



阵列信号处理大作业

题目：SBL/CNN/CBF/Capon/MUSIC/ESPR-
IT/OMP 的 DOA 估计性能比较

姓 名	黄楚涵
学 号	22215977
院 系	电子与通信工程学院
专 业	电子信息
教 师	段克清 副教授

2023 年 3 月 31 日

第 1 章 题目背景

阵列信号处理具有波束控制灵活、信号增益高、空间分辨率高以及干扰抑制能力强等优点，自提出以来得到了快速发展和广泛应用。其基本原理是将多个传感器放置在空间中不同位置构成一个传感器阵列，由传感器阵列对空间电磁或者声学信号进行空域并行采样，然后再通过处理器进行信号处理，从中得到信号有用信息的理论和技术。阵列信号处理包括两方面内容，一方面是自适应信号处理，利用信号的空域特性增强有用信号方向上的能量，对其它方向上的干扰和噪声信号进行抑制。另一方面则是空间谱估计技术，也称为波达方向估计（Direction Of Arrival, DOA）、角度估计、方向估计或测向等。

最早基于阵列的空间谱估计方法是常规波束形成（Conventional Beamforming, CBF）法，该方法本质上是时域傅里叶谱估计方法向空域的简单扩展，原理简单，但分辨能力有限，会受到阵列瑞利限的限制。要提高对目标的分辨能力，就需要实现超分辨，这是空间谱估计中的重要研究方向。最典型的超分辨谱估计方法是多重信号分类（Multiple Signal Classification, MUSIC）方法以及旋转不变子空间（Estimation of Signal Parameters via Rotational Invariance Technique, ESPRIT）方法，这两种方法基于信号子空间，利用信号子空间的性质，实现了超分辨测向。这两类算法均需要较多独立同分布的快拍数据，一般要求快拍数几倍于阵元数，不足之处是要有目标数量的先验信息。

本次作业主要实现对几种谱估计算法的仿真，包括 CBF 算法、Capon 算法、MUSIC 算法、ESPRIT 算法、OMP 算法、SBL 算法以及 CNN 算法。对七种算法的性能进行对比。

第2章 空间谱估计理论

2.1 信号波达角

信号波达角 (DOA) 为空间信号的到达方向。在三维空间中, 接收信号的 DOA 信息为 (θ, ϕ) , θ 为方位角, ϕ 为俯仰角, 两者的取值范围为 $\theta \in [0, 360^\circ)$, $\phi \in [0, 90^\circ]$ 。为方便后续仿真, 我们在这里建立 DOA 检测的一维均匀天线阵列模型。在远场的假设下可以近似认为各阵元接收到的信号功率一样, 只有相位信息不同。故有具体数学模型如下。

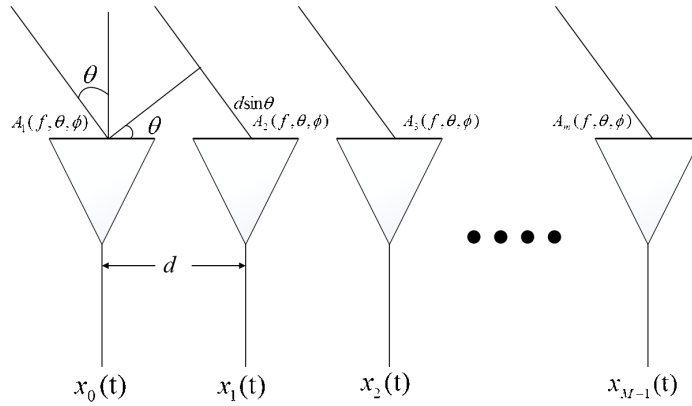


图 2-1 转发式欺骗干扰示意图

如图 2-1, L 阵元一维均匀天线阵列水平排列, 假设 L 阵元各向同性, 各个阵元间隔为 d , 忽略任何干扰, 则第 k 个信号源 $s_k(t)$ 的入射角为 $\theta_k \in [-\frac{\pi}{2}, \frac{\pi}{2}]$, $k = 1, 2, \dots, K$, 则第 l 个阵元所接收到的信号为:

$$x_l(t) = \sum_{k=1}^K s_k(t) e^{-j \frac{2\pi}{\lambda} (l-1) d \sin \theta_k} + n_l(t), \quad l = 1, 2, \dots, L \quad (2.1)$$

其中, λ 为信号波长, 其中 $n_l(t)$ 表示第 l 个阵元在 t 时刻所接收的满足 $N(0, \sigma^2)$ 分布的加性高斯白噪声。将信号模型 2.1 写成矩阵形式:

$$\begin{bmatrix} x_1(t) \\ x_2(t) \\ \vdots \\ x_L(t) \end{bmatrix} = \begin{bmatrix} e^{-j \frac{2\pi}{\lambda} (1-1) d \sin \theta_1} & e^{-j \frac{2\pi}{\lambda} (1-1) d \sin \theta_2} & \dots & e^{-j \frac{2\pi}{\lambda} (1-1) d \sin \theta_K} \\ e^{-j \frac{2\pi}{\lambda} (2-1) d \sin \theta_1} & e^{-j \frac{2\pi}{\lambda} (2-1) d \sin \theta_2} & \dots & e^{-j \frac{2\pi}{\lambda} (2-1) d \sin \theta_K} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-j \frac{2\pi}{\lambda} (L-1) d \sin \theta_1} & e^{-j \frac{2\pi}{\lambda} (L-1) d \sin \theta_2} & \dots & e^{-j \frac{2\pi}{\lambda} (L-1) d \sin \theta_K} \end{bmatrix} \begin{bmatrix} s_1(t) \\ s_2(t) \\ \vdots \\ s_K(t) \end{bmatrix} + \begin{bmatrix} n_1(t) \\ n_2(t) \\ \vdots \\ n_L(t) \end{bmatrix} \quad (2.2)$$

式2.2的矢量形式是:

$$\mathbf{x}(t) = \mathbf{A}\mathbf{s}(t) + \mathbf{n}(t) \quad (2.3)$$

其中 $\mathbf{x}(t)$, $\mathbf{s}(t)$ 分别为阵元接收快拍数据矢量和信号源矢量, $\mathbf{n}(t)$ 为加性高斯白噪声矢量, \mathbf{A} 表示信号源对均匀天线阵列的导向矩阵。

导向矩阵第 l 行、第 k 列元素表示第 l 个阵元对第 k 个信号的响应系数:

$$a_{l,k} = e^{-j\frac{2\pi}{\lambda}(l-1)d\sin\theta_k} \quad (2.4)$$

L 阵元天线阵列对第 k 个信号的导向矢量:

$$\mathbf{a}_k = \mathbf{a}(\theta_k) = \begin{bmatrix} e^{-j\frac{2\pi}{\lambda}(1-1)d\sin\theta_k} \\ e^{-j\frac{2\pi}{\lambda}(2-1)d\sin\theta_k} \\ \vdots \\ e^{-j\frac{2\pi}{\lambda}(L-1)d\sin\theta_k} \end{bmatrix}, \quad k = 1, 2, \dots, K \quad (2.5)$$

由式2.4可知, $a_{l,k}$ 只与阵元位置和信号入射角有关。不同位置的阵元在接收信号时存在波程差, 由于阵元间的距离远小于阵元到信号源的距离, 因此我们可以利用不同阵元接收到的相位差估计 DOA。

2.2 CBF 算法原理

由于2.3 中空域导向矩阵 $\mathbf{A}(\theta)$ 的作用, 接收信号中含不同的相位差, 因而如果设置权重矢量 \mathbf{w} 补偿每个阵元接收信号的相位差, 就能实现 $\mathbf{s}(n)$ 的同相累加, 提高信噪比, 有利于从低信噪比条件下提取出信号。此时权重矢量为

$$\mathbf{w} = \mathbf{a}(\theta) = \left[1, e^{-j\frac{2\pi d \sin \theta}{\lambda}}, \dots, e^{-j\frac{2\pi(M-1)d \sin \theta}{\lambda}} \right]^T \quad (2.6)$$

阵列输出信号为

$$\mathbf{y}(n) = \mathbf{w}^H \mathbf{x}(n) = \sum_{m=1}^M w_m^* x_m(n) \quad (2.7)$$

阵列输出功率为

$$P(\mathbf{w}) = E \{ |\mathbf{y}(n)|^2 \} = \mathbf{w}^H \mathbf{R} \mathbf{w} = \mathbf{a}^H(\theta) \mathbf{R} \mathbf{a}(\theta), \quad (2.8)$$

其中 $\mathbf{R} = E \{ \mathbf{x}(n) \mathbf{x}^H(n) \}$ 为阵列接收信号的自相关矩阵。CBF 算法可以看做是

对回波数据做空域傅里叶变换, 得到数据在各个方向的功率值, 实现各个角度的相参积累。

2.3 Capon 算法原理

Capon 法也被称为最小方差无失真响应 (Minimum Variance Distortionless Response, MVDR) 法。Capon 法对权重矢量 \mathbf{w} 的要求是: 保证某确定方向 θ_d 信号能被正确接收, 而其他入射方向的信号或干扰被完全抑制, 即

$$\begin{cases} \min_{\mathbf{w}} \mathbf{w}^H \mathbf{R} \mathbf{w} \\ \text{s.t. } \mathbf{w}^H \mathbf{a}(\theta_d) = 1 \end{cases} \quad (2.9)$$

通过拉格朗日乘子法求解的权重矢量为

$$\mathbf{w} = \frac{\mathbf{R}^{-1} \mathbf{a}(\theta_d)}{\mathbf{a}^H(\theta_d) \mathbf{R}^{-1} \mathbf{a}(\theta_d)}, \quad (2.10)$$

其中 $\mathbf{R} = E\{\mathbf{x}(\mathbf{n})\mathbf{x}^H(\mathbf{n})\}$ 为阵列接收信号的自相关矩阵。阵列输出功率为

$$P = \mathbf{w}^H \mathbf{R} \mathbf{w} = \frac{1}{\mathbf{a}^H(\theta_d) \mathbf{R}^{-1} \mathbf{a}(\theta_d)}, \quad (2.11)$$

实际中, 由于目标信号来向 θ_d 是未知的, 因而需要对来向进行扫描, 计算得到谱曲线

$$P(\theta) = \frac{1}{\mathbf{a}^H(\theta) \mathbf{R}^{-1} \mathbf{a}(\theta)} \quad (2.12)$$

2.4 MUSIC 算法原理

根据式2.3, 一维 L 元 ULA 接收 K 个信号源的信号模型为:

$$\mathbf{x}(t) = \mathbf{A} \mathbf{s}(t) + \mathbf{n}(t) \quad (2.13)$$

$\mathbf{x}(t)$ 的协方差矩阵 $\mathbf{R}_{xx} \in \mathbb{C}^{L \times L}$ 可表示为:

$$\mathbf{R}_{xx} = E[\mathbf{x}(t)\mathbf{x}^H(t)] = \mathbf{A} \mathbf{S} \mathbf{A}^H + \mathbf{R}_N \quad (2.14)$$

$$\mathbf{S} = E[\mathbf{s}(t)\mathbf{s}^H(t)] \quad (2.15)$$

$$\mathbf{R}_N = E[\mathbf{n}(t)\mathbf{n}^H(t)] \quad (2.16)$$

其中 $E[\cdot]$, $[\cdot]^H$ 分别表示矩阵元素均值与矩阵的共轭转置运算, \mathbf{S} 为信号源矢量 \mathbf{R}_N 为噪声矢量的协方差矩阵; 由于 $\mathbf{n}_l(t), l = 1, 2, \dots, L$ 是同分布、不相干的满足 $N(0, \sigma^2)$ 分布的加性高斯白噪声, 因此 \mathbf{R}_N 可表示为:

$$\mathbf{R}_N = \sigma^2 \mathbf{I}_L \quad (2.17)$$

其中 $\mathbf{I} \in \mathbb{C}^{L \times L}$ 为单位矩阵, 则式2.14可表示为:

$$\mathbf{R}_{xx} = E[\mathbf{x}(t)\mathbf{x}^H(t)] = \mathbf{A}\mathbf{S}\mathbf{A}^H + \sigma^2 \mathbf{I}_L \quad (2.18)$$

由于信号与噪声两者的子空间的正交性, 我们可对2.18的 \mathbf{R}_{xx} 进行特征分解:

$$\mathbf{R}_{xx} = \mathbf{U}_S \mathbf{\Sigma}_S \mathbf{U}_S^H + \mathbf{U}_N \mathbf{\Sigma}_N \mathbf{U}_N^H \quad (2.19)$$

式2.18中的 $\mathbf{\Sigma}_S \in \mathbb{C}^{K \times K}$ 为 \mathbf{R}_{xx} 的 L 个特征值中的最大的 K 个特征值组成的对角阵:

$$\mathbf{\Sigma}_S = \text{diag}(\lambda_1, \lambda_2, \dots, \lambda_K) \quad (2.20)$$

其对应的特征矢量阵 $\mathbf{U}_S \in \mathbb{C}^{L \times K}$ 张成的 $\text{span}(\mathbf{U}_S)$ 是信号子空间。式(2.10)中的 $\mathbf{\Sigma}_N \in \mathbb{C}^{(L-K) \times (L-K)}$ 为 \mathbf{R}_{xx} 的 L 个特征值中的最小的 $L - K$ 个特征值组成的对角阵:

$$\mathbf{\Sigma}_N = \text{diag}(\lambda_{K+1}, \lambda_{K+2}, \dots, \lambda_L) \quad (2.21)$$

其对应的特征矢量阵 $\mathbf{U}_N \in \mathbb{C}^{L \times (L-K)}$ 张成的 $\text{span}(\mathbf{U}_N)$ 是噪声子空间。根据式2.17:

$$\mathbf{\Sigma}_N = \text{diag}(\lambda_{K+1}, \lambda_{K+2}, \dots, \lambda_L) = \text{diag}(\sigma^2, \sigma^2, \dots, \sigma^2) = \sigma^2 \mathbf{I}_L \quad (2.22)$$

综上所述, 2.20-2.22 中的特征值满足:

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_K > \lambda_{K+1} = \lambda_{K+2} = \dots = \lambda_L = \sigma^2 \quad (2.23)$$

由于噪声子空间与信号子空间导向矢量 $\mathbf{a}(\theta_k) \in \mathbb{C}^{L \times 1}$ 也具有正交性:

$$\mathbf{a}^H(\theta_k) \mathbf{U}_N = \mathbf{0}, \quad k = 1, 2, \dots, K \quad (2.24)$$

由于 ULA 接收的待处理数据是有限长的; 因此无法得到准确的 \mathbf{U}_N , 所以我们只能利用数据协方差矩阵 \mathbf{R}_{xx} 来进行 DOA 估计。令 $L \times L$ 维阵列接收数据 \mathbf{R}_{xx} 的最大似然估计 $\hat{\mathbf{R}}_{xx}$ 为:

$$\hat{\mathbf{R}}_{xx} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}(t) \mathbf{x}^H(t) \quad (2.25)$$

其中, N 为采样的快拍数。对 $\hat{\mathbf{R}}_{xx}$ 进行特征分解, 可得到式 2.19 -2.22 中的信号与噪声空间的特征值以及特征向量的估计; 令噪声子空间特征矢量矩阵的估计为 $\hat{\mathbf{U}}_N$ 。由于噪声和有限长数据截断造成的误差, 信号子空间导向矢量 $\mathbf{a}^H(\theta_k)$ 与 $\hat{\mathbf{U}}_N$ 并不完全正交, 也即式 2.21 不完全等于 $\mathbf{0}$ 而仅仅是非常接近。因此 MUSIC 算法进行 DOA 估计以最小化搜索实现:

$$\theta_{k, \text{MUSIC}} = \underset{\theta}{\operatorname{argmin}} \mathbf{a}^H(\theta) \hat{\mathbf{U}}_N \hat{\mathbf{U}}_N^H \mathbf{a}(\theta), \quad k = 1, 2, \dots, K \quad (2.26)$$

取倒数转写成:

$$\theta_{k, \text{MUSIC}} = \underset{\theta}{\operatorname{argmax}} \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{U}}_N \hat{\mathbf{U}}_N^H \mathbf{a}(\theta)}, \quad k = 1, 2, \dots, K \quad (2.27)$$

定义:

$$P_{\text{MUSIC}}(\theta) = \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{U}}_N \hat{\mathbf{U}}_N^H \mathbf{a}(\theta)}, \quad \theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right] \quad (2.28)$$

为 MUSIC-DOA 估计谱。当由入射角搜索区间角 θ 的确定的导向矢量 $\mathbf{a}(\theta)$ 属于信号子空间时, 也就是该入射角是 DOA 的估计入射角参数, 式 2.28 的分母趋近于 0 ; 而当其不属于信号子空间时, 式 2.28 的分母是一个不为 0 的值。因此式 2.28 给出的 DOA 估计谱 P_{MUSIC} 会在区间 $\theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ 的 DOA 估计方向产生尖峰, 而在非入射方向相对平坦。MUSIC-DOA 的算法流程如下:

算法 1: MUSIC-DOA 估计算法

输入: 快拍数据矢量 $\mathbf{x}(n) \in \mathbb{C}^{L \times N}$ 和信号源数 K

- 1 由阵列接收数据得到数据协方差矩阵 $\hat{\mathbf{R}}_{xx} = \frac{1}{N} \sum_{i=1}^N \mathbf{x}(n) \mathbf{x}^H(n)$
 - 2 对 \mathbf{R}_{xx} 进行特征分解: $\mathbf{R}_{xx} = \mathbf{U}_S \mathbf{\Sigma}_S \mathbf{U}_S^H + \mathbf{U}_N \mathbf{\Sigma}_N \mathbf{U}_N^H$, 确定噪声子空间 $\hat{\mathbf{U}}_N$
 - 3 根据式 2.28: $P_{\text{MUSIC}}(\theta) = \frac{1}{\mathbf{a}^H(\theta) \hat{\mathbf{U}}_N \hat{\mathbf{U}}_N^H \mathbf{a}(\theta)}$, $\theta \in \left[-\frac{\pi}{2}, \frac{\pi}{2}\right]$ 进行搜索
 - 4 前 K 个极大值 $\hat{\theta}_k$ 点就是 DOA 估计的结果
-

Algorithm 1: 伪距单点定位

Input: 卫星观测数据和星历文件

Output: 接收段位置坐标

1: 初始化接收机位置 (x_0, y_0, z_0) 、接收机钟差 dT

2: **while** 计算误差 > 阈值 **do**

3: 根据星历信息与本地接收机时钟计算信号传输间隔 t

4: **for** 可视卫星 PRN_i **do**

5: 计算伪距 $\tilde{\rho}_k^i = \rho_k^i + \delta I_k^i + \delta T_k^i + c \cdot \delta t_k - c \cdot \delta t^i$

6: **end for**

7: 构造 L 矩阵、系数矩阵 \mathbf{B}

8: 求解定位结果 $\mathbf{X} = \begin{pmatrix} \delta x_k \\ \delta y_k \\ \delta z_k \\ C \cdot \delta t_k \end{pmatrix} = (\mathbf{B}^T \mathbf{B})^{-1} \mathbf{B}^T \mathbf{L}$

9: 更新接收端位置 $X_{k+1} = X_k + X$

10: **end while**

2.5 ESPRIT 算法原理

假设在 x 轴上存在两个结构和接收特性完全相同的子阵, 阵元数皆为 $m(m = L - 1)$; 两个子阵之间的间距为 D 。由于结构完全相同, 两个子阵对对于同一个信号源的响应系数只有一个相位差 ϕ_k 。设 $\mathbf{x}_1(t)$ 为第一个子阵接收的信号, $\mathbf{x}_2(t)$ 为第二个子阵接收的信号。根据式2.2给出的信号模型:

$$\mathbf{x}_1(t) = \begin{bmatrix} \mathbf{a}(\theta_1) & \mathbf{a}(\theta_2) & \cdots & \mathbf{a}(\theta_K) \end{bmatrix} \mathbf{s}(t) + \mathbf{n}(t) = \mathbf{A} \mathbf{s}(t) + \mathbf{n}(t) \quad (2.29)$$

$$\mathbf{x}_2(t) = \begin{bmatrix} \mathbf{a}(\theta_1) e^{j\phi_1} & \mathbf{a}(\theta_2) e^{j\phi_2} & \cdots & \mathbf{a}(\theta_K) e^{j\phi_K} \end{bmatrix} \mathbf{s}(t) + \mathbf{n}(t) = \mathbf{A} \Phi \mathbf{s}(t) + \mathbf{n}(t) \quad (2.30)$$

旋转矩阵 $\Phi \in \mathbb{C}^{K \times K}$ 表征了子阵 1 和子阵 2 的旋转不变关系:

$$\Phi = \text{diag}(e^{j\phi_1}, e^{j\phi_2}, \dots, e^{j\phi_K}) \quad (2.31)$$

$$\phi_k = \frac{2\pi D \sin \theta_k}{\lambda} \quad (2.32)$$

式2.32的 θ_k 就是需要进行估计的 DOA 参数, 由此可知, 只需要得到相位差 ϕ_k ,

就能够反过来对 θ_k 进行 DOA 估计。因此只需要得到两个子阵间的旋转不变关系 Φ , 就可以求解信号源的 DOA 参数。将式2.29和式2.30分别表示的阵列接收信号矢量拼接得 $2m \times K$ 维矩阵:

$$\mathbf{x}_t = \begin{bmatrix} \mathbf{x}_1(t) \\ \mathbf{x}_2(t) \end{bmatrix} = \begin{bmatrix} \mathbf{A} \\ \mathbf{A}\Phi \end{bmatrix} \mathbf{s}(t) + \mathbf{n}(t) = \overline{\mathbf{A}}\mathbf{s}(t) + \mathbf{n}(t) \quad (2.33)$$

同样对上式求 $2m \times 2m$ 维协方差矩阵:

$$\mathbf{R}_{xx} = E[\mathbf{x}(t)\mathbf{x}^H(t)] = \overline{\mathbf{A}}\mathbf{S}\overline{\mathbf{A}}^H + \mathbf{R}_N \quad (2.34)$$

同样对式2.34进行特征分解:

$$\mathbf{R}_{xx} = \mathbf{U}_S \mathbf{\Sigma}_S \mathbf{U}_S^H + \mathbf{U}_N \mathbf{\Sigma}_N \mathbf{U}_N^H \quad (2.35)$$

上式得到的特征值由如下关系:

$$\lambda_1 \geq \lambda_2 \geq \dots \geq \lambda_K > \lambda_{K+1} = \lambda_{K+2} = \dots = \lambda_{2m} = \sigma^2 \quad (2.36)$$

特征矢量阵 $\mathbf{U}_S \in \mathbb{C}^{2m \times K}$ 张成的 $\text{span}(\mathbf{U}_S)$ 是与这 K 个大特征值对应的信号子空间; 特征矢量阵 $\mathbf{U}_N \in \mathbb{C}^{2m \times (2m-K)}$ 张成的 $\text{span}(\mathbf{U}_N)$ 是与这 $2m - K$ 个小特征值对应的噪声子空间。式 (2.26) 的特征分解中信号特征矢量 \mathbf{U}_S 张成的信号子空间与阵列流型 $\overline{\mathbf{A}}$ 张成的子空间相同, 即:

$$\text{span}(\mathbf{U}_S) = \text{span}(\overline{\mathbf{A}}) \quad (2.37)$$

因此, 存在一个惟一的非奇异矩阵 $\mathbf{T} \in \mathbb{C}^{L \times K}$, 满足:

$$\begin{aligned} \mathbf{U}_S &= \overline{\mathbf{A}}\mathbf{T} \\ \mathbf{U}_S &= \begin{bmatrix} \mathbf{U}_{S1} \\ \mathbf{U}_{S2} \end{bmatrix} = \begin{bmatrix} \mathbf{A}\mathbf{T} \\ \mathbf{A}\Phi\mathbf{T} \end{bmatrix} \end{aligned} \quad (2.38)$$

根据式2.38 可知两个子阵 $\mathbf{U}_{S1}, \mathbf{U}_{S2}$ 的信号子空间满足:

$$\mathbf{U}_{S2} = \mathbf{U}_{S1}\mathbf{T}^{-1}\Phi\mathbf{T} = \mathbf{U}_{S1}\Psi \quad (2.39)$$

这个式子表征了两个子阵的阵列接收数据的信号子空间的旋转不变性。因此, $m \times K$ 维信号子空间阵 $\mathbf{U}_{S1}, \mathbf{U}_{S2}$ 与 $m \times K$ 维阵列流型矩阵 \mathbf{A} 张成的子空间相同:

$$\text{span}(\mathbf{U}_{S1}) = \text{span}(\mathbf{A}) = \text{span}(\mathbf{U}_{S2}) \quad (2.40)$$

在本文的理论阐述与实验设计中, 设定信号源数 K 小于子阵数目 $m(m = M - 1)$, 因此 $m \times K$ 维阵列流型 \mathbf{A} 是满秩的, 其秩为 K ; 则 \mathbf{U}_{S1} 同样满秩, 秩也为 K , 由式 (2.31) 可得:

$$\Phi = \mathbf{T}\Psi\mathbf{T}^{-1} \quad (2.41)$$

已知 \mathbf{T} 是非奇异的, 因此, m 维方阵 Φ 与 Ψ 是相似矩阵, 它们有着相同的特征值, Ψ 就是 Φ 的特征值对角阵。因此只要能够解得旋转不变关系矩阵 Ψ , 就可以通过式 (2.23) 得到信号源的 DOA 估计。下面使用总体最小二乘法解方程 2.39。考虑:

$$(\mathbf{U}_{S1} + \Delta\mathbf{U}_{S1})\Psi = \mathbf{U}_{S2} + \Delta\mathbf{U}_{S2} \quad (2.42)$$

可转写为:

$$\left(\begin{bmatrix} -\Delta\mathbf{U}_{S2} & \Delta\mathbf{U}_{S1} \end{bmatrix} + \begin{bmatrix} -\Delta\mathbf{U}_{S2} & \Delta\mathbf{U}_{S2} \end{bmatrix} \right) \begin{bmatrix} 1 \\ \Psi \end{bmatrix} = (\Delta\mathbf{U} + \mathbf{U})z = 0 \quad (2.43)$$

将两个子阵的信号特征矢量横向拼接, 得到:

$$\mathbf{U}_{S12} = [\mathbf{U}_{S1} \mid \mathbf{U}_{S2}] \quad (2.44)$$

计算 $\mathbf{U}_{S12}^H \mathbf{U}_{S12}$ 的矩阵分解:

$$\mathbf{U}_{S12}^H \mathbf{U}_{S12} = \mathbf{E}\hat{\Sigma}\mathbf{E}^H \quad (2.45)$$

其中, $\hat{\Sigma} \in \mathbb{C}^{2K \times 2K}$ 是由特征值构成的对角矩阵, \mathbf{E} 表征其对应的特征向量。令:

$$\mathbf{E} = \begin{bmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{bmatrix} \quad (2.46)$$

其中

$$\mathbf{E}_N = \begin{bmatrix} \mathbf{E}_{12} \\ \mathbf{E}_{22} \end{bmatrix} \quad (2.47)$$

是由对应特征值为 0 的特征矢量构成的矩阵, 它属于噪声子空间。令:

$$\mathbf{F} = \begin{bmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \end{bmatrix} \triangleq \begin{bmatrix} \mathbf{E}_{12} \\ \mathbf{E}_{22} \end{bmatrix} \quad (2.48)$$

则有:

$$\mathbf{U}_{12}\mathbf{F} = [\mathbf{U}_{S1} \mid \mathbf{U}_{S2}] \begin{bmatrix} \mathbf{F}_1 \\ \mathbf{F}_2 \end{bmatrix} = \mathbf{U}_{S1}\mathbf{F}_1 + \mathbf{U}_{S2}\mathbf{F}_2 = 0 \quad (2.49)$$

由式2.38, 可得:

$$\mathbf{A}\mathbf{T}\mathbf{F}_1 + \mathbf{A}\Phi\mathbf{T}\mathbf{F}_2 = 0 \quad (2.50)$$

则有

$$\Psi = -\mathbf{F}_1\mathbf{F}_2^{-1} \quad (2.51)$$

$$\mathbf{T}^{-1}\Phi\mathbf{T} = \Psi \quad (2.52)$$

因此, ESPRIT-DOA 估计的解为:

$$\Phi_{TLS} = -\mathbf{E}_{12}\mathbf{E}_{22}^{-1} \quad (2.53)$$

算法 2: ESPRIT-DOA 估计算法

输入: 快拍数据矢量 $\mathbf{x}(n) \in \mathbb{C}^{L \times N}$ 和信号源数 K

1 由阵列接收数据得到数据协方差矩阵 $\hat{\mathbf{R}}_{xx} = \frac{1}{N} \sum_{n=1}^N \mathbf{x}(n)\mathbf{x}^H(n)$

2 对 $\hat{\mathbf{R}}_{xx}$ 进行特征分解: $\mathbf{R}_{xx} = \mathbf{U}_S \Sigma_S \mathbf{U}_S^H + \mathbf{U}_N \Sigma_N \mathbf{U}_N^H$, 确定信号子空间 $\hat{\mathbf{U}}_S$

3 构造矩阵 $\mathbf{U}_{S12} = [\mathbf{U}_{S1} \mid \mathbf{U}_{S2}]$, 计算 $\mathbf{U}_{S12}^H \mathbf{U}_{S12}$ 特征值分解 $\mathbf{U}_{S12}^H \mathbf{U}_{S12} = \mathbf{E} \hat{\Sigma} \mathbf{E}^H$

4 \mathbf{E} 分块为 $\mathbf{E} = \begin{bmatrix} \mathbf{E}_{11} & \mathbf{E}_{12} \\ \mathbf{E}_{21} & \mathbf{E}_{22} \end{bmatrix}$

5 定义 $\Phi_{TLS} = -\mathbf{E}_{12}\mathbf{E}_{22}^{-1}$, 进而得到 ϕ 和 θ

2.6 OMP 算法原理

2.6.1 贪婪算法

贪婪 (greedy) 算法为一类近似 L_0 范数优化的算法, 通过某种准则, 逐步地取原子, 然后根据选取的原子确定非零元素的系数。贪婪算法基本思想: 不求整体最优解, 而是试图尽快找到某种意义上的局部最优解。贪婪算法虽然不能够对所有问题得到整体最优解, 但对范围相当广泛的诸多问题均能得到整体最优或近

似整体最优解。

2.6.2 OMP 算法步骤

匹配追踪 (MP) 算法的提出使贪婪算法成为了压缩感知稀疏重构算法中重要的组成部分, 是贪婪算法发展的里程碑。OMP 算法是对 MP 算法的进一步改进, 它将选择的全部原子进行正交化处理, 对加快算法的收敛起到了一定的作用。其核心思想为: 初始化一个空索引集, 将残差信号与过完备原子库中的原子依次进行内积相关性运算, 选择最为相关的原子对应的索引值放入索引集中, 并进行残差更新, 反复迭代直到到达终止条件为止。OMP 算法流程可概括如下:

算法 3: OMP 算法

输入: 感知矩阵 Φ , 观测向量 y , 稀疏度 K 。

- 1 初始化残差 $r_0 = y$, 支撑集 $\Lambda_0 = \phi$, 迭代计数 $k = 1$ 。
- 2 令残差与观测矩阵做内积运算, 找到索引 λ_k , 使其满足 $\lambda_k = \arg \max |\langle r_{k-1}, \varphi_\lambda \rangle|$;
- 3 更新索引集 $\Lambda_k = \Lambda_{k-1} \cup \{\lambda_k\}$, 更新观测矩阵列向量集合 $\Phi_k = \Phi_{k-1} \cup \{\Phi_{\lambda_k}\}$;
- 4 根据最小二乘法得到重构数据 $\hat{x} = (\Phi_k^T \Phi_k)^{-1} \Phi_k^T y$; $U_{S12}^H U_{S12} = E \hat{\Sigma} E^H$
- 5 更新残差 $r_k = y - \Phi_k \hat{x}$, 更新迭代次数 $k = k + 1$;
- 6 判断是否满足迭代停止条件, 满足则迭代停止, 不满足则回到步骤 1 继续执行。
- 7 算法输出: 原始信号 x 的迭代估计值 \hat{x} 。

2.7 SBL 算法原理

2.7.1 稀疏优化与贝叶斯

为了使讨论的问题更具一般意义, 这里给出稀疏模型的一般形式:

$$\mathbf{t} = \Phi \mathbf{w} + \varepsilon \quad (2.54)$$

上式中的 \mathbf{t} 表示观测数据, $\Phi \in \mathbb{C}^{M \times N}$ 是 M 个基向量组成的字典, ε 通常为高斯噪声, 此模型为单次观测模型。该模型对应点源稀疏模型, 则 \mathbf{t} 为阵列输出 $\mathbf{y}(t)$, Φ 是阵列流形矩阵 $\mathbf{A}(\Phi)$, ε 为噪声分量 $\mathbf{n}(t)$ 。为了便于下文讨论, 定义

$$d(\mathbf{w}) = \|\mathbf{w}\|_0 = \sum_{i=1}^N \mathcal{I}[\|\mathbf{w}_i\| > 0] \quad (2.55)$$

$\mathcal{I}[\cdot]$ 表示符号函数, $d(\mathbf{w})$ (diversity) 为向量 \mathbf{w} 中非零项的个数。稀疏度 (sparsity)

表示向量中的零元素个数。因此,二者具有如下的关系: $\text{diversity} = N - \text{sparsity}$ 。 Φ 表示字典矩阵, $\text{spark}(\Phi)$ 表示字典矩阵中线性独立的最小列数, 根据定义, Φ 的 spark 满足 $2 \leq \text{spark}(\Phi) \leq M + 1$ ^[61]。如果 $\text{spark}(\Phi) = M + 1$, 则表示 Φ 中的每 M 个列都线性独立, 信号表示具有唯一性; 如果 $\text{rank}(\Phi) = M$ 和 $M < N$, 则 Φ 称为过完备字典 (一般情况下, $M \ll N$ 。回到稀疏恢复问题, 首先考虑最直接的情况, 即2.54中的 $\varepsilon = 0$, 如果 Φ 为过完备字典, 则稀疏恢复是一个欠定问题 (病态求逆问题), 除非有额外的条件限制。例如, 如果产生 \mathbf{w}_{gen} 的矩阵 Φ 满足如下关系:

$$\|\mathbf{w}_{\text{gen}}\|_0 < \text{spark}(\Phi) - 1 \quad (2.56)$$

则不存在其他的 \mathbf{w} 使 $\mathbf{t} = \Phi\mathbf{w}$ 和 $\|\mathbf{w}\|_0 \leq \|\mathbf{w}_{\text{gen}}\|_0$ 得到满足。则恢复 \mathbf{w}_{gen} 等价于解决如下优化问题:

$$\mathbf{w}_{\text{gen}} = \mathbf{w}_0 = \arg \min_w \|\mathbf{w}\|_0, \quad \text{s.t. } \mathbf{t} = \Phi\mathbf{w} \quad (2.57)$$

这有时被称为精确稀疏回复问题, 因为其解使等式 $\mathbf{t} = \Phi\mathbf{w}$ 严格成立。一般情况下, 求解2.57是一个 NP-hard 问题, 因此必须近似求解。在本章的下文部分也将详细讨论2.57的解。在本章中, 只要 $\varepsilon = 0$, 就假设 \mathbf{w}_{gen} 满足2.56, 另外, 最大稀疏解 \mathbf{w}_0 和 \mathbf{w}_{gen} 可相互替换。当 $\varepsilon \neq 0$ 时, 问题变得更加麻烦。因为噪声的存在, 通常不预期能够精确表示 \mathbf{t} , 而是采用松弛优化问题2.58来获得 \mathbf{w}_0 的近似解

$$\mathbf{w}_0(\lambda) = \arg \min_w \|\mathbf{t} - \Phi\mathbf{w}\|_2^2 + \lambda \|\mathbf{w}\|_0 \quad (2.58)$$

上式中参数 λ 称为正则化参数, 用于平衡拟合误差和稀疏度。遗憾地是求解2.58也是一个 NP-hard 问题, 也不知道如何选择正则化参数 λ 。另外, 即使有优化的正则化参数 λ 可用, 也不能保证全局最优解是 \mathbf{w}_{gen} 的最佳估计。从概念上来说, 2.58可以通过添加常量和使用指数变换 $\exp[-(\cdot)]$, 将其改写成贝叶斯形式。假定方差为 λ 的高斯似然函数 $p(\mathbf{t} | \mathbf{w})$ 为:

$$p_1(\mathbf{w}) \propto \exp \left(- \sum_{i=1}^N |w_i| \right) \quad (2.59)$$

$$p_1(\mathbf{w}) \propto \exp \left(- \sum_{i=1}^N |w_i| \right) \quad (2.60)$$

该分布常被用于 Basis Pursuit (BP) 等方法, 利用该先验的 MAP 估计, 即解决如

下优化问题:

$$\mathbf{w}_{\text{BP}} = \underset{\mathbf{w}}{\text{arcmin}} \|\mathbf{t} - \Phi \mathbf{w}\|_2^2 + \lambda \sum_{i=1}^N |w_i| \quad (2.61)$$

该凸代价函数可以被各种标准的优化包很容易地进行全局最小化。虽然该松他通常比较有效,但是在实践中, BP 的解并不是足够稀疏。为解决拉普拉斯先验存在的问题,第二个常用的先验是 scale-invariant Jeffreys 先验:

$$p_J(\mathbf{w}) \propto \prod_i^N \frac{1}{|w_i|} \quad (2.62)$$

该先验具有严重的长尾现象,在零值处具有尖峰,其反应了稀疏分布的特点。利用该先验的 MAP 估计,即解决如下优化问题:

$$\mathbf{w}_{\text{Jeffreys}} = \underset{\mathbf{w}}{\text{arcmin}} \|\mathbf{t} - \Phi \mathbf{w}\|_2^2 + \lambda \sum_{i=1}^M \log |w_i| \quad (2.63)$$

这里的正则化有时又被称为高斯熵。基于 Jeffreys 代价函数的优化存在很多局部最小值的问题,但是如果初始化足够的好,它可能找到比 BP 解更优的解。考虑一般的高斯先验:

$$p(\mathbf{w}) \propto \exp \left(- \sum_{i=1}^N |w_i|^p \right) \quad (2.64)$$

上式中的 $p \in [0, 1]$, 是使用者定义参数。该先验对应的优化问题,有时被称为 FOCUSS 方法,即解决如下优化问题:

$$\mathbf{w}_{\text{Focuss}} = \underset{\mathbf{w}}{\text{arcmin}} \|\mathbf{t} - \Phi \mathbf{w}\|_2^2 + \lambda \sum_{i=1}^N |w_i|^p \quad (2.65)$$

如果 $p \rightarrow 0$, 则看似接近于最优解,但是在这种情况下,对于任何参数 λ , 方法的更新规则只能使方法收敛于局部最小值。当 $p = 1$ 时, FOCUSS 方法变为实现 BP 方法的内点法。Jeffreys 方法也是 FOCUSS 框架的一个特殊形式。在实际应用中,有时通过联合选取参数 p, λ 的值,从而是方法优于

2.8 CNN-DOA 算法原理

2.8.1 BP 神经网络

BP(Back-Propagation) 神经网络的出现再次引发人们对神经网络的关注,它属于前馈神经网络,所以包含输入层、隐藏层和输出层,各层之间全部互相连接,但

单层之间的神经元不会相互连通。其拓扑图如图2-2所示。假设在输入层输入的矢量为 x , 且 $x \in R^n$, 其中 $x = (x_1, x_2, \dots, x_n)^T$ 。 x 会被送至隐含层的每一个神经元。

假设在隐藏层中有 n_1 个神经元, 则这些神经元的输出 $x' = (x'_1, x'_2, \dots, x'_n)^T$ 。 x' 也会被送至输出层的每一个神经元。其中隐藏层的输出为

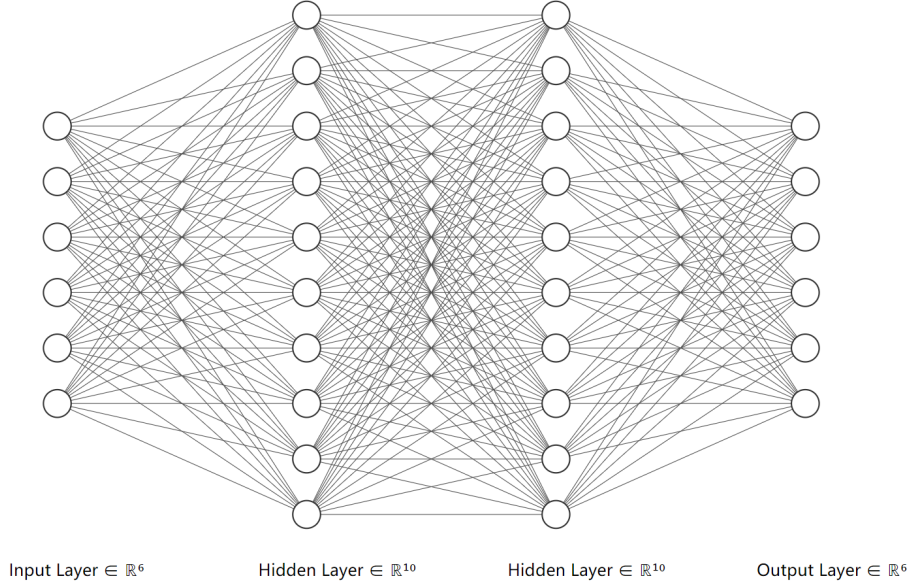


图 2-2 BP 神经网络模型

$$\mathbf{Z} = f(\text{net}_k) = f\left(\sum_{i=1}^n (w_{ji} \cdot x_i + b_j)\right) \quad (2.66)$$

式中 $w_{ij}(i = 1, 2, \dots, n, j = 1, 2, \dots, l)$ 为第 i 个隐藏层或第 j 个隐含层神经元与第 i 个输入层神经元之间的权重, b_{nj} 为第 j 个隐含层神经元的偏置, 经过激活函数后, 隐含层输出数据为 $\mathbf{Z} \in \mathbb{C}^{n \times 1}$

相同道理, 输出层的输出为

$$\mathbf{Y} = f(\text{net}_j) = f\left(\sum_{j=1}^l (w_{kj} \cdot z_j + b_k)\right) \quad (2.67)$$

式中, $w_{kj}(j = 1, 2, \dots, m, k = 1, 2, \dots, l)$ 为第 k 个输出层神经元与前一隐含层第 j 个神经元之间的权重, b_k 为第 k 个输出层神经元的阈值, 经过激活函数后, 最终输出层数据为 $\mathbf{Y} \in \mathbb{C}^{m \times 1}$ 。

以上所述即为 BP 神经网络输入数据前向传播的过程, 如果仅有前向传播, 将无法修改前面的参数, 因此引入误差后向传播。假设有 n 个样本数, 输入训练集数

据 x , 经过上述 BP 神经网络后得到的输出为 \mathbf{Y} , 计算均方误差得:

$$E^n = \frac{1}{2} \sum_{n=1}^N \sum_{k=1}^m (t_k - y_k)^2 \quad (2.68)$$

式中 t_k 为 BP 神经网络预测值, y_k 为样本输出的真实值。

BP 神经网络可以通过收集系统所产生的误差并调整神经元的权重, 从而达到对非线性输入输出关系的模拟。调整的算法一般由随机梯度下降法和最小二乘法。代价函数的介绍将放置于本小节的第三部分。BP 神经网络具有性能好, 结构简单和操作性强等优点。但它容易陷入局部最小值, 并且收敛速度慢。所以本文的深度学习结构采用了卷积神经网络, 其具体介绍如下。

2.8.2 卷积神经网络

卷积神经网络同样属于前馈网络中的一种, 常应用于图像分类、识别等领域。相较于 BP 神经网络, 卷积神经网络所需训练的参数较少。卷积神经网络包含卷积层、池化层和全连接层三个部分。接下来将按照这三者具体介绍。

卷积层 (convolutional layer): 卷积层中, 需要设计卷积核 (kernel) 提取数据的局部特征。提取的结果相当于信号通过滤波器的输出, 其输出为特征映射图。用卷积核在输入矩阵上按照特定的步幅滑动, 卷积核的值与对应的输入输入数据相乘, 最后得到的特征映射图中所有元素都由卷积核共享权重和偏置而来。

池化层 (pooling layer): 一种非线性采样, 其目的是降低运算量, 降低数据的尺寸。一般包含最大池化 (Max Pooling) 和平均池化 (Average Pooling)。与卷积类似, 池化层也需要用一个窗口在数据矩阵上滑行, 得到采样数据。最大池化、平均池化即去窗口中的最大值与平均值作为采样数据。

全连接层 (fully connected layer): 全连接层即都上一层的所有结点与该层每一个结点相连, 用于综合提取到的特征。

2.8.3 卷积神经网络的设计

完整的前馈神经网络模型设计应包括网络模型的设定、激活函数和代价函数的选定、训练策略设计以及测试模型性能。我们想要设计一个符合要求的卷积神经网络模型, 应该确定输入层输出层的形式和隐藏层的数量和结构。

2.8.3.1 设计原理

首先给出函数挤压性与万能近似定理^[2]的定义, 若有一个函数 Γ 将所有实数域映射到 $[0,1]$ 中, 且该函数为非递减的, 并满足 $\lim_{\lambda \rightarrow \infty} \Gamma(\lambda) = 1$, $\lim_{\lambda \rightarrow -\infty} \Gamma(\lambda) = 0$, 则称该函数具有挤压性。而万能近似定理是指, 若一个网络存在具有挤压性的激活函数隐藏层和线性输出层, 并且网络具有足够数量的隐藏单元, 那么这个网络可以逼近有限空间映射到有限空间波莱尔的可测函数。

由以上两个定义可知, 任何满足定理条件的映射都可以利用深度前馈神经网络逼近其功能。但对于具体搭建的网络, 若是太简单, 学习的深度和层数不足, 将导致学习特征不足, 导致网络预测效果不佳, 从而造成欠拟合。然而, 一味地增加网络的层数和深度亦不可取, 将导致噪声的过度影响, 从而造成过拟合。同时, 隐藏层的设计也至关重要, 隐藏层的层数与神经元的个数也会影响网络的效果。

2.8.3.2 激活函数

引入激活函数的目的是增强非线性输入输出关系的模拟性能。激活函数的选取对模型的影响很大, 决定输入集经过节点后的输出。激活函数选取的原则主要是使梯度下降算法的各层输出可微分。其次, 选取的激活函数需解决梯度消失和梯度爆炸。

Swish 函数

为解决神经元坏死问题, Google 大脑团队提出了 Swish 激活函数, 其大致形状与 ReLU 函数差不多。其具体定义为

$$swish(x) = x \cdot Sigmoid(\beta x) \quad (2.69)$$

其中 Sigmoid 函数即是式(2.68), 其函数形状如图2-3, 可见其形状与 ReLU 函数类似, 但在负半轴上其取值不为 0, 因此不会出现神经元梯度为 0 的情况。而且 Swish 处处可导, 连续光滑, 在大多数情况下其性能优于 ReLU 函数。但其缺点就是计算量大, 收敛较慢。

2.8.3.3 代价函数

代价函数主要是用来衡量模型计算的准确性, 均方误差代价函数即将模型计算的输出与标签值之间的均方误差作为输入样本的代价函数。所用 \hat{y} 表示输出值,

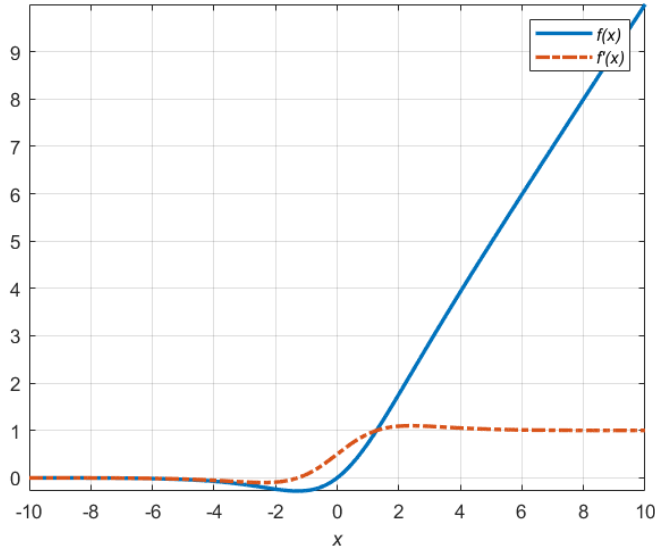


图 2-3 Swish 示意图

用 \mathbf{y} 表示标签值，则两者的均方误差可表示为：

$$\text{MSE}(\{\hat{\mathbf{y}}\}_m, \{\mathbf{y}\}_m) = \frac{1}{m} \sum_{i=1}^m \|\hat{\mathbf{y}}_i - \mathbf{y}_i\|_2 \quad (2.70)$$

2.8.3.4 Adam 优化函数

Adam 算法结合了 AdaGrad (Adaptive Gradient) 和 RMSProp (Root Mean Square Propagation) 算法优点。该算法中的参数根据元素运算使得每个元素都分别有自己的学习率，而且优化损失函数在更新中摆幅过大的问题，并加快了收敛速度。是一个综合性能优越的优化算法。其具体的算法流程如表2-1所示。

2.8.3.5 CNN 模型设计思路

A、角度量化

为方便后续样本的生成和计算，在确保不妨碍欺骗干扰检测的前提下，将一维均匀阵列天线接收信号到达角进行量化^[7]。根据欺骗干扰原理可知，两个卫星信号的 DOA 一般都相差较大。因此，可以将 DOA 区间量化。

根据2.1小节所提到的一维均匀阵列天线信号接收模型, 真实信号来向为:

$$\Theta = \{\theta_1, \theta_2, \dots, \theta_K\} \quad (2.71)$$

表 2-1 Adam 算法

算法 4: Adam 算法
输出参数: 初始学习率 l 、网络参数 θ 、数值稳定常数 δ 一阶和二阶矩变量 $s = 0, r = 0$ 以及时间 $t = 0$ while 终止条件未满足 do 从训练集中采包含 m 个样本 $\{\mathbf{x}^{(1)}, \dots, \mathbf{x}^{(m)}\}$ 的小批量, 其中 $\mathbf{x}^{(i)}$ 对应目标标签为 $y^{(i)}$; 计算梯度估计: $\hat{\mathbf{g}} \leftarrow \frac{1}{m} \nabla_{\theta} \sum_i L(f(\mathbf{x}^{(i)}; \theta), y^{(i)})$; 累计平方梯度: $\mathbf{r} \leftarrow \rho \mathbf{r} + (1 - \rho) \mathbf{g} \odot \mathbf{g}$; 更新时间: $t \leftarrow t + 1$; 更新有偏一阶矩估计: $s \leftarrow \rho_1 s + (1 - \rho_1) \hat{\mathbf{g}}$; 更新有偏二阶矩估计: $r \leftarrow \rho_2 r + (1 - \rho_2) \mathbf{r} \odot \mathbf{r}$; 修正一阶矩的偏差: $\hat{s} \leftarrow \frac{s}{1 - \rho_1^t}$; 修正二阶矩的偏差: $\hat{r} \leftarrow \frac{r}{1 - \rho_2^t}$; 计算更新: $\Delta \theta = -l \frac{\hat{s}}{\sqrt{\hat{r} + \delta}}$ (逐元素地应用除和求平方根) 应用更新: $\theta \leftarrow \theta + \Delta \theta$; end while

将量化后的真实信号来向代入2.3中, 可得:

$$\bar{\mathbf{x}}(t) = \sum_{p=1}^P \alpha(\phi_p) \bar{\mathbf{s}}_p(t) + \mathbf{n}(t) \quad (2.72)$$

其矩阵线性变换形式为:

$$\begin{bmatrix} \bar{x}_1(t) \\ \bar{x}_2(t) \\ \vdots \\ \bar{x}_L(t) \end{bmatrix} = \begin{bmatrix} e^{-j\frac{2\pi}{\lambda}(1-1)d \sin \theta_1} & e^{-j\frac{2\pi}{\lambda}(1-1)d \sin \theta_2} & \dots & e^{-j\frac{2\pi}{\lambda}(1-1)d \sin \theta_K} \\ e^{-j\frac{2\pi}{\lambda}(2-1)d \sin \theta_1} & e^{-j\frac{2\pi}{\lambda}(2-1)d \sin \theta_2} & \dots & e^{-j\frac{2\pi}{\lambda}(2-1)d \sin \theta_K} \\ \vdots & \vdots & \ddots & \vdots \\ e^{-j\frac{2\pi}{\lambda}(L-1)d \sin \theta_1} & e^{-j\frac{2\pi}{\lambda}(L-1)d \sin \theta_2} & \dots & e^{-j\frac{2\pi}{\lambda}(L-1)d \sin \theta_K} \end{bmatrix} \begin{bmatrix} \bar{s}_1(t) \\ \bar{s}_2(t) \\ \vdots \\ \bar{s}(t) \end{bmatrix} + \begin{bmatrix} n_1(t) \\ n_2(t) \\ \vdots \\ n_L(t) \end{bmatrix} \quad (2.73)$$

其矢量形式为:

$$\bar{\mathbf{x}}(t) = \mathbf{A} \bar{\mathbf{s}}(t) + \mathbf{n}(t) \quad (2.74)$$

式中, $\mathbf{n}(t)$ 表示加性高斯白噪声矢量, $\bar{\mathbf{x}}(t)$ 为阵元接收快拍矢量; $\bar{\mathbf{s}}(t)$ 为信号源矢量, 其具体表达如下:

$$\bar{s}_p(t) = \begin{cases} s_k(t), & \phi_p = \theta_k \\ 0, & \phi_p \neq \theta_k \end{cases}, \quad p = 1, 2, \dots, P \quad (2.75)$$

\mathbf{A} 表示信号源 $\{\bar{s}_p(t)\}$ 对一维 L 元的均匀阵列天线流型矩阵, 其第 p 列 $L \times 1$ 维导向矢量表示 L 元均匀阵列天线对第 p 个信号的导向矢量:

$$\mathbf{a}(\phi_p) = \begin{bmatrix} e^{-j\frac{2\pi}{\lambda}(1-1)d \sin \phi_p} \\ e^{-j\frac{2\pi}{\lambda}(2-1)d \sin \phi_p} \\ \vdots \\ e^{-j\frac{2\pi}{\lambda}(L-1)d \sin \phi_p} \end{bmatrix}, \quad p = 1, 2, \dots, P \quad (2.76)$$

阵列天线接收快拍矢量 $\bar{\mathbf{x}}(t)$ 的协方差矩阵 $\bar{\mathbf{R}}_{xx} \in \mathbb{C}^{L \times L}$ 可表示为:

$$\begin{aligned} \bar{\mathbf{R}}_{xx} &= E [\bar{\mathbf{x}}(t)\bar{\mathbf{x}}^H(t)] + \sigma^2 \mathbf{I}_L \\ &= E [\mathbf{A}\bar{\mathbf{s}}(t)\bar{\mathbf{s}}^H(t)\mathbf{A}^H] + \sigma^2 \mathbf{I}_L \\ &= \mathbf{A} E [\bar{\mathbf{s}}(t)\bar{\mathbf{s}}^H(t)] \mathbf{A}^H + \sigma^2 \mathbf{I}_L \end{aligned} \quad (2.77)$$

式中, \mathbf{I}_L 为 $L \times L$ 的单位矩阵。再用最大似然估计即可完成对量化角度的估计。

B、网络设置

Swish 激活函数在大多数情况下都有较好的性能, 因此该神经网络选用的激活函数为 Swish。并且使用小批量学习, 即打乱数据集后随机抽取样本进行训练。具体的 CNN 网络设置如表2-2。该算法的具体流程如图2-4所示。

表 2-2 CNN 网络设置

网络结构	选用
激活函数	Swish 函数
代价函数	MSE(均方误差)
优化算法	Adam 算法

2.8.4 样本生成思路

运用深度学习来解决 DOA 估计问题, 首要解决的是生成合适的训练样本, 再进入搭建好的神经网络进行训练。首先将均匀阵列天线接收信号 DOA 估计区间量化, 产生两个随机角度的信号, 并添加一定的信噪比, 运用 MUSIC 算法得到的 DOA 估计谱^[7], 以此作为的特征值, 并用对应的两个角度作为标签值。即以入射角度和信噪比为变量生成深度学习的训练样本, 环境参数设置如表2-3所示, 所生成的样本特征与标签值如表2-4。

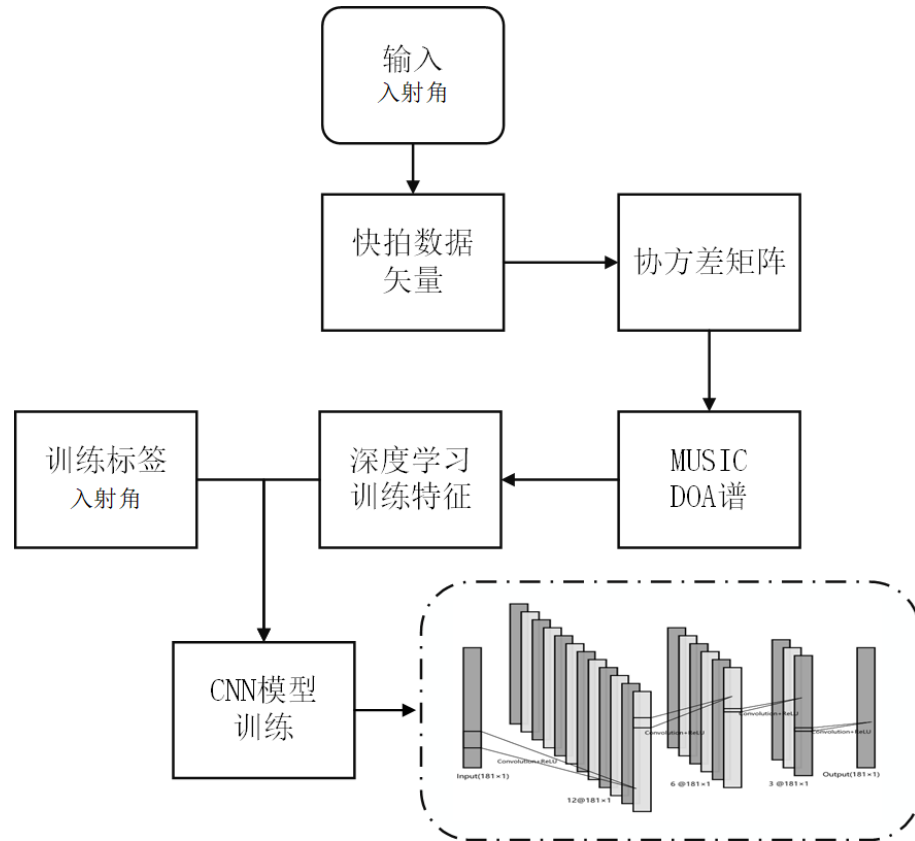


图 2-4 算法流程图

在大批量样本中随机生成两个样本加以观察，其特征和标签如图2-5。在图2-5a中特征与标签值对应很明显，但当低信噪比环境或者入射角度相差很小时，其特征值的峰起与标签值会有一定的偏差，甚至只将两个入射角检测出一个峰值。如图2-5b所示，当入射角为 24° 、 30° ，并且 SNR 为 -10dB 时，特征值与标签值不相对应。所以我们要利用 CNN 来检测这类低信噪比的 DOA 估计，从而增强防欺骗的性能。

表 2-3 环境参数设置

变量	符号	值（分布）
阵元数	L	8
信号数	K	2
快拍数	N	64
量化角度数	P	181
信噪比	SNR	$[-10, 30]\text{dB}$

表 2-4 训练集样本

名称	维度	意义
theta_train	$90300 \times 2 \times 1$	两个信号的入射角度 $\{\theta_1, \theta_2\}$
Signal_eta	$90300 \times 181 \times 1$	输入特征 $\hat{\eta}$
Signal_label	$90300 \times 181 \times 1$	输出标签 \mathbf{y}

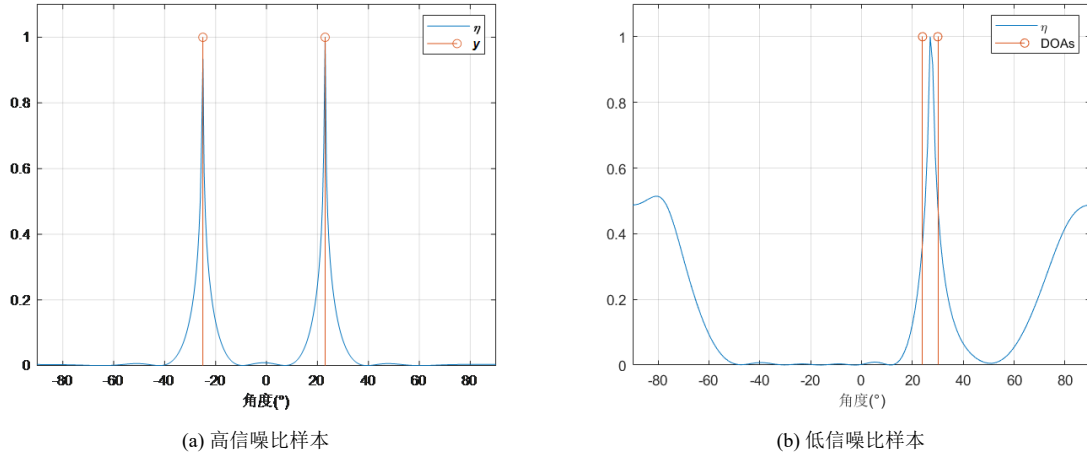


图 2-5 训练集样本

2.9 本章小结

本章首先解释信号波达角的含义并具体说明了基于阵列天线的 DOA 检测模型均匀阵列天线 (ULA)。并介绍了 CBF、Capon、MUSIC、ESPRIT、OMP、SBL 与 CNN 的 DOA 估计原理。简单介绍深度学习的原理, 根据一维均匀阵列天线信号接收模型, 搭建了 CNN 的 DOA 估计网络。

第3章 仿真设计

本章将通过 MATLAB 环境仿真分析 CBF、Capon、MUSIC、ESPRIT、OMP、SBL 的 DOA 估计性能随 SNR、阵元数、阵元间距波长比以及阵元误差的关系。并使用 MATLAB 和 Python 环境进行了 CNN 网络模型的仿真、训练、测试与性能分析。最后将 CNN-DOA 结果与经典 DOA 方法做对比。

3.1 阵元间距波长比对 DOA 检测性能影响

估计两个信号的 DOA，接收机参数及环境参数设置如表3-1，图3-1为波长对 DOA 检测性能的影响。当阵列天线各个阵列元的间距小于半波长 ($d \leq \frac{\lambda}{2}$)，DOA 的估计谱都只有两个峰值，并且峰值对应两个入射角 (在图3-1中表现为两条黄线)，当阵元间距波长比小于 0.5 时，增大阵列元间距相当于增大阵列孔径，从而导致分辨率的增大。但当阵列元间距大于半波长时，由图3-1右图的上部分可见，出现了虚假谱峰。这是因为此时相位所在的区间超过 2π ，存在多个能与导向矢量正交的相角，从而导致 DOA 估计失效，检测出现虚假谱峰。因此后续小节将取 $d = \frac{\lambda}{2}$ 。

表 3-1 变量参数

变量	符号	值
阵元间距波长比	$c = \frac{d}{\lambda}$	[0.1:0.1:2]
阵元数	L	8
信噪比	SNR	5dB
入射角	θ	$-20^\circ, 45^\circ$

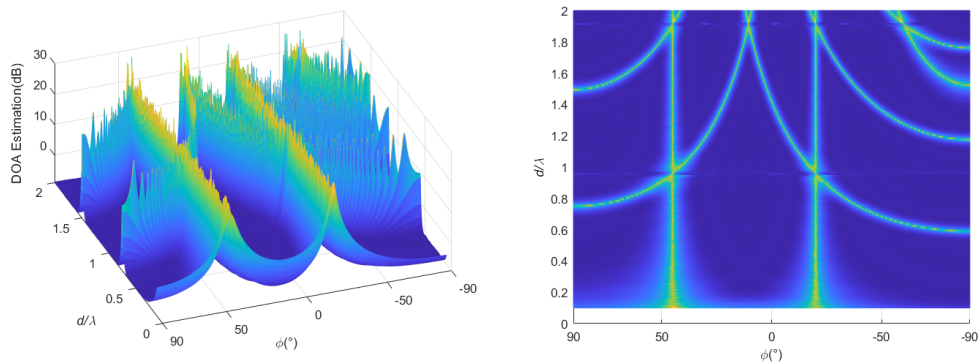


图 3-1 波长对 DOA 检测性能影响

3.2 DOA 估计谱比较

仿真实验分析 CBF、Capon、MUSIC、ESPRIT、OMP、SBL 与 CNN 的 DOA 估计谱，首先设置接收机参数及环境参数如表3-2。各 DOA 估计方法的 DOA 谱如3-2。在目标的角度间隔较大，信噪比和快拍数都比较大时，七种算法都能实现目标的分辨。但 CBF 的分辨率最低，Capon 此之，MUSIC 算法的分辨率较高，CNN 分辨率最高。ESPRIT、OMP、ESPRIT 计算的入射角度也比较接近真实值。

表 3-2 变量参数

变量	符号	值（分布）
阵元数	L	8
信号数	K	2
快拍数	N	128
入射角	θ	$-20^\circ, 45^\circ$
信噪比	SNR	$20dB$

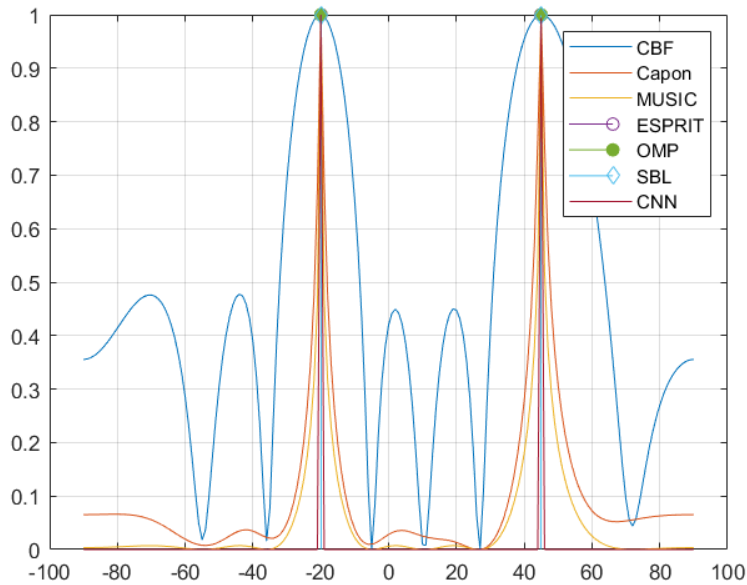


图 3-2 DOA 估计谱比较

将入射角设置为 -20° 、 -15° ，快拍数置为 24，CBF、Capon、MUSIC、SBL 与 CNN 的 DOA 估计谱如图3-3。CBF 只有一个谱峰，而 Capon 与 MUSIC 的两个峰值也不明显，而 CNN 依然保持高分辨率。ESPRIT、SBL 的 DOA 估计值也很接近真实值，OMP 的估计值略有偏差。

增加信号源个数，设置入射角角度为 -20° 、 -15° 、 -0° ，Capon、MUSIC、SBL 与 CNN 的 DOA 估计谱如图3-4。CNN、SBL 对三个角度都准确估计，而 Capon、

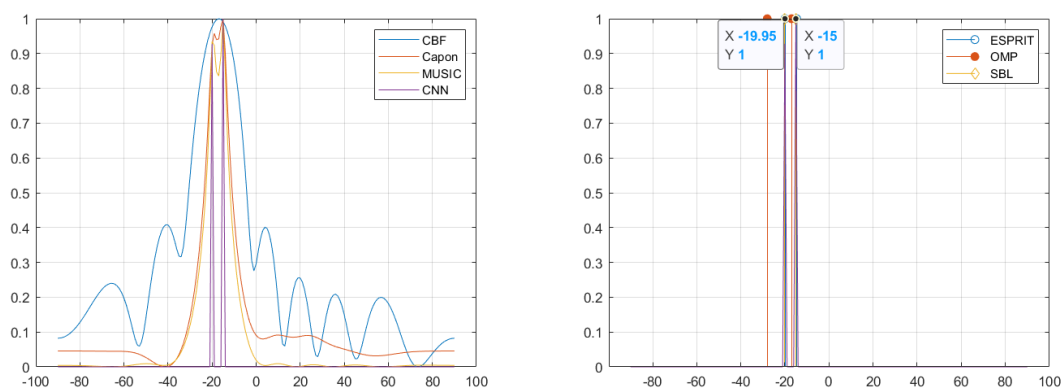


图 3-3 DOA 估计谱比较

MUSIC 算法在角度相近时分辨不出。

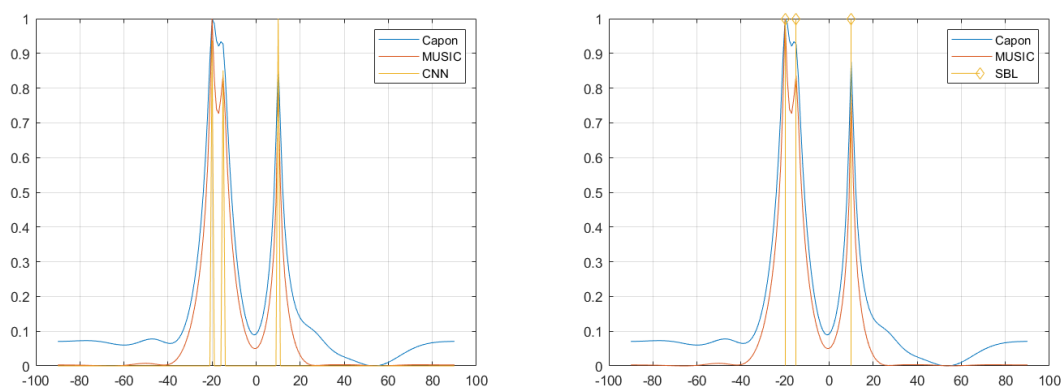


图 3-4 DOA 估计谱比较

3.3 信噪比对 DOA 估计性能影响

仿真实验分析信噪比对 DOA 估计性能的影响，首先设置接收机参数及环境参数如表3-3。随机分布入射角和信噪比，并以信噪比划分数据集。对双目标入射角进行 DOA 估计，进行谱峰搜索后比较得到 DOA 估计值，对 DOA 估计值与正确值作比较。得到 DOA 估计正确率与均方根误差 (RMSE) 随信噪比的关系如图3-5。随着信噪比的增大，DOA 估计正确率增大，均方根误差减小。其中 SBL 算法，CNN 算法的估计正确率较高，均方根误差较小。在信噪比较低时，ESPRIT 的估计准确率较低，但均方根误差相对较小。

表 3-3 变量参数

变量	符号	值
阵元间距波长比	$c = \frac{d}{\lambda}$	0.5
阵元数	L	8
快拍数	N	64
入射角	θ	$[-90^\circ:95^\circ]$
信噪比	SNR	$[-10:30]\text{dB}$

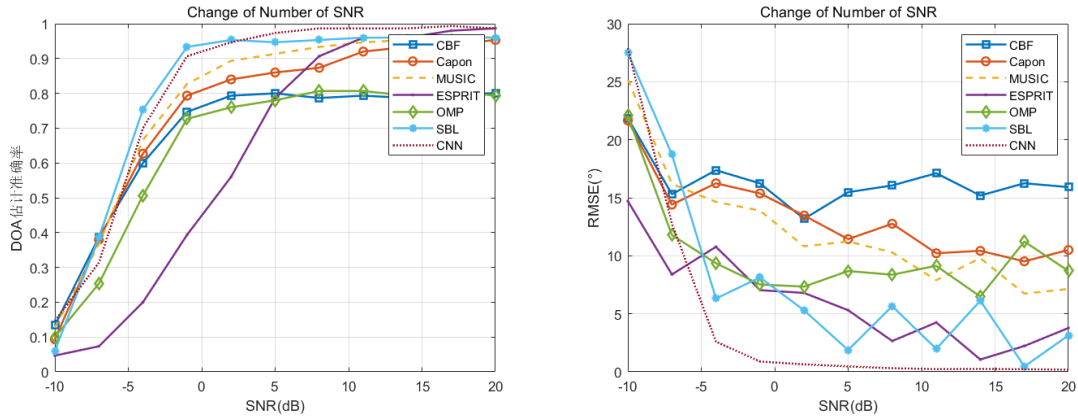


图 3-5 DOA 估计正确率与均方根误差随 SNR 的关系

3.4 快拍数对 DOA 估计性能影响

仿真实验分析快拍数对 DOA 估计性能的影响，首先设置接收机参数及环境参数如表3-4。随机分布入射角和快拍数，并以快拍数划分数据集。对双目标入射角进行 DOA 估计，进行谱峰搜索后比较得到 DOA 估计值，对 DOA 估计值与正确值作比较。得到 DOA 估计正确率与均方根误差 (RMSE) 随信噪比的关系如图3-6。可见随着快拍数的增大，DOA 估计正确率增大，均方根误差减小。其中 SBL 算法，CNN 算法的估计正确率较高，均方根误差较小。

表 3-4 变量参数

变量	符号	值
阵元间距波长比	$c = \frac{d}{\lambda}$	0.5
阵元数	L	8
快拍数	N	$[12:256]$
入射角	θ	$[-90^\circ:90^\circ]$
信噪比	SNR	0dB

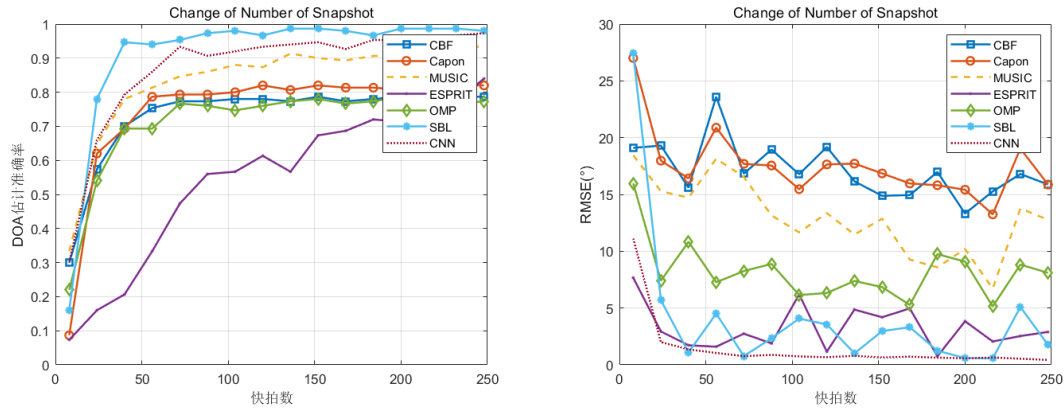


图 3-6 DOA 估计正确率与均方根误差随快拍数的关系

3.5 阵列元个数对 DOA 估计性能影响

仿真分析阵列元个数对 DOA 检测性能的影响，首先设置接收机及环境参数如表3-5。随机分布入射角，并以阵列元个数划分数数据集。对双目标入射角进行 DOA 估计。得到阵列元个数与 DOA 估计正确率、均方根误差关系如图3-7。可见 DOA 估计正确率随着阵列元个数的增大而增大，在环境参数条件较好时，准确率和均方根误差很快趋于稳定。

表 3-5 变量参数

变量	符号	值
阵元数	L	[2:32]
信噪比	SNR	0dB
快拍数	N	64

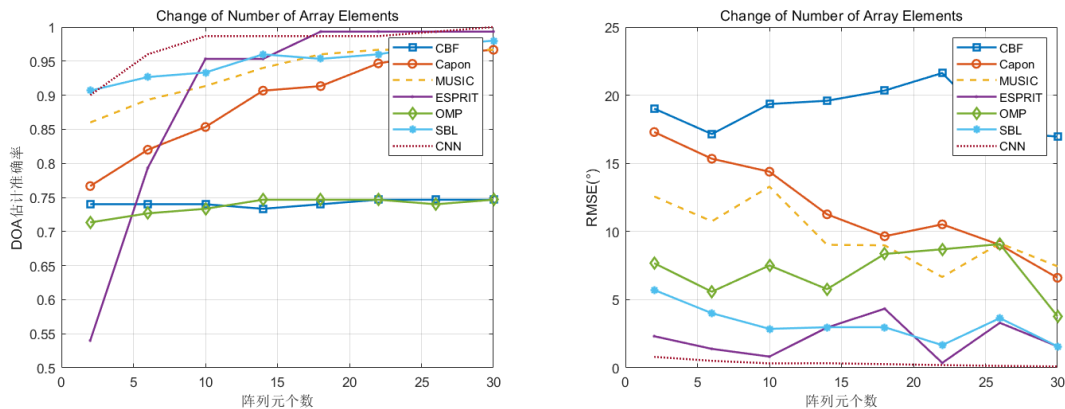


图 3-7 DOA 估计正确率与均方根误差随阵列元个数的关系

3.6 阵元误差对 DOA 估计性能影响

仿真分析阵元误差对 DOA 检测性能的影响，首先设置接收机及环境参数如表3-6。添加阵元误差后的 DOA 估计结果如图3-8，相较于图3-2，CBF 的估计谱波动更为明显，旁瓣较高。受阵元误差的影响，Capon 算法、MUSIC 算法与 CNN 算法在入射角 45° 的信号峰值较小。而 ESPRIT、OMP、SBL 依然保持较高的准确率，鲁棒性较强。继续加大阵元误差，Capon 算法、MUSIC 算法与 CNN 算法开始出现估计角度的偏移，DOA 估计不准确。

表 3-6 变量参数

变量	符号	值（分布）
阵元数	L	8
信号数	K	2
快拍数	N	128
入射角	θ	$-20^\circ, 45^\circ$
信噪比	SNR	$20dB$

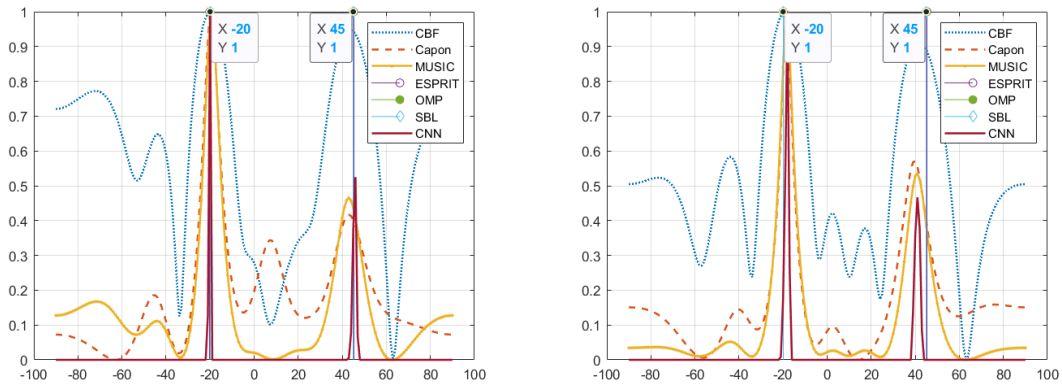


图 3-8 阵元误差对 DOA 估计性能影响

3.7 本章小结

本章将通过 MATLAB 与 Python 仿真并分析了 CBF、Capon、MUSIC、ESPRIT、OMP、SBL 与 CNN 的 DOA 估计方法。并分析其性能随 SNR、阵元数、阵元间距波长比、快拍数以及阵元误差的关系。

参考文献

- [1] Elbir A M. DeepMUSIC: Multiple Signal Classification via Deep Learning[J]. IEEE Sensors Letters, 2020, 4(4): 1-4.
- [2] Capon J. High-resolution frequency-wavenumber spectrum analysis[J]. Proceedings of the IEEE, 1969, 57(8):1408-1418.
- [3] 王永良. 空间谱估计理论与算法 [M]. 北京: 清华大学出版社, 2004.
- [4] 张小飞, 陈华伟, 仇小锋. 阵列信号处理及 MATLAB 实现 [M]. 电子工业出版社, 2015.
- [5] Emadi M, Miandji E, Unger J. OMP-based DOA estimation performance analysis[J]. Digital signal processing, 2018, 79: 57-65.