这是典型的 Weiner 滤波器。

由式 (5.8.12) 易得

$$C(n) = \lambda C(n-1) + s_x(n)\boldsymbol{x}(n)\boldsymbol{x}^{\mathrm{T}}(n) \tag{5.8.14a}$$

$$\boldsymbol{c}(n) = \lambda \boldsymbol{c}(n-1) + s_x(n)x(n)\boldsymbol{x}^{\mathrm{T}}(n) \tag{5.8.14b}$$

对式 (5.8.14a) 应用矩阵求逆引理，即可得到 RLS 算法如下 [60]。

算法 5.8.1 (基于三阶累积量的 RLS 自适应算法)

初始化

$\boldsymbol{P}(N_0) = \boldsymbol{C}^{-1}(N_0)$

更新 $k = N_0+1, N_0+2, \cdots$

$$s_x(n) = \sum_{i=-M}^{M} \beta(i)x(n+i)$$

$$\boldsymbol{K}(n) = \frac{s_x(n)\boldsymbol{P}(n-1)\boldsymbol{x}(n)}{\lambda + s_x(n)\boldsymbol{x}^{\mathrm{T}}(n)\boldsymbol{P}(n-1)\boldsymbol{x}(n)}$$

$$a(n) = x(n) - \boldsymbol{x}^{\mathrm{T}}(n)\boldsymbol{w}(n-1)$$

$$\boldsymbol{w}(n) = \boldsymbol{w}(n-1) + \boldsymbol{K}(n)a(n)$$

$$\boldsymbol{P}(n) = \lambda^{-1}\boldsymbol{P}(n-1) - \lambda^{-1}\boldsymbol{K}(n)\boldsymbol{x}^{\mathrm{T}}(n)\boldsymbol{P}(n-1)$$

上述算法中，遗忘因子 $\lambda < 1$ 取接近于 1 的数，例如 $\lambda = 0.995$ 或 0.997 等。

5.8.3 超定的辅助变量自适应算法

考虑超定方程

$$\boldsymbol{M}\boldsymbol{w} = \boldsymbol{r} \tag{5.8.15}$$

其中，矩阵 $\boldsymbol{M}$ 的行数大于列数。权向量的最小二乘解为

$$\boldsymbol{w} = (\boldsymbol{M}^{\mathrm{T}}\boldsymbol{M})^{-1}\boldsymbol{M}^{\mathrm{T}}\boldsymbol{r} \tag{5.8.16}$$

由于最小二乘解对噪声敏感，所以在噪声影响必须考虑时，需要使权向量的解在理论上能够对噪声免疫。辅助变量方法就是这样一种方法。

辅助变量方法的基本思想是引入一辅助变量矩阵 $\boldsymbol{Z}$，将原矩阵 $\boldsymbol{M}$ 和向量 $\boldsymbol{r}$ 分别变成

$$\boldsymbol{M} = \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{X} \tag{5.8.17a}$$

$$\boldsymbol{r} = \boldsymbol{Z}^{\mathrm{T}}\boldsymbol{y} \tag{5.8.17b}$$

式中，$\boldsymbol{X}$ 和 $\boldsymbol{y}$ 分别为适当选择的数据矩阵和数据向量，而 $\boldsymbol{Z}$ 称为辅助变量矩阵，其选择原则是：使 $\boldsymbol{M}=\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{X}$ 和 $\boldsymbol{r}=\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{y}$ 理论上能够对噪声免疫。这样一来，原超定方程的最小二乘解即变成

$$\boldsymbol{w}=(\boldsymbol{X}^{\mathrm{T}}\boldsymbol{Z}\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{X})^{-1}\boldsymbol{X}^{\mathrm{T}}\boldsymbol{Z}\boldsymbol{Z}^{\mathrm{T}}\boldsymbol{y} \tag{5.8.18}$$

这就是辅助变量方法的解。

下面以式 (5.5.4) 描述的 GM 方程

$$\sum_{i=0}^{q} b^2(i)R_x(m-i)=\epsilon_3\sum_{i=0}^{q} b(i)c_{3x}(m-i,m-i),\quad -q\leqslant m\leqslant 2q \tag{5.8.19}$$

为例，考查如何设计数据矩阵 $\boldsymbol{X}$、数据向量 $\boldsymbol{y}$ 和辅助变量矩阵 $\boldsymbol{Z}$。

式 (5.8.19) 可用矩阵形式写作

$$\left[\begin{array}{ccc:ccc} c_{3x}(-q) & \cdots & 0 & 0 & \cdots & 0\\ c_{3x}(-q+1) & \ddots & \vdots & R_x(-q) & \ddots & \vdots\\ \vdots & \ddots & c_{3x}(-q) & \vdots & \ddots & 0\\ c_{3x}(q) & \cdots & c_{3x}(-q+1) & R_x(q) & \cdots & R_x(-q)\\ \vdots & \ddots & \vdots & \vdots & \ddots & \vdots\\ 0 & \cdots & c_{3x}(q) & 0 & \cdots & R_x(q)\end{array}\right]\begin{bmatrix}\epsilon\\ \epsilon b_1\\ \vdots\\ \epsilon b_q\\ -b_1^2\\ \vdots\\ -b_q^2\end{bmatrix}$$

$$=\begin{bmatrix}R_x(-q)\\ R_x(-q+1)\\ \vdots\\ R_x(q)\\ 0\\ \vdots\\ 0\end{bmatrix} \tag{5.8.20}$$

为了将上式变成辅助变量矩阵方程形式，定义

$$\boldsymbol{Z}(k)=\begin{bmatrix}x(0) & \cdots & 0\\ \vdots & \ddots & \vdots\\ \vdots & \cdots & x(0)\\ \vdots & \cdots & \vdots\\ x(k) & \cdots & x(k-3q)\end{bmatrix} \tag{5.8.21a}$$

$$\boldsymbol{y}(k)=[0,\cdots,0,x(0),\cdots,x(k-q)]^{\mathrm{T}} \tag{5.8.21b}$$

和

$$
\boldsymbol{U}(k)=\begin{bmatrix}
0 & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & 0 \\
x^2(0) & \ddots & \vdots \\
\vdots & \ddots & 0 \\
\vdots & \cdots & x^2(0) \\
\vdots & \cdots & \vdots \\
x^2(k-q) & \cdots & x^2(k-2q)
\end{bmatrix} \tag{5.8.21c}
$$

$$
\boldsymbol{V}(k)=\begin{bmatrix}
0 & \cdots & 0 \\
\vdots & \ddots & \vdots \\
0 & \cdots & 0 \\
x(0) & \ddots & \vdots \\
\vdots & \ddots & 0 \\
\vdots & \cdots & x(0) \\
\vdots & \cdots & \vdots \\
x(k-q) & \cdots & x(k-2q)
\end{bmatrix} \tag{5.8.21d}
$$

构造

$$
\left.\begin{aligned}
\boldsymbol{M} &= \boldsymbol{Z}^{\mathrm{T}}(k)[\boldsymbol{U}(k) \mid \boldsymbol{V}(k)] \\
\boldsymbol{r} &= \boldsymbol{Z}^{\mathrm{T}}(k)\boldsymbol{y}(k)
\end{aligned}\right\} \tag{5.8.22}
$$

则容易看出，矩阵 $\boldsymbol{M}$ 和向量 $\boldsymbol{r}$ 的元素为观测数据 $x(0),x(1),\cdots,x(n)$ 的三阶累积量，所以具有对高斯有色噪声免疫的能力。容易看出，矩阵 $\boldsymbol{Z}^{\mathrm{T}}(k)\boldsymbol{U}(k)$ 和 $\boldsymbol{Z}^{\mathrm{T}}(k)\boldsymbol{V}(k)$ 第 i 行、第 j 列的元素分别为

$$
[\boldsymbol{Z}^{\mathrm{T}}(k)\boldsymbol{U}(k)]_{i,j}=\sum_{n=\max(i,q+j)}^{k} x(n-i)x^2(n-q-j) \tag{5.8.23a}
$$

$$
[\boldsymbol{Z}^{\mathrm{T}}(k)\boldsymbol{V}(k)]_{i,j}=\sum_{n=\max(i,q+j+1)}^{k} x(n-i)x(n-q-j-1) \tag{5.8.23b}
$$

类似地，向量 $\boldsymbol{Z}^{\mathrm{T}}(k)\boldsymbol{y}(k)$ 的元素为

$$
[\boldsymbol{Z}^{\mathrm{T}}(k)\boldsymbol{y}(k)]_i=\sum_{n=\max(i,q)}^{k} x(n-i)x(n-q) \tag{5.8.23c}
$$

求解超定的辅助变量方程 (5.8.18) 的自适应算法如下 [68]。

算法 5.8.2 (超定递推辅助变量 (ORIV) 算法)

初始化

$\boldsymbol{S}(0)=\mu[\boldsymbol{I}_{2q+1} \mid \boldsymbol{0}]$ $(2q+1)\times(3q+1)$ 矩阵

$\boldsymbol{c}(0)=\boldsymbol{0}$ $(3q+1)\times 1$ 向量

$\boldsymbol{P}(0)=\mu^{-2}\boldsymbol{I}_{2q+1}$ $(2q+1)\times(2q+1)$ 矩阵

$\boldsymbol{w}(0)=\boldsymbol{0}$ $(2q+1)\times 1$ 向量

循环 $k=0,1,2,\cdots$

$\boldsymbol{z}(k+1)=[x(k+1),\cdots,x(k-3q+1)]^{\mathrm{T}}$ $(3q+1)\times 1$ 向量

$\boldsymbol{y}(k+1)=[x^2(k-q+1),\cdots,x^2(k-2q+1) \mid x(k-q),\cdots,x(k-2q+1)]^{\mathrm{T}}$ $(2q+1)\times 1$向量

$\boldsymbol{v}(k+1)=\boldsymbol{S}(k)\boldsymbol{z}(k+1)$ $(2q+1)\times 1$ 向量

$\boldsymbol{S}(k+1)=\lambda\boldsymbol{S}(k)+\boldsymbol{y}(k+1)\boldsymbol{z}^{\mathrm{T}}(k+1)$ $(2q+1)\times(3q+1)$ 矩阵

$\boldsymbol{A}(k+1)=[\boldsymbol{v}(k+1) \mid \boldsymbol{y}(k+1)]$ $(2q+1)\times 2$ 矩阵

$$\boldsymbol{B}(k+1)=\begin{bmatrix}-\boldsymbol{z}^{\mathrm{T}}(k+1)\boldsymbol{z}(k+1) & \lambda\\ \lambda & 0\end{bmatrix} \quad 2\times 2 \text{ 矩阵}$$

$\boldsymbol{K}(k+1)=\boldsymbol{P}(k)\boldsymbol{A}(k+1)[\boldsymbol{B}(k+1)+\boldsymbol{A}^{\mathrm{T}}(k+1)\boldsymbol{P}(k)\boldsymbol{A}(k+1)]^{-1}$ $(2q+1)\times 2$矩阵

$\boldsymbol{P}(k+1)=\lambda^{-2}[\boldsymbol{P}(k)-\boldsymbol{K}(k+1)\boldsymbol{A}^{\mathrm{T}}(k+1)\boldsymbol{P}(k)]$ $(2q+1)\times(2q+1)$ 矩阵

$$\boldsymbol{u}(k+1)=\begin{bmatrix}\boldsymbol{z}^{\mathrm{T}}(k+1)\boldsymbol{c}(k)\\ x(k-q+1)\end{bmatrix} \quad 2\times 1 \text{ 向量}$$

$\boldsymbol{c}(k+1)=\lambda\boldsymbol{c}(k)+\boldsymbol{z}(k+1)x(k-q+1)$ $(3q+1)\times 1$ 向量

$\boldsymbol{w}(k+1)=\boldsymbol{w}(k)+\boldsymbol{K}(k+1)[\boldsymbol{u}(k+1)-\boldsymbol{A}^{\mathrm{T}}(k+1)\boldsymbol{w}(k)]$ $(2q+1)\times 1$ 向量

根据应用情况的不同，还有一些基于高阶累积量的自适应算法，感兴趣的读者可参考高阶统计量的专著 [142] 和 [227]。

5.9 时 延 估 计

在声纳、雷达、生物医学和地球物理等应用中，时延估计是一个重要的问题。例如，声纳和雷达的目标定位，地层构造 (如大坝等) 等问题都需要确定两个传感器接收信号之间的时间延迟或接收信号相对于发射信号之间的时间延迟。这两种时间延迟简称为时延。

5.9.1 广义互相关方法

考查两个空间分离的传感器，它们的测量数据 $x(n)$ 和 $y(n)$ 满足下列方程：

$$x(n) = s(n) + w_1(n) \tag{5.9.1a}$$

$$y(n) = s(n-D) + w_2(n) \tag{5.9.1b}$$

式中 $s(n)$ 为实信号，$s(n-D)$ 表示 $s(n)$ 的时延信号，其中时间移位为 D，可能还包括幅值因子 α；而 $w_1(n)$ 和 $w_2(n)$ 分别是两个传感器的观测噪声，它们相互统计独立，并且与信号 $s(n)$ 不相关。时延估计问题的提法是：利用观测数据 $x(n)$ 和 $y(n)$（其中 $n=1,\cdots,N$）估计时延参数 D。

本质上，时延估计就是寻求两个信号的最大相似性发生的时间差（滞后）。在信号处理中，"寻求二者之间的相似性"可以译为"求它们之间的互相关函数"。为此，我们来考查 $x(n)$ 与 $y(n)$ 之间的互相关函数

$$\begin{aligned} R_{xy}(\tau) &\stackrel{\text{def}}{=} \mathrm{E}\{x(n)y(n+\tau)\} \\ &= \mathrm{E}\{[s(n)+w_1(n)][s(n+\tau-D)+w_2(n+\tau)]\} \\ &= R_{ss}(\tau-D) \end{aligned} \tag{5.9.2}$$

式中 $R_{ss}(\tau)=\mathrm{E}\{s(n)s(n+\tau)\}$ 是信号 $s(n)$ 的自相关函数．由于自相关函数具有性质 $R_{ss}(\tau)\leqslant R_{ss}(0)$，所以互相关函数 $R_{xy}(\tau)=R_{ss}(\tau-D)$ 在 $\tau=D$ 处取最大值。换句话说，互相关函数取最大值时的滞后 τ 即给出时延 D 的估计。

由于使用 $x(n)$ 和 $y(n)$ 估计的互相关函数可能存在比较大的偏差，所以为了获得较好的时延估计，需要对估计的互相关函数进行平滑．例如，使用

$$R_{xy}(\tau) = \mathcal{F}^{-1}[P_{xy}(\omega)W(\omega)] = R_{xy}(\tau) * w(\tau) \tag{5.9.3}$$

估计时延参数 D，式中 $*$ 表示卷积，而 $P_{xy}(\omega)=\mathcal{F}[R_{xy}(\tau)]$ 是互相关函数 $R_{xy}(\tau)$ 的 Fourier 变换即 $x(n)$ 和 $y(n)$ 的互功率谱。这一方法是 Knapp 和 Carter 提出的，被称为广义互相关方法 [102]。

广义互相关方法的关键是平滑窗函数 $w(n)$ 的选择，下面是几种典型的窗函数。

1. 平滑相干变换窗

Knapp 和 Carter 提出使用窗函数 [102]

$$W(\omega) = \frac{1}{\sqrt{P_x(\omega)P_y(\omega)}} = H_1(\omega)H_2(\omega) \tag{5.9.4}$$

式中 $P_x(\omega)$ 和 $P_y(\omega)$ 分别是 $x(n)$ 和 $y(n)$ 的功率谱，而

$$H_1(\omega)=\frac{1}{\sqrt{P_x(\omega)}} \tag{5.9.5}$$

$$H_2(\omega)=\frac{1}{\sqrt{P_y(\omega)}} \tag{5.9.6}$$

由于在无噪声情况下 $y(n)=s(n-D)$ 是 $x(n)=s(n)$ 的相干信号，所以 $H_1(\omega)H_2(\omega)$ 可视为一相干变换，这正是称平滑窗函数 $W(\omega)$ 为平滑相干变换窗的缘故。

2. 最大似然窗或 Hannan-Thompson 窗

Chan 等人[38] 提出窗函数取

$$W(\omega)=\frac{z(\omega)}{|P_{xy}(\omega)|}=\frac{1}{|P_{xy}(\omega)|}\frac{|\gamma_{xy}(\omega)|}{1-|\gamma_{xy}(\omega)|^2} \tag{5.9.7}$$

式中

$$|\gamma_{xy}(\omega)|^2=\frac{|P_{xy}(\omega)|^2}{P_x(\omega)P_y(\omega)} \tag{5.9.8}$$

为幅值平方的相干系数，其取值在 0 和 1 之间。Chan 等人证明了，对于零均值的不相关高斯过程 $x(n)$ 和 $y(n)$ 而言，

$$z(\omega)=\frac{|\gamma_{xy}(\omega)|}{1-|\gamma_{xy}(\omega)|^2}\propto\frac{1}{P_{xy}(\omega)\ \text{的相位方差}} \tag{5.9.9}$$

式中 $a\propto b$ 表示 a 与 b 成正比。

在使用最大似然窗的广义互相关方法里，与互相关函数

$$R_{xy}(\tau)=\mathcal{F}^{-1}\left[\frac{P_{xy}(\omega)}{|P_{xy}(\omega)|}z(\omega)\right] \tag{5.9.10}$$

的最大幅值相对应的滞后 τ 即作为时延参数 D 的估计。

5.9.2 高阶统计量方法

在很多实际应用 (例如被动式和主动式声纳) 中，信号 $s(n)$ 往往是非高斯过程，而加性噪声 $w_1(n)$ 和 $w_2(n)$ 则是高斯过程[166],[201]。在这类情况下，使用高阶统计量作时延估计更加合理，因为高斯噪声理论上可以得到完全抑制。

令 $x(n)$ 和 $y(n)$ 都是零均值化的观测数据，则信号 $x(n)$ 的三阶累积量为

$$\begin{aligned}c_{3x}(\tau_1,\tau_2)&\overset{\text{def}}{=}\mathrm{E}\{x(n)x(n+\tau_1)x(n+\tau_2)\}\\&=c_{3s}(\tau_1,\tau_2)\end{aligned} \tag{5.9.11}$$

式中

$$c_{3s}(\tau_1,\tau_2) \stackrel{\text{def}}{=} \mathrm{E}\{s(n)s(n+\tau_1)s(n+\tau_2)\} \tag{5.9.12}$$

定义信号 $x(n)$ 和 $y(n)$ 之间的互三阶累积量

$$\begin{aligned} c_{xyx}(\tau_1,\tau_2) &\stackrel{\text{def}}{=} \mathrm{E}\{x(n)y(n+\tau_1)x(n+\tau_2)\} \\ &= c_{3s}(\tau_1-D,\tau_2) \end{aligned} \tag{5.9.13}$$

分别对式 (5.9.11) 和式 (5.9.13) 作二维 Fourier 变换，则得双谱关系式

$$P_{3x}(\omega_1,\omega_2) = P_{3s}(\omega_1,\omega_2) \tag{5.9.14}$$

和互双谱关系式

$$P_{xyx}(\omega_1,\omega_2) = P_{3s}(\omega_1,\omega_2)\mathrm{e}^{\mathrm{j}\omega_1 D} \tag{5.9.15}$$

此外，我们还可写出

$$P_{3x}(\omega_1,\omega_2) = |P_{3x}(\omega_1,\omega_2)|\mathrm{e}^{\mathrm{j}\phi_{3x}(\omega_1,\omega_2)} \tag{5.9.16}$$

$$P_{xyx}(\omega_1,\omega_2) = |P_{xyx}(\omega_1,\omega_2)|\mathrm{e}^{\mathrm{j}\phi_{xyx}(\omega_1,\omega_2)} \tag{5.9.17}$$

式中 $\phi_{3x}(\omega_1,\omega_2)$ 和 $\phi_{xyx}(\omega_1,\omega_2)$ 分别代表双谱 $P_{3x}(\omega_1,\omega_2)$ 和互双谱 $P_{xyx}(\omega_1,\omega_2)$ 的相位。

下面是时延估计的几种高阶统计量方法。

1. 非参数化双谱方法 1 [166],[167]

定义函数

$$I(\omega_1,\omega_2) = \frac{P_{xyx}(\omega_1,\omega_2)}{P_{3x}(\omega_1,\omega_2)} \tag{5.9.18}$$

将式 (5.9.14) 和式 (5.9.15) 代入式 (5.9.18)，可将函数 $I(\omega_1,\omega_2)$ 改写作

$$I_1(\omega_1,\omega_2) = \mathrm{e}^{\mathrm{j}\omega_1 D} \tag{5.9.19}$$

故函数

$$\begin{aligned} T_1(\tau) &= \int_{-\infty}^{\infty}\int_{-\infty}^{\infty} I_1(\omega_1,\omega_2)\mathrm{e}^{-\mathrm{j}\omega_1\tau}\mathrm{d}\omega_1\mathrm{d}\omega_2 \\ &= \int_{-\infty}^{\infty}\mathrm{d}\omega_2\int_{-\infty}^{\infty}\mathrm{e}^{\mathrm{j}\omega_1(D-\tau)}\mathrm{d}\omega_1 \end{aligned} \tag{5.9.20}$$

在 $\tau = D$ 处取峰值。

2. 非参数化双谱方法 2 [140]

利用互双谱 $P_{xyx}(\omega_1,\omega_2)$ 的相位与双谱 $P_{3x}(\omega_1,\omega_2)$ 的相位之差，可以定义一新的相位

$$\phi(\omega_1,\omega_2)=\phi_{xyx}(\omega_1,\omega_2)-\phi_{3x}(\omega_1,\omega_2) \tag{5.9.21}$$

然后，利用这一新的相位，即可构造一新的函数

$$I_2(\omega_1,\omega_2)=\mathrm{e}^{\mathrm{j}\phi(\omega_1,\omega_2)} \tag{5.9.22}$$

则由式 (5.9.14) ~ 式 (5.9.17) 知

$$I_2(\omega_1,\omega_2)=\mathrm{e}^{\mathrm{j}\omega_1 D} \tag{5.9.23}$$

故函数

$$\begin{aligned} T_2(\tau)&=\int_{-\infty}^{\infty}\int_{-\infty}^{\infty}I_2(\omega_1,\omega_2)\mathrm{e}^{-\mathrm{j}\omega_1\tau}\mathrm{d}\omega_1\mathrm{d}\omega_2 \\ &=\int_{-\infty}^{\infty}\mathrm{d}\omega_2\int_{-\infty}^{\infty}\mathrm{e}^{\mathrm{j}\omega_1(D-\tau)}\mathrm{d}\omega_1 \end{aligned} \tag{5.9.24}$$

也在 $\tau=D$ 处取峰值.

算法 5.9.1 (时延估计的非参数化双谱方法)

步骤 1 将 N 个数据样本分为 K 段，每段含 M 个数据，并且相邻两段之间有 50% 数据重叠，即 $N=KM/2$。第 k 段的数据记作 $x^{(k)}(n)$ 和 $y^{(k)}(n)$，其中 $k=1,\cdots,K$; $n=0,1,\cdots,M-1$.

步骤 2 对每段数据计算下列离散 Fourier 变换:

$$X^{(k)}(\omega)=\sum_{n=0}^{M-1}x^{(k)}(n)\mathrm{e}^{-\mathrm{j}\frac{n\omega}{M}} \tag{5.9.25}$$

$$Y^{(k)}(\omega)=\sum_{n=0}^{M-1}y^{(k)}(n)\mathrm{e}^{-\mathrm{j}\frac{n\omega}{M}} \tag{5.9.26}$$

式中 $k=1,\cdots,K$。

步骤 3 估计每段数据的双谱和互双谱

$$P_{3x}^{(k)}(\omega_1,\omega_2)=X^{(k)}(\omega_1)X^{(k)}(\omega_2)X^{(k)*}(\omega_1+\omega_2) \tag{5.9.27}$$

$$P_{xyx}^{(k)}(\omega_1,\omega_2)=X^{(k)}(\omega_1)Y^{(k)}(\omega_2)X^{(k)*}(\omega_1+\omega_2) \tag{5.9.28}$$

式中 $k=1,\cdots,K$，而 $X^{(k)*}(\omega)$ 表示 $X^{(k)}(\omega)$ 的复数共轭。

步骤 4 平滑 K 段双谱和互双谱，得到 N 个数据的双谱和互双谱:

$$\hat{P}_{3x}(\omega_1,\omega_2) = \frac{1}{K}\sum_{k=1}^{K} P_{3x}^{(k)}(\omega_1,\omega_2) \tag{5.9.29}$$

$$\hat{P}_{xyx}(\omega_1,\omega_2) = \frac{1}{K}\sum_{k=1}^{K} P_{xyx}^{(k)}(\omega_1,\omega_2) \tag{5.9.30}$$

步骤 5 计算双谱和互双谱的相位

$$\hat{\phi}_{3x}(\omega_1,\omega_2) = \arctan\left\{\frac{\mathrm{Im}[\hat{P}_{3x}(\omega_1,\omega_2)]}{\mathrm{Re}[\hat{P}_{3x}(\omega_1,\omega_2)]}\right\} \tag{5.9.31}$$

$$\hat{\phi}_{xyx}(\omega_1,\omega_2) = \arctan\left\{\frac{\mathrm{Im}[\hat{P}_{xyx}(\omega_1,\omega_2)]}{\mathrm{Re}[\hat{P}_{xyx}(\omega_1,\omega_2)]}\right\} \tag{5.9.32}$$

步骤 6 计算

$$\hat{\phi}(\omega_1,\omega_2) = \hat{\phi}_{xyx}(\omega_1,\omega_2) - \hat{\phi}_{3x}(\omega_1,\omega_2) \tag{5.9.33}$$

并构造

$$\hat{I}_2(\omega_1,\omega_2) = e^{j\hat{\phi}(\omega_1,\omega_2)} \tag{5.9.34}$$

步骤 7 计算

$$\hat{T}_2(\tau) = \sum_{\omega_1=0}^{M-1}\sum_{\omega_2=0}^{M-1} \hat{I}_2(\omega_1,\omega_2)e^{-j\omega_1\tau}$$

步骤 8 选择 $\hat{T}_2(\tau)$ 取最大值时的 τ 值作为时延参数的估计 $\hat{D}$。

显然，只要将上述算法中的式 (5.9.34) 换成式 (5.9.18)，即可得到 Sasaki 等人估计时延参数的算法 [166]。

3. 互倒双谱方法 [141]

上面的方法都不是直接估计时延参数，属非直接法一类。下面介绍 Nikias 和 Liu 提出的一种直接法 [141]。这种方法同时使用互倒双谱和互双谱。

由式 (5.9.18) 和式 (5.9.19)，可以用 Z 变换形式将式 (5.9.18) 改写作

$$\frac{P_{3x}(z_1,z_2)}{P_{xyx}(z_1,z_2)} = z_1^{-D} \tag{5.9.35}$$

式中 $z_1^{-D} = e^{j\omega_1 D}$，且

$$P_{3x}(\omega_1,\omega_2) = P_{3x}(z_1,z_2)\big|_{z_1=e^{-j\omega_1},z_2=e^{-j\omega_2}} \tag{5.9.36a}$$

$$P_{xyx}(\omega_1,\omega_2) = P_{xyx}(z_1,z_2)\big|_{z_1=e^{-j\omega_1},z_2=e^{-j\omega_2}} \tag{5.9.36b}$$

若取式 (3.8.35) 的复对数，则有

$$\ln[P_{xyx}(z_1,z_2)] - \ln[P_{3x}(z_1,z_2)] = -D\ln\frac{1}{z_1}\ln[z_1] \tag{5.9.37}$$

式中 $\ln[P_{3x}(z_1,z_2)]$ 和 $\ln[P_{xyx}(z_1,z_2)]$ 分别称为 $x(n)$ 的倒双谱和 $x(n)$ 与 $y(n)$ 的互倒双谱。求式 (5.9.37) 相对于 z_1 的偏导数，则有

$$\frac{1}{P_{xyx}(z_1,z_2)}\frac{\partial P_{xyx}(z_1,z_2)}{\partial z_1} - \frac{1}{P_{3x}(z_1,z_2)}\frac{\partial P_{3x}(z_1,z_2)}{\partial z_1} = \frac{D}{z_1} \tag{5.9.38}$$

与之对应的时域表达式为

$$\begin{aligned} & c_{3x}(m,n) * [m\cdot c_{xyx}(m,n)] - c_{xyx}(m,n) * [m\cdot m_{3x}(m,n)] \\ & = Dc_{xyx}(m,n) * c_{3x}(m,n) \end{aligned} \tag{5.9.39}$$

对上式两边作关于变量 m 和 n 的二维 Fourier 变换，即可得到

$$D(\omega_1,\omega_2) = \frac{\mathcal{F}_2[m\cdot c_{xyx}(m,n)]}{\mathcal{F}_2[c_{xyx}(m,n)]} - \frac{\mathcal{F}_2[m\cdot c_{3x}(m,n)]}{\mathcal{F}_2[c_{3x}(m,n)]} \tag{5.9.40}$$

式中

$$\mathcal{F}_2[m\cdot c_{xyx}(m,n)] = \int_{-\infty}^{\infty} m\cdot c_{xyx}(m,n)\mathrm{e}^{-\mathrm{j}(\omega_1 m+\omega_2 n)}\mathrm{d}m\mathrm{d}n$$

$$\mathcal{F}_2[m\cdot c_{3x}(m,n)] = \int_{-\infty}^{\infty} m\cdot c_{3x}(m,n)\mathrm{e}^{-\mathrm{j}(\omega_1 m+\omega_2 n)}\mathrm{d}m\mathrm{d}n$$

$D(\omega_1,\omega_2)$ 的峰值即给出时延参数 D 的估计。注意，为了减小估计误差和方差，可以采取算法 5.9.1 的分段平滑法：先求出 K 段数据的 $\hat{D}^{(k)}(\omega_1,\omega_2)$，再计算

$$\hat{D}(\omega_1,\omega_2) = \frac{1}{K}\sum_{k=1}^{K}\hat{D}^{(k)}(\omega_1,\omega_2) \tag{5.9.41}$$

最后，使用其峰值作为时延 D 的估计值。

4. 四阶统计量方法 [190]

时延估计问题可以通过优化方法求解。 Chan 等人 [39] 指出，无加权即加矩形窗函数的广义互相关方法等价于选择时延 D，使二阶统计量的代价函数

$$J_2(D) = \mathrm{E}\{[x(n-D)-y(n)]^2\} \tag{5.9.42}$$

最小化。