end{bmatrix} \quad (13)$$

式中下标 $R$ 和 $I$ 分别代表矩阵的实部和虚部。

第3步 计算中间矩阵 $\mathbf{Q}$

$$\mathbf{Q} = \mathbf{Z}\mathbf{Z}^T \quad (14)$$

第4步 计算前后向平滑协方差矩阵

$$\hat{\mathbf{R}} = \mathbf{F}\mathbf{Q}\mathbf{F}^H \quad (15)$$

其中 $N \times N$ 维矩阵 $\mathbf{F}$ 定义为

$$\mathbf{F} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I} & \mathbf{I} \\ \mathbf{J} & -\mathbf{J} \end{bmatrix} \quad (16)$$

## 4 波束空间前后向平滑协方差矩阵

当采用波束空间进行波束合成处理时,协方差矩阵同样通过估计得到。估计方法在首先根据式(2),式(6)确定波束空间矩阵 $\mathbf{B}$ 前提下,可以采用式(1),式(7)的前向协方差估计算法,也可以采用式(1),式(11)的前后向平滑协方差矩阵估计算法。为了使估计误差更小,此处仅仅考虑后者。

### 4.1 矩阵 $\mathbf{R}_{BS}$ 实数性

引理<sup>[4]</sup> 如果矩阵 $\mathbf{A}$ 和矩阵 $\mathbf{E}$ 是列共轭对称或者反对称的,即满足 $\mathbf{A} = \mathbf{J}\mathbf{A}^*$ 和 $\mathbf{E} = \mathbf{J}\mathbf{E}^*$ 或者 $\mathbf{A} = -\mathbf{J}\mathbf{A}^*$ 和 $\mathbf{E} = -\mathbf{J}\mathbf{E}^*$ ,则中心赫密特矩阵 $\mathbf{R}$ 与二者乘积 $\mathbf{A}^H\mathbf{R}\mathbf{E}$ 是实数矩阵。

根据上述性质,考虑在波束空间的协方差矩阵式(1)表达,只要 $\mathbf{B}$ 满足列共轭对称或者反对称条件, $\mathbf{R}_{BS}$ 就是实数矩阵。

证明 假设第 $i$ 个阵元位置坐标矢量为 $\mathbf{p}_i$ ,根据对称阵列特点,必然存在一个阵元 $j$ ,其坐标矢量 $\mathbf{p}_j$ 满足

$$\mathbf{p}_j = -\mathbf{p}_i \quad (17)$$

同时,阵列波数矢量为 $\mathbf{k}$ ,则阵列方向矢量可以表示为

$$\mathbf{v}(\mathbf{k}) = \begin{bmatrix} e^{-j\mathbf{k}\mathbf{p}_0} \\ \dots \\ e^{-j\mathbf{k}\mathbf{p}_{N-1}} \end{bmatrix} \quad (18)$$

根据波束空间矩阵定义式(6),只需要证明矩阵

$$\mathbf{V} = [\mathbf{v}(\mathbf{k}_0) \cdots \mathbf{v}(\mathbf{k}_{N_{BS}-1})] \quad (19)$$

的列共轭对称性。根据式(17)和式(18)

$$\begin{aligned} \mathbf{J}\mathbf{V}^* &= [\mathbf{J}\mathbf{v}^*(\mathbf{k}_0) \cdots \mathbf{J}\mathbf{v}^*(\mathbf{k}_{N_{BS}-1})] = \begin{bmatrix} e^{j\mathbf{k}_0\mathbf{p}_{N-1}} & \dots & e^{j\mathbf{k}_{N_{BS}-1}\mathbf{p}_{N-1}} \\ \dots & \dots & \dots \\ e^{j\mathbf{k}_0\mathbf{p}_0} & \dots & e^{j\mathbf{k}_{N_{BS}-1}\mathbf{p}_0} \end{bmatrix} \\ &= \begin{bmatrix} e^{-j\mathbf{k}_0\mathbf{p}_0} & \dots & e^{-j\mathbf{k}_{N_{BS}-1}\mathbf{p}_0} \\ \dots & \dots & \dots \\ e^{-j\mathbf{k}_0\mathbf{p}_{N-1}} & \dots & e^{-j\mathbf{k}_{N_{BS}-1}\mathbf{p}_{N-1}} \end{bmatrix} = \mathbf{V} \end{aligned} \quad (20)$$

由此可见

$$\mathbf{J}\mathbf{B}^* = \mathbf{B} \quad (21)$$

从而

$$\mathbf{R}_{BS} = \text{Re}(\mathbf{B}^H\mathbf{R}\mathbf{B}) \quad (22)$$

### 4.2 基本算法

在阵列几何结构是对称前提下,其协方差矩阵是中心赫密特矩阵,可以采用前后向平滑估计,即通过第3节中介绍的4个步骤,完成对协方差矩阵 $\hat{\mathbf{R}}$ 的求解。而在利用式(1)进行波束空间协方差矩阵求解时,考虑 $\hat{\mathbf{R}}_{BS}$ 是实数矩阵的性质,可以进行运算简化。

$$\begin{aligned} \hat{\mathbf{R}}_{BS} &= \mathbf{B}^H\hat{\mathbf{R}}\mathbf{B} = \text{Re} \left[ (\mathbf{B}_R + j\mathbf{B}_I)^H (\hat{\mathbf{R}}_R + j\hat{\mathbf{R}}_I) (\mathbf{B}_R + j\mathbf{B}_I) \right] \\ &= \mathbf{B}_R^T\hat{\mathbf{R}}_R\mathbf{B}_R + \mathbf{B}_I^T\hat{\mathbf{R}}_R\mathbf{B}_I - \mathbf{B}_R^T\hat{\mathbf{R}}_I\mathbf{B}_I + \mathbf{B}_I^T\hat{\mathbf{R}}_I\mathbf{B}_R \end{aligned} \quad (23)$$

显然, 此时的计算只需考虑实数部分, 而无需对虚部进行计算, 从而使运算量减少一半。

### 4.3 算法改进

如果考虑  $\widehat{\mathbf{R}}$  的计算方法,  $\widehat{\mathbf{R}}_{\text{BS}}$  运算还可以进一步简化。由于

$$\widehat{\mathbf{R}}_{\text{BS}} = \text{Re}(\mathbf{B}^H \widehat{\mathbf{R}} \mathbf{B}) = \text{Re}(\mathbf{B}^H \mathbf{F} \mathbf{Q} \mathbf{F}^H \mathbf{B}) \quad (24)$$

而当  $N$  为偶数时, 矩阵

$$\mathbf{B} = \begin{bmatrix} \mathbf{A} \\ \mathbf{J}\mathbf{A}^* \end{bmatrix} \quad (25)$$

则根据  $\mathbf{F}$  定义

$$\mathbf{F}^H \mathbf{B} = \frac{1}{\sqrt{2}} \begin{bmatrix} \mathbf{I} & \mathbf{J} \\ -j\mathbf{I} & j\mathbf{J} \end{bmatrix} \begin{bmatrix} \mathbf{A} \\ \mathbf{J}\mathbf{A}^* \end{bmatrix} = \sqrt{2} \begin{bmatrix} \mathbf{A}_R \\ \mathbf{A}_I \end{bmatrix} \quad (26)$$

代入式(24)

$$\widehat{\mathbf{R}}_{\text{BS}} = 2 \begin{bmatrix} \mathbf{A}_R \\ \mathbf{A}_I \end{bmatrix}^T \mathbf{Q} \begin{bmatrix} \mathbf{A}_R \\ \mathbf{A}_I \end{bmatrix} \quad (27)$$

相对于基本算法, 改进算法通过矩阵  $\mathbf{Q}$  直接计算出  $\widehat{\mathbf{R}}_{\text{BS}}$ , 用式(27)代替了式(15)和式(23), 这在步骤上得到了简化。由于没有  $\mathbf{F}$  参与计算, 不涉及复数运算问题。这些, 都使得运算量大大减少。

## 5 运算量分析

### 5.1 前向估计算法

波束空间协方差矩阵估计可以采用前向估计算法, 此时首先是计算式(7), 然后是计算式(1)。由于每个快拍  $\mathbf{x}(k)$  是一个复数构成的  $N$  维列矢量, 所以, 式(1)运算量为  $KN^2$  次复数乘法,  $(K-1)N^2$  复数加法。考虑 1 次复数乘法等效于 4 次实数乘法和两次实数加法, 一次复数加法等于两次实数加法, 则式(7)运算量为

$$\left. \begin{array}{l} 4KN^2 \otimes \\ 4KN^2 - 2N^2 \oplus \end{array} \right\} \quad (28)$$

其中  $\otimes$  表示实数乘法,  $\oplus$  表示实数加法。而对于式(1), 为 3 个复数矩阵乘积, 根据各个矩阵维数, 运算量为

$$\left. \begin{array}{l} 4NN_{\text{BS}}(N + N_{\text{BS}}) \otimes \\ 2N_{\text{BS}}(N^2 + NN_{\text{BS}} + N + N_{\text{BS}}) \oplus \end{array} \right\} \quad (29)$$

结合上述结果, 前向估计算法总的运算量为

$$\left. \begin{array}{l} 4KN^2 + 4NN_{\text{BS}}(N + N_{\text{BS}}) \otimes \\ 4KN^2 - 2N^2 + 2N_{\text{BS}}(N^2 + NN_{\text{BS}} + N + N_{\text{BS}}) \oplus \end{array} \right\} \quad (30)$$

### 5.2 前后向平滑基本算法

如果采用前后向平滑算法, 即首先根据第 3 节中介绍的 4 个步骤得到  $\widehat{\mathbf{R}}$ , 再通过式(23)得到  $\widehat{\mathbf{R}}_{\text{BS}}$ 。对于  $\widehat{\mathbf{R}}$ , 其运算量为

$$\left. \begin{array}{l} 2KN^2 \otimes \\ 4NK + (2K+1)N^2 \oplus \end{array} \right\} \quad (31)$$

而对于式(23)是 4 个因子之和, 它的运算量为

$$\left. \begin{array}{l} 4N^2N_{\text{BS}} + 4NN_{\text{BS}}^2 \otimes \\ 4N_{\text{BS}}(N-1)(N + N_{\text{BS}}) + 3N_{\text{BS}}^2 \oplus \end{array} \right\} \quad (32)$$

于是, 总的运算量为

$$\left. \begin{array}{l} 2KN^2 + 4NN_{\text{BS}}(N + N_{\text{BS}}) \otimes \\ 4NK + (2K+1)N^2 + 4N_{\text{BS}}(N^2 + NN_{\text{BS}} - N - N_{\text{BS}}) + 3N_{\text{BS}}^2 \oplus \end{array} \right\} \quad (33)$$

### 5.3 改进的算法

对于进一步改进的波束空间前后向平滑协方差矩阵估计算法, 首先通过第 3 节中介绍的方法得到矩阵  $\mathbf{Q}$ , 然后利用式(27)计算  $\widehat{\mathbf{R}}_{\text{BS}}$ 。对于得到  $\mathbf{Q}$ , 所需运算量为

$$\left. \begin{array}{l} 2KN^2 \otimes \\ 4NK + (2K-1)N^2 \oplus \end{array} \right\} \quad (34)$$

而式(27)运算量为

$$\left. \begin{array}{l} N^2N_{\text{BS}} + NN_{\text{BS}}^2 \otimes \\ N_{\text{BS}}(N-1)(N + N_{\text{BS}}) \oplus \end{array} \right\} \quad (35)$$

总的运算量为

$$\left. \begin{array}{l} 2KN^2 + NN_{\text{BS}}(N + N_{\text{BS}}) \otimes \\ 4NK + (2K-1)N^2 + N_{\text{BS}}(N^2 + NN_{\text{BS}} - N - N_{\text{BS}}) \oplus \end{array} \right\} \quad (36)$$

从上述表达式可以看出, 不论是加法运算量还是乘法运算量, 都是改进算法性能最好, 前向估计算法最差。

## 6 仿真

假设阵元数  $N = 32$ , 而波束选择  $N_{\text{BS}} = 9$ , 利用式(30), 式(33), 式(36), 图 3 用曲线方式, 描绘了估计采用快拍数同计算量的关系。图 3(a)表示乘法运算量, 图 3(b)表示加法运算量。进一步可以知道, 相对于前向平滑估计, 基本前后向平滑运算量只有其 50%, 改进算法则只有其 40%。

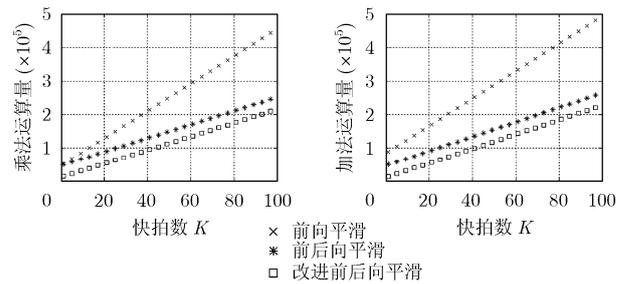


图 3 3 种算法运算量比较

对于同样的阵列条件, 从图 2 所示波束扇范围可以看出, 期望和干扰信号的角度范围只能在  $-0.25 < u < 0.25$  之间。假设期望信号  $u_T = 0$ , 信噪比  $\text{SNR} = 20\text{dB}$ , 存在一个干扰信号  $u_i = 0.15$ , 干扰噪声比  $\text{INR} = 10\text{dB}$ 。图 4 为采用改进前后向平滑估计算法时, 阵列方向图。从图中可以看出, 主瓣对准了期望, 零陷对准干扰, 实现了波束合成目的, 说明

运算量减少并没有牺牲阵列性能。

最后,考虑采用蒙特卡罗仿真方法比较波束空间前向平滑估计算法和前后向平滑估计算法误差性能,假设阵列条件和信号条件与上文相同。定义参数为归一化阵列输出信干噪比

$$\rho = \frac{\widehat{\text{SINR}}}{\text{SINR}} \quad (37)$$

其中 SINR 为协方差矩阵无误差时情况,而  $\widehat{\text{SINR}}$  为用有限快拍估计得到协方差矩阵时的情况。由于  $\rho$  是一个变量,所以仿真比较不同情况下  $\rho$  的均值,如图 5 所示。从图中可以看出两点:一是在相同快拍数时,前后向平滑算法比前向算法误差要小,从而输出 SINR 更大,尤其在快拍数较少时;二是说明运算量减少没有牺牲前后向平滑估计算法在误差上的优势。

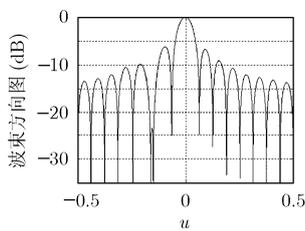


图 4 波束合成方向图

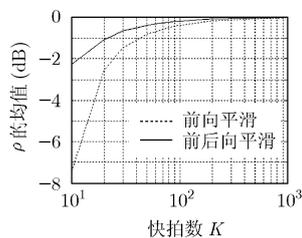


图 5 两种估计算法误差比较

## 7 结束语

对于基于波束空间的波束合成器,在进行协方差矩阵估计时,利用波束矩阵的列共轭对称性和前后向平滑估计算法的特殊性质,可以使复数运算转化为实数运算,从而减少了运算量。

## 参考文献

[1] Godara L C. Applications of antenna arrays to mobile

communications, part I: Performance improvement, feasibility, and system considerations. *Proc. of the IEEE*, 1997, 85(7): 1029-1030.

- [2] Godara L C. Application of antenna arrays to mobile communications, part II: Beam-forming and direction-of-arrival considerations. *Proc. of the IEEE*, 1997, 85(8): 1195-1245.
- [3] Krim H and Viberg M. Two decades of array signal processing research. *IEEE Signal Processing Magazine*, 1996, 13(4): 67-94.
- [4] Linebarger D A, Degroat R D, and Dowling E M. Efficient direction-finding methods employing forward/backward averaging. *IEEE Trans. on Signal Processing*, 1994, 42(8): 2136-2145.
- [5] Amini A N and Georgiou T T. Avoiding ambiguity in beamspace processing. *IEEE Singal Processing Letters*, 2005, 12(5): 372-375.
- [6] Hassanien A, Elkader S A, and Gershman A B. Convex optimization based beam-space preprocessing with improved robustness against out-of-sector sources. *IEEE Trans. on Signal Processing*, 2006, 54(5): 1587-1595.
- [7] Belloni F and Koivunen V. Beamspace transform for UCA: error analysis and bias reduction. *IEEE Trans. on Signal Processing*, 2006, 54(8): 3078-3089.
- [8] 张贤达. 矩阵分析与应用. 北京: 清华大学出版社, 2004: 133-136.
- [9] Van Trees L. Optimum Array Processing. New York: Jhon Wiley, 2002: 722-725.

曾浩: 男, 1977年生, 讲师, 博士, 研究方向为阵列信号处理。

庞宇: 男, 1981年生, 硕士生, 研究方向为空间谱估计。

杨力生: 男, 1973年生, 教授, 博士生导师, 研究方向为无线通信技术。