9

Time Delay Estimation and Acoustic Source Localization

With the material presented in Chap. 2 on MIMO system modeling and that in Chap. 6 on blind channel identification, we are now ready to investigate the problem of time delay estimation (TDE) and its application for acoustic source localization. We begin this chapter by outlining the TDE problem, and then address the estimation of time-difference-of-arrival (TDOA) information by measuring the cross correlation between receivers' outputs and identifying the channel impulse responses from the radiating source to the multiple receivers. The chapter also discusses how to employ the TDOA measurements to locate radiating sources in the acoustic wavefield that emit signals to receivers.

9.1 Time Delay Estimation

In an acoustic wavefield with multiple receivers at distinct spatial locations, a wavefront, emanating from a radiating sound source, arrives at different receiving sensors at different time instances. The corresponding time differences of arrival between pairs of receivers are important parameters in acoustic MIMO signal processing, which characterize the direction and location from which the signal impinges on the receivers. The estimation of such parameters, often referred to as the time-delay-estimation (TDE) problem, has been of fundamental importance.

In an acoustic environment, speech sound often arrives at each individual sensor through multiple paths. To make the analysis easy to understand, however, it is convenient to decouple the problem into two cases, i.e, single-path and multipath problems. The former is an ideal case, where each sensor's output can be modeled as a delayed and attenuated copy of the source signal corrupted by additive noise. Assuming that there is only one (unknown) source in the sound field, we can write the output of receiver n ($n = 1, 2, \cdots, N$) as

$$x_n(k) = \alpha_n s(k - D_n) + b_n(k), \tag{9.1}$$

where α_n, which satisfies $0 \leq \alpha_n \leq 1$, is an attenuation factor due to propagation effects, D_n corresponds to the propagation time from the unknown source to receiver n, and $s(k)$, which is often speech from either a talker or a loudspeaker, is broadband in nature. In the context of TDE, it is assumed that the noise component $b_n(k)$ in (9.3) is a Gaussian random process, which is uncorrelated with both the source signal and the noise signals at other sensors.

With this signal model, the time difference of arrival (TDOA) between the ith and jth sensors can be written as

$$\tau_{ij} = D_j - D_i, \tag{9.2}$$

where $i, j = 1, 2, \cdots, N$, and $i \neq j$. The goal of TDE, then, is to obtain an estimate $\hat{\tau}_{ij}$ of τ_{ij} based on the observation signals $x_n(k)$.

In the multipath situation, the output of each receiver can be modeled as a convolution between the source signal and the corresponding channel impulse response from the source to the receiver, as given in (2.4) and (2.5). Let us rewrite (2.4) here:

$$x_n(k) = \mathbf{h}_n^T \mathbf{s}(k) + b_n(k), \quad n = 1, 2, \cdots, N. \tag{9.3}$$

Again it is assumed that $\mathbf{s}(k)$ is reasonably broadband and each $b_n(k)$ is a Gaussian random process. In addition, we assume that in $\mathbf{h}_n$ the entry corresponding to the direct path from the source to the nth receiver has the maximal magnitude out of all the elements so that once we know the impulse responses $\mathbf{h}_n$, $n = 1, 2, \cdots, N$, the TDOA information can be determined by identifying the direct paths. This assumption is reasonable from the wave propagation point of view since reflected signals take a longer path to reach the receivers and suffer energy absorption from reflection boundaries. However, multipath signals may combine at sensors both destructively and constructively in a non-coherent and unpredictable manner. As a result, it may happen occasionally that the combination of several multipath signals has a higher energy than the direct-path signal. In such a case, the correct time delay estimate can be determined by either taking into account previous estimates or incorporating some *a priori* knowledge about the source.

Comparing (9.3) with (9.1), we can see that the single-path problem is a special case of the multipath problem. As a matter of fact, if set

$$\begin{aligned}
\mathbf{h}_n &= \begin{bmatrix} h_{n,0} & h_{n,1} & \cdots & h_{n,l} & \cdots & h_{n,L-1} \end{bmatrix}^T \\
&= \begin{bmatrix} 0 & 0 & \cdots & 0 & h_{n,D_n} & 0 & \cdots & 0 \end{bmatrix}^T \\
&= \begin{bmatrix} 0 & 0 & \cdots & 0 & \alpha_n & 0 & \cdots & 0 \end{bmatrix}^T, \quad n = 1, 2, \cdots, N,
\end{aligned} \tag{9.4}$$

then (9.3) is the same as (9.1). In other words, in the single-path situation, the channel impulse response $\mathbf{h}_n$ has only one non-zero element at $l = D_n$, where $h_{n,D_n} = \alpha_n$ is governed by the propagation attenuation.

Numerous algorithms have been developed to fulfil the estimation of TDOA in the ideal propagation situation. The most widely used technique thus far is, perhaps, the generalized cross-correlation (GCC) algorithm [205], which originated from the research of passive sonar signal processing. In this approach, the delay estimate $\hat{\tau}_{ij}$ is obtained as the lag time that maximizes the generalized cross-correlation function between the filtered versions of signals $x_i(k)$ and $x_j(k)$. The GCC method has been intensively investigated and can achieve reasonably accurate estimation of time delay in the presence of weak to moderate levels of noise but absence of the multipath effect.

However, in acoustical environments in the context of speech communication, as we mentioned earlier, each sensor receives not only the direct-path signal, but also multiple attenuated and delayed replicas of the source signal due to reflections of the wavefront from boundaries and objects in the enclosure environments. This multipath propagation effect introduces echoes and spectral distortion into the observation signals, termed as reverberation, which may severely deteriorate the TDE performance of the GCC technique. In addition, the TDE problem is often complicated by conditions of poor signal-to-noise ratio, nonstationarity of speech signals, and a changing TDOA owing to the motion of the speech sources.

In the first half of this chapter, we address the problem of time delay estimation in acoustic environments with microphone receivers. Starting with the simple cross-correlation method, we will study the problem with emphasis on how to deal with strong noise and reverberation. Specifically, we discuss three ways in improving the robustness of TDE with respect to noise and reverberation. The first is to incorporate some *a priori* knowledge about the distortion sources into the cross-correlation method to ameliorate its performance. The second is to use multiple (more than two) sensors and take advantage of the redundancy to enhance the delay estimate between the two desired sensors. The third is to exploit the advanced system identification techniques to improve TDE in reverberation conditions.

9.2 Cross-Correlation Method

Suppose that we have two sensors, with their outputs being denoted as $x_1(k)$ and $x_2(k)$ respectively. The cross-correlation function (CCF) between the two signals is defined as:

$$r_{x_1x_2}(p) = E\left[x_1(k)x_2(k+p)\right]. \tag{9.5}$$

Considering the ideal propagation model given in (9.1) and substituting for $x_n(k)$ $(n = 1, 2)$ in terms of $s(k)$ and $b_n(k)$, we can readily deduce that:

$$\begin{aligned} r_{x_1x_2}(p) = \alpha_1\alpha_2 r_{ss}(p + D_1 - D_2) + \alpha_1 r_{sb_2}(p + D_1) + \\ \alpha_2 r_{sb_1}(p - D_2) + r_{b_1b_2}(p). \end{aligned} \tag{9.6}$$

With the assumption that $b_n(k)$ is Gaussian random noise, which is uncorrelated with both the signal and the noise observed at the other sensor, it can be easily checked that $r_{x_1x_2}(p)$ reaches its maximum at $p = D_2 - D_1$. Therefore, given the CCF, we can obtain an estimate of the TDOA between $x_1(k)$ and $x_2(k)$ as

$$\hat{\tau}_{12} = \arg\max_p r_{x_1x_2}(p), \tag{9.7}$$

where $p \in [-\tau_{\max}, \tau_{\max}]$, and $\tau_{\max}$ is the maximum possible delay.

In digital implementation of (9.7), some approximations are required because the CCF is not known and must be estimated. A normal practice is to replace the CCF defined in (9.5) by its time-averaged estimate. Suppose at time instant t we have a set of observation samples of x_n, $\{x_n(t), x_n(t+1), \cdots, x_n(t+k-1), \cdots, x_n(t+K-1)\}$, $n = 1, 2$, the corresponding CCF can be estimated either as

$$\hat{r}_{x_1x_2}(p) = \begin{cases} \dfrac{1}{K} \displaystyle\sum_{k=0}^{K-p-1} x_1(t+k)x_2(t+k+p), & p \geq 0 \\ \hat{r}_{x_2x_1}(-p), & p < 0 \end{cases}, \tag{9.8}$$

or by

$$\hat{r}_{x_1x_2}(p) = \begin{cases} \dfrac{1}{K-p} \displaystyle\sum_{k=0}^{K-p-1} x_1(t+k)x_2(t+k+p), & p \geq 0 \\ \hat{r}_{x_2x_1}(-p), & p < 0 \end{cases}, \tag{9.9}$$

where K is the block size. The difference between (9.8) and (9.9) is that the former leads to a biased estimator, while the latter is an unbiased one. However, since it has a lower estimation variance and is asymptotically unbiased, the former has been widely adopted in many applications.

Another way to estimate CCF is through the discrete Fourier transform (DFT) and the inverse discrete Fourier transform (IDFT), i.e.,

$$\hat{r}_{x_1x_2}(p) = \frac{1}{K} \sum_{k'=0}^{K-1} X_1(\omega_{k'})X_2^*(\omega_{k'})e^{j\omega_{k'}p}, \tag{9.10}$$

where

$$\omega_{k'} = \frac{2\pi k'}{K}, \quad k' = 0, 1, \cdots, K-1, \tag{9.11}$$

is the angular frequency,

$$X_n(\omega_{k'}) = \sum_{k=0}^{K-1} x_n(t+k)e^{-j\omega_{k'}k} \tag{9.12}$$

is the short-term DFT of $x_n(k)$ at time t. Both (9.8) and (9.10) generate the same CCF estimate. However, the latter can be implemented more efficiently

Table 9.1. The cross-correlation method for time delay estimation.

Parameter:	$\hat{\tau}_{12}$
Estimation:	For $t = 0, 1, \cdots$ (a) Obtain a frame of observation signals at time instant t: $\{x_n(t), x_n(t+1), \cdots, x_n(t+K-1)\}$, $n = 1, 2$ (b) Estimate the spectrum of x_n: $X_n(\omega_{k'}) = \mathrm{FFT}\{x_n(t+k)\} = \sum_{k=0}^{K-1} x_n(t+k) e^{-j\omega_{k'} k}$ (c) Estimate cross-correlation function: $\hat{r}_{x_1x_2}(p) = \mathrm{IFFT}\{X_1(\omega_{k'}) X_2^*(\omega_{k'})\}$ $= \frac{1}{K} \sum_{k'=0}^{K-1} X_1(\omega_{k'}) X_2^*(\omega_{k'}) e^{j\omega_{k'} p}$ (d) Obtain the time delay: $\hat{\tau}_{12} = \arg\max_p \hat{r}_{x_1x_2}(p)$

using the fast Fourier transform (FFT) and the inverse fast Fourier transform (IFFT), and therefore it has been widely adopted in systems.

The CC method for TDE is summarized in Table 9.1.

9.3 Magnitude-Difference Method

It can be seen that the cross-correlation function reaches its maximum when two realizations of the same source signal are synchronized. Therefore, the CC method indeed obtains the TDOA estimate by measuring the synchrony between the two signals. Another way to measure the synchrony is by identifying the minimum of the magnitude-difference function (MDF), which is defined as

$$\Psi_{x_1x_2}(p) = E\{|x_1(k) - x_2(k+p)|\}. \tag{9.13}$$

With the signal model given in (9.1), we can derive that [189], [190]:

$$\Psi_{x_1x_2}(p) = \sqrt{\frac{2}{\pi}\left\{E[x_1^2(k)] + E[x_2^2(k)] - 2r_{x_1x_2}(p)\right\}}. \tag{9.14}$$

It can be checked that $\Psi_{x_1x_2}(p)$ reaches its minimum at $p = D_2 - D_1$. Therefore, given the MDF, we can obtain the TDOA between $x_1(k)$ and $x_2(k)$ as

$$\hat{\tau}_{12} = \arg\min_p \Psi_{x_1x_2}(p). \tag{9.15}$$

Similar to the CC method, we have to estimate $\Psi_{x_1x_2}(p)$ in the implementation of (9.15). Normally, MDF is approximated by the average-magnitude-difference function (AMDF), which at time instance t is computed as

Table 9.2. The magnitude-difference method for time delay estimation.

```
Parameter:   $\hat{\tau}_{12}$
Estimation:  For $t = 0, 1, \cdots$
               (a) Obtain a frame of observation signals at time instant $t$:
                   $\{x_n(t), x_n(t+1), \cdots, x_n(t+K-1)\}$, $n = 1, 2$
               (b) Estimate the AMDF between $x_1$ and $x_2$:
                   If $p \geq 0$:
                     $\hat{\Psi}_{x_1x_2}(p) = \frac{1}{K-p} \sum_{k=0}^{K-1-p} |x_1(t+k) - x_2(t+k+p)|$
                   If $p < 0$:
                     $\hat{\Psi}_{x_1x_2}(p) = \hat{\Psi}_{x_2x_1}(-p)$
               (c) Obtain the time delay:
                   $\hat{\tau}_{12} = \arg\min_{p} \hat{\Psi}_{x_1x_2}(p)$
```

$$\hat{\Psi}_{x_1x_2}(p) = \begin{cases} \dfrac{1}{K} \displaystyle\sum_{k=0}^{K-1-p} |x_1(t+k) - x_2(t+k+p)|, & p \geq 0 \\ \hat{\Psi}_{x_2x_1}(-p), & p < 0 \end{cases} . \tag{9.16}$$

The algorithm to estimate time delay based on AMDF is summarized in Table 9.2.

9.4 Maximum Likelihood Method

Maximum likelihood (ML) is one of the most popularly used statistical estimators due to its asymptotic optimal property, i.e., the estimation variance can achieve the Cramèr-Rao lower bound (CRLB) when the number of observation samples approaches infinity. For the signal model given in (9.1), if we assume that both $s(k)$ and $b_n(k)$ $(n = 1, 2)$ are mutually independent, zero-mean, stationary Gaussian random processes, D_n are constant over time, and the observation interval approximates infinity, i.e, $K \to \infty$, we can then derive the ML estimator either in the time domain or in the frequency domain [205], [67], [287]. In the frequency domain, the DFT of $x_n(k)$ is formed as:

$$\begin{aligned} X_n(\omega_{k'}) &= \sum_{k=0}^{K-1} x_n(k) e^{-j\omega_{k'}k} \\ &= \sum_{k=0}^{K-1} \left[\alpha_n s_n(k - D_n) + b_n(k)\right] e^{-j\omega_{k'}k} \\ &= \alpha_n S(\omega_{k'}) e^{-j\omega_{k'}D_n} + B_n(\omega_{k'}), \quad n = 1, 2, \end{aligned} \tag{9.17}$$

which is a Gaussian random variable. It follows that:

$$\begin{aligned} E\left[X_n(\omega_i)X_n^*(\omega_j)\right] &= \begin{cases} \alpha_n^2 P_s(\omega_i) + P_{b_n}(\omega_i), & i = j \\ 0, & i \neq j \end{cases}, \\ E\left[X_1(\omega_i)X_2^*(\omega_j)\right] &= \begin{cases} \alpha_1\alpha_2 P_s(\omega_i)e^{-j\omega_i(D_1-D_2)}, & i = j \\ 0, & i \neq j \end{cases}, \end{aligned} \tag{9.18}$$

where $n = 1, 2$, $P_s(\omega_i) = E[S(\omega_i)S^*(\omega_i)]$ and $P_{b_n}(\omega_i) = E[B_n(\omega_i)B_n^*(\omega_i)]$ are the power spectral densities of $s(k)$ and $b_n(k)$ respectively.

Now let us define three vectors:

$$\begin{aligned} \mathbf{X}_1 &\triangleq \left[\, X_1(\omega_0) \;\; X_1(\omega_1) \;\; \cdots \;\; X_1(\omega_{K-1}) \,\right]^T, \\ \mathbf{X}_2 &\triangleq \left[\, X_2(\omega_0) \;\; X_2(\omega_1) \;\; \cdots \;\; X_2(\omega_{K-1}) \,\right]^T, \\ \mathbf{X} &\triangleq \left[\, \mathbf{X}_1^T \;\; \mathbf{X}_2^T \,\right]^T. \end{aligned} \tag{9.19}$$

The covariance matrix of $\mathbf{X}$ can be written as

$$\begin{aligned} \mathbf{\Sigma} &\triangleq E\left[\mathbf{X}\mathbf{X}^H\right] \\ &= E\begin{bmatrix} \mathbf{X}_1\mathbf{X}_1^H & \mathbf{X}_1\mathbf{X}_2^H \\ \mathbf{X}_2\mathbf{X}_1^H & \mathbf{X}_2\mathbf{X}_2^H \end{bmatrix} \\ &\triangleq \begin{bmatrix} \mathbf{\Sigma}_{11} & \mathbf{\Sigma}_{12} \\ \mathbf{\Sigma}_{12}^H & \mathbf{\Sigma}_{22} \end{bmatrix}. \end{aligned}$$

From (9.18) and (9.19), we can check that $\mathbf{\Sigma}_{11}$, $\mathbf{\Sigma}_{22}$, and $\mathbf{\Sigma}_{12}$ all are diagonal matrices of size $K \times K$:

$$\begin{aligned} \mathbf{\Sigma}_{11} = E\left[\mathbf{X}_1\mathbf{X}_1^H\right] = \operatorname{diag}\Big[\, & \alpha_1^2 P_s(\omega_0) + P_{b_1}(\omega_0),\; \alpha_1^2 P_s(\omega_1) + P_{b_1}(\omega_1), \\ & \cdots,\; \alpha_1^2 P_s(\omega_{K-1}) + P_{b_1}(\omega_{K-1}) \,\Big], \end{aligned}$$

$$\begin{aligned} \mathbf{\Sigma}_{22} = E\left[\mathbf{X}_2\mathbf{X}_2^H\right] = \operatorname{diag}\Big[\, & \alpha_2^2 P_s(\omega_0) + P_{b_2}(\omega_0),\; \alpha_2^2 P_s(\omega_1) + P_{b_2}(\omega_1), \\ & \cdots,\; \alpha_2^2 P_s(\omega_{K-1}) + P_{b_2}(\omega_{K-1}) \,\Big], \end{aligned}$$

and

$$\begin{aligned} \mathbf{\Sigma}_{12} = E\left[\mathbf{X}_1\mathbf{X}_2^H\right] = \operatorname{diag}\Big[\, & \alpha_1\alpha_2 P_s(\omega_0)e^{j\omega_0\tau_{12}},\; \alpha_1\alpha_2 P_s(\omega_1)e^{j\omega_1\tau_{12}}, \\ & \cdots,\; \alpha_1\alpha_2 P_s(\omega_{K-1})e^{j\omega_{K-1}\tau_{12}} \,\Big], \end{aligned}$$

where $\tau_{12} = (D_2 - D_1)$.

The log-likelihood function of $\mathbf{X}$, given the delay, attenuation factors, and the power spectral densities of both the speech and noise signals, can be written as:

$$\mathcal{L} \;=\; \ln p\left[\mathbf{X} \mid D_1, D_2, \alpha_1, \alpha_2, P_s(\omega_{k'}), P_{b_1}(\omega_{k'}), P_{b_2}(\omega_{k'})\right]. \tag{9.20}$$

Since $\mathbf{X}$ is Gaussian distributed, we have

$$\mathcal{L} \;=\; -\frac{K}{2}\ln(2\pi) - \frac{1}{2}\ln[\det(\mathbf{\Sigma})] - \frac{1}{2}\mathbf{X}^H\mathbf{\Sigma}^{-1}\mathbf{X}. \tag{9.21}$$

The determinant of the matrix $\mathbf{\Sigma}$, due to its block structure, can easily be derived as follows [43]:

$$\begin{aligned}\det(\mathbf{\Sigma}) \;=\; & \prod_{k'=0}^{K-1}\left[\alpha_1^2 P_s(\omega_{k'}) + P_{b_1}(\omega_{k'})\right] \cdot \\ & \prod_{k'=0}^{K-1}\left[P_{b_2}(\omega_{k'}) + \frac{\alpha_1^2 P_s(\omega_{k'})P_{b_1}(\omega_{k'})}{\alpha_1^2 P_s(\omega_{k'}) + P_{b_1}(\omega_{k'})}\right],\end{aligned} \tag{9.22}$$

which is independent of D_1 and D_2. The inverse of $\mathbf{\Sigma}$ can be obtained as:

$$\mathbf{\Sigma}^{-\mathbf{1}} \stackrel{\triangle}{=} \mathbf{A} = \begin{bmatrix}\mathbf{A}_{11} & \mathbf{A}_{12}\\ \mathbf{A}_{12}^H & \mathbf{A}_{22}\end{bmatrix}, \tag{9.23}$$

where $\mathbf{A}_{11}$, $\mathbf{A}_{22}$ and $\mathbf{A}_{12}$ are three diagonal matrices of size $K \times K$:

$$\begin{aligned}\mathbf{A}_{11} \;=\; \mathrm{diag}\Bigg[& \left(P_{b_1}(\omega_0) + \frac{\alpha_1^2 P_s(\omega_0)P_{b_2}(\omega_0)}{\alpha_2^2 P_s(\omega_0) + P_{b_2}(\omega_0)}\right)^{-1}, \quad \ldots, \\ & \left(P_{b_1}(\omega_{K-1}) + \frac{\alpha_1^2 P_s(\omega_{K-1})P_{b_2}(\omega_{K-1})}{\alpha_2^2 P_s(\omega_{K-1}) + P_{b_2}(\omega_{K-1})}\right)^{-1}\Bigg],\end{aligned}$$

$$\begin{aligned}\mathbf{A}_{22} \;=\; \mathrm{diag}\Bigg[& \left(P_{b_2}(\omega_0) + \frac{\alpha_2^2 P_s(\omega_0)P_{b_1}(\omega_0)}{\alpha_1^2 P_s(\omega_0) + P_{b_1}(\omega_0)}\right)^{-1}, \quad \ldots, \\ & \left(P_{b_2}(\omega_{K-1}) + \frac{\alpha_2^2 P_s(\omega_{K-1})P_{b_1}(\omega_{K-1})}{\alpha_1^2 P_s(\omega_{K-1}) + P_{b_1}(\omega_{K-1})}\right)^{-1}\Bigg],\end{aligned}$$

and

$$\begin{aligned}\mathbf{A}_{12} \;=\; \mathrm{diag}\Bigg[& -\frac{\alpha_1\alpha_2 P_s(\omega_0)e^{j\omega_0\tau_{12}}}{\alpha_1^2 P_s(\omega_0)P_{b_2}(\omega_0) + P_{b_1}(\omega_0)P_{b_2}(\omega_0)}, \quad \ldots, \\ & -\frac{\alpha_1\alpha_2 P_s(\omega_{K-1})e^{j\omega_{K-1}\tau_{12}}}{\alpha_1^2 P_s(\omega_{K-1})P_{b_2}(\omega_{K-1}) + P_{b_1}(\omega_{K-1})P_{b_2}(\omega_{K-1})}\Bigg].\end{aligned}$$

Substituting the determinant and the inverse matrix of $\mathbf{\Sigma}$ into (9.21) yields

$$\mathcal{L} = \mathcal{L}_1 + \mathcal{L}_2, \tag{9.24}$$

where

$$\begin{aligned}
\mathcal{L}_1 \ = \ & -\frac{K}{2}\ln(2\pi) - \frac{1}{2}\sum_{k'=0}^{K-1} \ln[\alpha_1^2 P_s(\omega_{k'}) + P_{b_1}(\omega_{k'})] \\
& -\frac{1}{2}\sum_{k'=0}^{K-1} \ln\left[P_{b_2} + \frac{\alpha_1^2 P_s(\omega_{k'})P_{b_1}(\omega_{k'})}{\alpha_1^2 P_s(\omega_{k'}) + P_{b_1}(\omega_{k'})}\right] \\
& -\frac{1}{2}\sum_{k'=0}^{K-1} \left[P_{b_1}(\omega_{k'}) + \frac{\alpha_1^2 P_s(\omega_{k'})P_{b_2}(\omega_{k'})}{\alpha_2^2 P_s(\omega_{k'}) + P_{b_2}(\omega_{k'})}\right]^{-1} X_1^*(\omega_{k'})X_1(\omega_{k'}) \\
& -\frac{1}{2}\sum_{k'=0}^{K-1} \left[P_{b_2}(\omega_{k'}) + \frac{\alpha_2^2 P_s(\omega_{k'})P_{b_1}(\omega_{k'})}{\alpha_1^2 P_s(\omega_{k'}) + P_{b_1}(\omega_{k'})}\right]^{-1} X_2^*(\omega_{k'})X_2(\omega_{k'}),
\end{aligned}$$

and

$$\mathcal{L}_2 = \sum_{k'=0}^{K-1} \left[\frac{\alpha_1\alpha_2 P_s(\omega_{k'})e^{j\omega_{k'}\tau_{12}}}{\alpha_1^2 P_s(\omega_{k'})P_{b_2}(\omega_{k'}) + P_{b_1}(\omega_{k'})P_{b_2}(\omega_{k'})}\right] X_1(\omega_{k'})X_2^*(\omega_{k'}).$$

Apparently, $\mathcal{L}_1$ is independent of τ_{12} and $\mathcal{L}_2$ is a function of τ_{12}. Therefore, maximizing the log likelihood function $\mathcal{L}$ with respect to τ_{12} is equal to selecting a τ_{12} that maximizes $\mathcal{L}_2$. In other words, the ML estimator for time delay is

$$\begin{aligned}
\hat{\tau}_{12} \ &= \ \arg\max_p \hat{r}_{x_1x_2}^{\mathrm{ML}}(p) \\
&= \ \arg\max_p \sum_{k'=0}^{K-1} \frac{\alpha_1\alpha_2 P_s(\omega_{k'})X_1(\omega_{k'})X_2^*(\omega_{k'})e^{j\omega_{k'}p}}{\alpha_1^2 P_s(\omega_{k'})P_{b_2}(\omega_{k'}) + P_{b_1}(\omega_{k'})P_{b_2}(\omega_{k'})}.
\end{aligned} \tag{9.25}$$

It can be seen from (9.25) that in order to achieve ML estimation, the attenuation factors α_1 and α_2 have to be known *a priori*. In addition, we need to know the power spectral densities of both the speech and noise signals (note that given the observation data, if the power spectral densities of the noise signals are known, we can easily estimate the power spectral density of the speech signal, and vice versa).

The ML method for TDE is summarized in Table 9.3.

9.5 Generalized Cross-Correlation Method

Comparing (9.25) with (9.10), we can see that the ML estimate is achieved by weighting the cross spectrum between sensors' outputs. As a matter of fact,

Table 9.3. The maximum likelihood method for time delay estimation.

```
Parameter: $\hat{\tau}_{12}$
Estimation: For $t = 0, 1, \cdots$
    (a) Obtain a frame of observation signals at time instant $t$:
        $\{x_n(t), x_n(t+1), \cdots, x_n(t+K-1)\}$, $n = 1, 2$
    (b) Estimate the spectrum of $x_n(t+k)$:
        $X_n(\omega_{k'}) = \text{FFT}\{x_n(t+k)\} = \sum_{k=0}^{K-1} x_n(t+k)e^{-j\omega_{k'}k}$
    (c) Estimate the cost function:
        $\hat{r}^{\text{ML}}_{x_1x_2}(p) = \sum_{k'=0}^{K-1} \frac{\alpha_1\alpha_2 P_s(\omega_{k'})X_1(\omega_{k'})X_2^*(\omega_{k'})e^{j\omega_{k'}p}}{\alpha_1^2 P_s(\omega_{k'})P_{b_2}(\omega_{k'}) + P_{b_1}(\omega_{k'})P_{b_2}(\omega_{k'})}$
        $= \text{IFFT}\left\{\frac{\alpha_1\alpha_2 P_s(\omega_{k'})X_1(\omega_{k'})X_2^*(\omega_{k'})}{\alpha_1^2 P_s(\omega_{k'})P_{b_2}(\omega_{k'}) + P_{b_1}(\omega_{k'})P_{b_2}(\omega_{k'})}\right\}$
    (d) Obtain the time delay:
        $\hat{\tau}_{12} = \arg\max_p \hat{r}^{\text{ML}}_{x_1x_2}(p)$
```

such a weighting processing has been found effective in improving TDE performance. The resulting technique is known as the generalized cross-correlation (GCC) method, which has gained its great popularity since the informative paper [205] was published by Knapp and Carter in 1976. With the signal model given in (9.1), the GCC framework estimates TDOA between $x_1(k)$ and $x_2(k)$ as:

$$\hat{\tau}_{12} = \arg\max_p \hat{r}^{\text{GCC}}_{x_1x_2}(p) \tag{9.26}$$

where

$$\begin{aligned}\hat{r}^{\text{GCC}}_{x_1x_2}(p) &= \sum_{k'=0}^{K-1} \Phi(\omega_{k'})G_{x_1x_2}(\omega_{k'})e^{j\omega_{k'}p} \\ &= \sum_{k'=0}^{K-1} \varsigma_{x_1x_2}(\omega_{k'})e^{j\omega_{k'}p}\end{aligned} \tag{9.27}$$

is the generalized cross-correlation function (GCCF),

$$G_{x_1x_2}(\omega_{k'}) = E[X_1(\omega_{k'})X_2^*(\omega_{k'})]$$

is the cross spectrum between $x_1(k)$ and $x_2(k)$, $\Phi(\omega_{k'})$ is a weighting function (sometimes called a *prefilter*), and

$$\varsigma_{x_1x_2}(\omega_{k'}) = \Phi(\omega_{k'})G_{x_1x_2}(\omega_{k'}) \tag{9.28}$$

is the weighted cross spectrum. In a practical system, the cross spectrum $G_{x_1x_2}(\omega_{k'})$ has to be estimated, which is normally achieved by replacing the expected value by its instantaneous value, i.e., $\hat{G}_{x_1x_2}(\omega_{k'}) = X_1(\omega_{k'})X_2^*(\omega_{k'})$.

Table 9.4. Commonly used weighting functions in the GCC method.

Method name	**Weighting function $\Phi(\omega_{k'})$**
Cross correlation	1
Phase transform (PHAT)	$\frac{1}{\lvert G_{x_1x_2}(\omega_{k'})\rvert}$
Smoothed coherence transform (SCOT)	$\frac{1}{\sqrt{P_{x_1}(\omega_{k'})P_{x_1}(\omega_{k'})}}$
Eckart	$\frac{P_s(\omega_{k'})}{P_{b_1}(\omega_{k'})P_{b_2}(\omega_{k'})}$
Maximum likelihood (ML)	$\frac{\alpha_1\alpha_2 P_s(\omega_{k'})}{\alpha_1^2 P_s(\omega_{k'})P_{b_2}(\omega_{k'}) + P_{b_1}(\omega_{k'})P_{b_2}(\omega_{k'})}$

Table 9.5. The generalized cross-correlation method for time delay estimation.

Parameter: $\hat{\tau}_{12}$

Estimation: Select a weighting function $\Phi(\omega_{k'})$

For $t = 0, 1, \cdots$

(a) Obtain a frame of observation signals at time instant t:

$\{x_n(t), x_n(t+1), \cdots, x_n(t+K-1)\}$, $n = 1, 2$

(b) Estimate the spectrum of $x_n(t+k)$:

$$X_n(\omega_{k'}) = \text{FFT}\{x_n(t+k)\} = \sum_{k=0}^{K-1} x_n(t+k)e^{-j\omega_{k'}k}$$

(c) Compute $\Phi(\omega_{k'})$

(d) Estimate the generalized cross-correlation function:

$$\begin{aligned}\hat{r}^{\text{GCC}}_{x_1x_2}(p) &= \text{IFFT}\{\Phi(\omega_{k'})X_1(\omega_{k'})X_2^*(\omega_{k'})\} \\ &= \frac{1}{K}\sum_{k'=0}^{K-1} \Phi(\omega_{k'})X_1(\omega_{k'})X_2^*(\omega_{k'})e^{j\omega_{k'}p}\end{aligned}$$

(d) Obtain the time delay:

$\hat{\tau}_{12} = \arg\max_p \hat{r}^{\text{GCC}}_{x_1x_2}(p)$

There are a number of member algorithms in the GCC family depending on how the weighting function $\Phi(\omega_{k'})$ is selected [205], [66], [261], [157], [65]. Commonly used weighting functions for single-path propagation environments are summarized in Table 9.4. Combination of some of these functions was also suggested [308].

Different weighting functions possess different properties. Some can make the delay estimates more immune to additive noise, while others can improve the robustness of TDE against the multipath effect. As a result, weighting functions should be selected according to the specific application requirements and the corresponding environmental conditions.

The GCC method for time delay estimation is summarized in Table 9.5.

9.6 Adaptive Eigenvalue Decomposition Algorithm

All the algorithms outlined in the previous sections basically achieve TDOA estimate by identifying the extremum of the (generalized) cross correlation function between two channels. For an ideal acoustic system where only attenuation and delay are taken into account, an impulse would appear at the actual TDOA. In practical room acoustic environments, however, additive background noise and room reverberation make the peak no longer well-defined or even no longer dominate in the estimated cross-correlation function. Many amendments to the correlation based algorithms have been proposed. But they are still unable to deal well with reverberation.

Recently, an adaptive eigenvalue decomposition (AED) algorithm was proposed to deal with the TDE problem in room reverberant environments [32]. Unlike the cross correlation based methods, this algorithm assumes a reverberation (multipath) signal model. It first identifies the channel impulse responses from the source to the two sensors. The delay estimate is then determined by finding the direct paths from the two measured impulse responses.

For the signal model given in (9.3) with two sensors, in the absence of noise, one can easily check that:

$$x_1(k) * h_2 = s(k) * h_1 * h_2 = x_2(k) * h_1. \tag{9.29}$$

At time instant k, this relation can be rewritten in a vector-matrix form as:

$$\mathbf{x}^T(k)\mathbf{u} = \mathbf{x}_1^T(k)\mathbf{h}_2 - \mathbf{x}_2^T(k)\mathbf{h}_1 = 0, \tag{9.30}$$

where

$$\begin{aligned}
\mathbf{x}_n(k) &= \begin{bmatrix} x_n(k) & x_n(k-1) & \cdots & x_n(k-L+1) \end{bmatrix}^T, \ n = 1, 2, \\
\mathbf{x}(k) &= \begin{bmatrix} \mathbf{x}_1^T(k) & \mathbf{x}_2^T(k) \end{bmatrix}^T, \\
\mathbf{u} &= \begin{bmatrix} \mathbf{h}_2^T & -\mathbf{h}_1^T \end{bmatrix}^T.
\end{aligned}$$

Left multiplying (9.30) by $\mathbf{x}(k)$ and taking expectation yields

$$\mathbf{R}_x\mathbf{u} = \mathbf{0}, \tag{9.31}$$

where $\mathbf{R}_x = E\left[\mathbf{x}(k)\mathbf{x}^T(k)\right]$ is the covariance matrix of the two observation sensor signals. This implies that the vector $\mathbf{u}$, which consists of two impulse responses, is in the null space of $\mathbf{R}_x$. More specifically, $\mathbf{u}$ is the eigenvector of $\mathbf{R}_x$ corresponding to the eigenvalue 0. It has been shown that the two channel impulse responses (i.e., $\mathbf{h}_1$ and $\mathbf{h}_2$) can be uniquely determined (up to a scale and a common delay) from (9.31) if the following two conditions are satisfied (see Sect. 6.2):

- The polynomials formed from $\mathbf{h}_1$ and $\mathbf{h}_2$ (i.e., the z-transforms of $\mathbf{h}_1$ and $\mathbf{h}_2$) are co-prime, or they do not share any common zeros;

Table 9.6. The adaptive eigenvalue decomposition algorithm for time delay estimation.

```
Parameters:   $\hat{\tau}_{12}$, $\hat{\mathbf{h}}_1$, $\hat{\mathbf{h}}_2$
Estimation:   Initialize $\hat{\mathbf{h}}_1$, $\hat{\mathbf{h}}_2$
              For $k = 0, 1, \cdots$
                (a) Construct the signal vector:
                    $\mathbf{x}_1(k) = \left[ \begin{array}{cccc} x_1(k) & x_1(k-1) & \cdots & x_1(k-L+1) \end{array} \right]^T$
                    $\mathbf{x}_2(k) = \left[ \begin{array}{cccc} x_2(k) & x_2(k-1) & \cdots & x_2(k-L+1) \end{array} \right]^T$
                    $\mathbf{x}(k) = \left[ \begin{array}{cc} \mathbf{x}_1^T(k) & \mathbf{x}_2^T(k) \end{array} \right]^T$
                (b) Computer the error signal:
                    $e(k) = \hat{\mathbf{u}}^T(k)\mathbf{x}(k)$
                (c) Update the filter coefficients:
                    $\hat{\mathbf{u}}(k+1) = \frac{\hat{\mathbf{u}}(k) - \mu e(k)\mathbf{x}(k)}{||\hat{\mathbf{u}}(k) - \mu e(k)\mathbf{x}(k)||}$
                (d) Obtain the time delay (after convergence):
                    $\hat{\tau}_{12} = \arg\max_l |\hat{h}_{2,l}| - \arg\max_l |\hat{h}_{1,l}|$
```

- The covariance matrix of the source signal $s(k)$, i.e., $\mathbf{R}_s = E\left[\mathbf{s}(k)\mathbf{s}^T(k)\right]$, is of full rank.

When an independent white noise signal is present at each sensor, it will regularize the covariance matrix; as a consequence, the covariance matrix $\mathbf{R}_x$ does not have a zero eigenvalue anymore. In such a case, an estimate of the impulse responses can be achieved through the following algorithm, which is an adaptive way to find the eigenvector associated with the smallest eigenvalue of $\mathbf{R}_x$:

$$\hat{\mathbf{u}}(k+1) = \frac{\hat{\mathbf{u}}(k) - \mu e(k)\mathbf{x}(k)}{||\hat{\mathbf{u}}(k) - \mu e(k)\mathbf{x}(k)||}, \tag{9.32}$$

with the constraint that $||\hat{\mathbf{u}}(k)|| = 1$, where

$$e(k) = \hat{\mathbf{u}}^T(k)\mathbf{x}(k) \tag{9.33}$$

is an error signal, and μ is the adaptation step.

With the identified impulse responses $\hat{\mathbf{h}}_1$ and $\hat{\mathbf{h}}_2$, the time delay estimate is determined as the time difference between the two direct paths, i.e.,

$$\hat{\tau}_{12} = \arg\max_l |\hat{h}_{2,l}| - \arg\max_l |\hat{h}_{1,l}|. \tag{9.34}$$

The AED algorithm for time delay estimation is summarized in Table 9.6.

9.7 Multichannel Cross-Correlation Algorithm

Time delay estimation would be an easy task if the observation signals were merely delayed and attenuated copies of the source signal. In acoustical environments, however, the source signal is generally immersed in ambient noise and reverberation, making the TDE problem more complicated. In the previous section, we discussed an adaptive eigenvalue decomposition algorithm, which is formulated based on a multipath signal model and can deal well with reverberation. Another way to better cope with noise and reverberation is through the use of multiple (more than two) sensors and taking advantage of the redundant information. To illustrate the redundancy provided by multiple microphone sensors, let us consider a three-sensor linear array, which can be partitioned into three sensor pairs. Three delay measurements can then be acquired with the observation data, i.e., τ_{12} (TDOA between sensor 1 and sensor 2), τ_{23} (TDOA between sensor 2 and sensor 3), and τ_{13} (TDOA between sensor 1 and sensor 3). Apparently, these three delays are not independent. As a matter of fact, if the source is located in the far field, it is easily seen that $\tau_{13} = \tau_{12} + \tau_{23}$. Such a relation was exploited in [203] to formulate a two-stage TDE algorithm. In the preprocessing stage, three delay measurements were measured independently using the GCC method. A state equation was then formed and the Kalman filter is used in the post-processing stage to enhance the delay estimate of τ_{12}. By doing so, the estimation variance of τ_{12} can be significantly reduced. Recently, several approaches based on multiple sensor pairs were developed to deal with TDE in room acoustic environments [240], [149], [97]. Different from the Kalman filter method, these approaches fuse the estimated cost functions from multiple sensor pairs before searching for the time delay. In what follows, we discuss a multichannel cross-correlation (MCC) algorithm, which is derived from the spatial linear prediction and interpolation techniques. It can take advantage of the redundancy among multiple sensors in a more natural and coherent way than both the two-stage and fusion methods.

9.7.1 Forward Spatial Linear Prediction

Suppose that we have a microphone array consisting of N sensors positioned in a specific geometric configuration. The sensors' outputs [i.e., $x_1(k), x_2(k), \cdots, x_N(k)$] relate to the source signal through the single-path propagation model given in (9.1). We now seek to predict the signal at the first sensor from spatial samples from all the other $N-1$ sensor signals. For the ease of presentation, let us express the TDOA between the first and the nth sensors in terms of τ_{12} (the TDOA between the first and the second sensors) as:

$$\tau_{1n} = D_n - D_1 = f_n(\tau_{12}), \qquad n = 1, 2, \cdots, N, \tag{9.35}$$

where $f_1(\tau_{12}) = 0$, $f_2(\tau_{12}) = \tau_{12}$ and for $n > 2$, the function f_n depends not only on τ_{12}, but also on the array geometry. Note that in the near field, the function may also be contingent on the source position. We here consider the far-field case.

Using (9.35), we can write the signal model in (9.1) as

$$x_n(k) = \alpha_n s[k - D_1 - f_n(\tau_{12})] + b_n(k). \tag{9.36}$$

It is clear then that the signal $x_1[k - f_N(\tau_{12})]$ is aligned with the signals $x_n[k - f_N(\tau_{12}) + f_n(\tau_{12})]$, $n = 2, 3, \cdots, N$. From these observations, we define the following forward spatial prediction error signal:

$$e_1(k, q) = x_1[k - f_N(q)] - \mathbf{x}_{2:N}^T[k - f_N(q)]\mathbf{a}_q, \tag{9.37}$$

where q is a guessed relative delay for τ_{12}, and

$$\begin{aligned}\mathbf{x}_{2:N}[k - f_N(q)] \;=\; \Big[\, & x_2[k - f_N(q) + f_2(q)] \;\; x_3[k - f_N(q) + f_3(q)] \\ & \cdots \;\; x_N(k) \,\Big]^T,\end{aligned}$$

and

$$\mathbf{a}_q = \begin{bmatrix} a_{q,2} & a_{q,3} & \cdots & a_{q,N} \end{bmatrix}^T$$

is the forward spatial linear predictor. Consider the criterion:

$$J_{q,1} = E\left\{e_1^2(k, q)\right\}. \tag{9.38}$$

Minimization of (9.38) leads to the Wiener-Hopf equations:

$$\mathbf{R}_{q,2:N}\mathbf{a}_q = \mathbf{r}_{q,2:N}, \tag{9.39}$$

where

$$\begin{aligned}\mathbf{R}_{q,2:N} &= E\{\mathbf{x}_{2:N}[k - f_N(q)]\mathbf{x}_{2:N}^T[k - f_N(q)]\} \\ &= \begin{bmatrix} E\{x_2^2(k)\} & r_{x_2x_3}(q) & \cdots & r_{x_2x_N}(q) \\ r_{x_3x_2}(q) & E\{x_3^2(k)\} & \cdots & r_{x_3x_N}(q) \\ \vdots & \vdots & \ddots & \vdots \\ r_{x_Nx_2}(q) & r_{x_Nx_3}(q) & \cdots & E\{x_N^2(k)\} \end{bmatrix}\end{aligned}$$

is the spatial correlation matrix with

$$\begin{aligned} r_{x_ix_j}(q) &= E\{x_i[k - f_N(q) + f_i(q)]x_j[k - f_N(q) + f_j(q)]\} \\ &= E\{x_i[k + f_i(q)]x_j[k + f_j(q)]\} \\ &= E\{x_i[k - f_j(q)]x_j[k - f_i(q)]\}, \end{aligned} \tag{9.40}$$

and

$$\begin{aligned}\mathbf{r}_{q,2:N} &= E\left\{\mathbf{x}_{2:N}[k-f_N(q)]x_1[k-f_N(q)]\right\}\\ &= \begin{bmatrix} E\{x_2[k-f_N(q)+f_2(q)]x_1[k-f_N(q)]\}\\ E\{x_3[k-f_N(q)+f_3(q)]x_1[k-f_N(q)]\}\\ \vdots\\ E\{x_N[k-f_N(q)+f_N(q)]x_1[k-f_N(q)]\}\end{bmatrix}\\ &= \begin{bmatrix} E\{x_2(k)x_1[k-f_2(q)]\}\\ E\{x_3(k)x_1[k-f_3(q)]\}\\ \vdots\\ E\{x_N(k)x_1[k-f_N(q)]\}\end{bmatrix}\end{aligned}$$

is the spatial correlation vector. Note that the spatial correlation matrix is not Toeplitz in general, except in some particular cases.

For $q=\tau_{12}$ and for the noise-free case (where $b_n(k)=0$, $n=1,2,\cdots,N$), it can easily be checked that with the signal model given in (9.36), the rank of matrix $\mathbf{R}_{q,2:N}$ is equal to 1. This means that the samples $x_1(k)$ can be perfectly predicted from any one of the other microphone samples. However, the noise is never absent in practice and is in general isotropic. The noise energy at different microphones is added to the main diagonal of the correlation matrix $\mathbf{R}_{q,2:N}$, which will regularize it and make it positive definite (which we suppose in the rest of this chapter). A unique solution to (9.39) is then guaranteed for any number of microphones. This solution is optimal from a Wiener theory point of view as explained in Chap. 3.

9.7.2 Backward Spatial Linear Prediction

Now we consider microphone N and we would like to align successive time samples of this microphone signal with spatial samples from the $N-1$ other microphone signals. It is clear that $x_N(k)$ is aligned with the signals $x_n(k-f_N(\tau_{12})+f_n(\tau_{12})]$, $n=1,2,\cdots,N-1$. From these observations, we define the following backward spatial prediction error signal:

$$e_N(k,q)=x_N(k)-\mathbf{x}_{1:N-1}^T[k-f_N(q)]\mathbf{b}_q, \tag{9.41}$$

where

$$\begin{aligned}\mathbf{x}_{1:N-1}[k-f_N(q)] = \Big[\, & x_1[k-f_N(q)+f_1(q)] \;\; x_2[k-f_N(q)+f_2(q)]\\ & \cdots\;\; x_{N-1}[k-f_N(q)+f_{N-1}(q)]\,\Big]^T\end{aligned}$$

and

$$\mathbf{b}_q=\begin{bmatrix} b_{q,1} & b_{q,2} & \cdots & b_{q,N-1}\end{bmatrix}^T$$

is the backward spatial linear predictor. Minimization of the criterion:

$$J_{q,N} = E\left\{e_N^2(k,q)\right\} \tag{9.42}$$

leads to the Wiener-Hopf equations:

$$\mathbf{R}_{q,1:N-1}\mathbf{b}_q = \mathbf{r}_{q,1:N-1}, \tag{9.43}$$

where

$$\begin{aligned}\mathbf{R}_{q,1:N-1} &= E\{\mathbf{x}_{1:N-1}[k-f_N(q)]\mathbf{x}_{1:N-1}^T[k-f_N(q)]\} \\ &= \begin{bmatrix} E\{x_1^2(k)\} & r_{x_1x_2}(q) & \cdots & r_{x_1x_{N-1}}(q) \\ r_{x_2x_1}(q) & E\{x_2^2(k)\} & \cdots & r_{x_2x_{N-1}}(q) \\ \vdots & \vdots & \ddots & \vdots \\ r_{x_{N-1}x_1}(q) & r_{x_{N-1}x_2}(q) & \cdots & E\{x_{N-1}^2(k)\} \end{bmatrix}\end{aligned}$$

and

$$\begin{aligned}\mathbf{r}_{q,1:N-1} &= E\{\mathbf{x}_{1:N-1}[k-f_N(q)]x_N(k)\} \\ &= \begin{bmatrix} E\{x_1(k)x_N[k+f_N(q)-f_1(q)]\} \\ E\{x_2(k)x_N[k+f_N(q)-f_2(q)]\} \\ \vdots \\ E\{x_{N-1}(k)x_N[k+f_N(q)-f_{N-1}(q)]\} \end{bmatrix}.\end{aligned}$$

9.7.3 Spatial Linear Interpolation

The ideas presented for forward/backward spatial linear prediction can easily be extended to spatial linear interpolation, where we seek to determine how any one of these signals can be interpolated from the others. For the nth sensor, the spatial interpolation error signal is defined as

$$e_n(k,q) = -\mathbf{x}_{1:N}^T[k-f_L(q)]\mathbf{c}_{q,n}, \tag{9.44}$$

where

$$\mathbf{x}_{1:N}[k-f_N(q)] = [x_1[k-f_N(q)+f_1(q)]\ x_2[k-f_N(q)+f_2(q)]\ \cdots\ x_N(n)]^T$$

and

$$\mathbf{c}_{q,n} = \begin{bmatrix} c_{q,n,1} & c_{q,n,2} & \cdots & c_{q,n,N} \end{bmatrix}^T$$

with the constraint $c_{q,n,n} = -1$. This $\mathbf{c}_{q,n}$ vector without the component $c_{q,n,n}$ is the nth interpolator. The criterion associated with (9.44) is:

$$J_{q,n} = E\left\{e_n^2(k,q)\right\}. \tag{9.45}$$

By using a Lagrange multiplier, it is easy to see that the solution to this optimization problem is:

$$\mathbf{R}_{q,1:N}\mathbf{c}_{q,n} = -\mathbf{c}_{q,n}^T\mathbf{R}_{q,1:N}\mathbf{c}_{q,n}\mathbf{u}_n, \tag{9.46}$$

where

$$\mathbf{R}_{q,1:N} = E\{\mathbf{x}_{1:N}[k - f_N(q)]\mathbf{x}_{1:N}^T[k - f_N(q)]\}$$

is the spatial correlation matrix, and

$$\mathbf{u}_n = [\, 0 \ \ldots \ 0 \ 1 \ 0 \ \ldots \ 0 \,]^T$$

is a vector of length N with its nth component equal to unity and all others zero.

9.7.4 Time Delay Estimation Using Spatial Linear Prediction

The spatial linear prediction, and more generally the spatial interpolation technique can be applied to the problem of TDE. Here, we give an example of using the forward spatial linear prediction. The idea can be easily generalized to the backward spatial linear prediction and the spatial interpolation.

Let $J_{q,1;\min}$ denote the minimum mean-squared error, for the value q, defined by

$$J_{q,1;\min} \stackrel{\triangle}{=} E\left\{e_{1;\min}^2(k,q)\right\}. \tag{9.47}$$

If we replace $\mathbf{a}_q$ by $\mathbf{R}_{q,2:N}^{-1}\mathbf{r}_{q,2:N}$ in (9.37), we get:

$$e_{1;\min}(k,q) = x_1[k - f_N(q)] - \mathbf{x}_{2:N}^T[k - f_N(q)]\mathbf{R}_{q,2:N}^{-1}\mathbf{r}_{q,2:N}. \tag{9.48}$$

It follows immediately that

$$J_{q,1;\min} = E\left\{x_1^2[k - f_N(q)]\right\} - \mathbf{r}_{q,2:N}^T\mathbf{R}_{q,2:N}^{-1}\mathbf{r}_{q,2:N}. \tag{9.49}$$

The value of q that gives the minimum $J_{q,1;\min}$ corresponds to the TDOA between the first and the second sensors, i.e., τ_{12}. Mathematically, the solution to the TDE problem is given by

$$\hat{\tau}_{12} = \arg\min_q J_{q,1;\min}. \tag{9.50}$$

Particular case: Two microphones ($N = 2$). In this case, it can be checked that the solution is:

$$\begin{aligned}\hat{\tau}_{12} &= \arg\min_q \left\{E\{x_1^2(k-q)\}\left[1 - \frac{E^2\{x_1(k-q)x_2(k)\}}{E\{x_1^2(k-q)\}E\{x_2^2(k)\}}\right]\right\} \\ &= \arg\min_q \left\{E\{x_1^2(k-q)\}\left[1 - \frac{r_{x_1x_2}^2(q)}{E\{x_1^2(k-q)\}E\{x_2^2(k)\}}\right]\right\}. \end{aligned} \tag{9.51}$$

Since $E\{x_1^2(k-q)\}$, which is the energy of $x_1(k)$, does not affect the peak position, (9.51) can be further written as

$$\begin{aligned}\hat{\tau}_{12} &= \arg\min_q \left\{E\{x_1^2(k-q)\}\left[1-\rho_{x_1x_2}^2(q)\right]\right\} \\ &= \arg\min_q \left\{1-\rho_{x_1x_2}^2(q)\right\} \\ &= \arg\max_q \rho_{x_1x_2}^2(q), \end{aligned} \tag{9.52}$$

where

$$\rho_{x_1x_2}(q) = \frac{r_{x_1x_2}(q)}{\sqrt{E\{x_1^2(k-q)\}E\{x_2^2(k)\}}} \tag{9.53}$$

is the cross-correlation coefficient between $x_1(k-q)$ and $x_2(k)$. Apparently, this result is similar to what is obtained with the cross-correlation method. Note that in the general case with multiple microphone sensors, this approach can be viewed as an extension of the cross-correlation method from the two-channel to the multichannel cases, which can take advantage of the knowledge of the microphone array to estimate time delay between the first and the second sensors optimally in a least-mean-square sense.

9.7.5 Spatial Correlation Matrix and Its Properties

Consider the N microphone signals $x_n,\ n=1,2,\cdots,N$. The corresponding spatial correlation matrix is

$$\begin{aligned}\mathbf{R}(q) &= \mathbf{R}_{q,1:N} = E\{\mathbf{x}_{1:N}[k-f_N(q)]\mathbf{x}_{1:N}^T[k-f_N(q)]\} \\ &= \begin{bmatrix} r_{x_1x_1}(q) & r_{x_1x_2}(q) & \cdots & r_{x_1x_N}(q) \\ r_{x_2x_1}(q) & r_{x_2x_2}(q) & \cdots & r_{x_2x_N}(q) \\ \vdots & \ddots & \ddots & \vdots \\ r_{x_Nx_1}(q) & r_{x_Nx_2}(q) & \cdots & r_{x_Nx_N}(q) \end{bmatrix}, \end{aligned} \tag{9.54}$$

which can be factored as:

$$\mathbf{R}(q) = \mathbf{E}\widetilde{\mathbf{R}}(q)\mathbf{E}, \tag{9.55}$$

where

$$\mathbf{E} = \begin{bmatrix} \sqrt{E[x_0^2(k)]} & 0 & \cdots & 0 \\ 0 & \sqrt{E[x_1^2(k)]} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \sqrt{E[x_N^2(k)]} \end{bmatrix} \tag{9.56}$$

is a diagonal matrix,

$$\widetilde{\mathbf{R}}(q) = \begin{bmatrix} 1 & \rho_{x_1x_2}(q) & \cdots & \rho_{x_1x_N}(q) \\ \rho_{x_1x_2}(q) & 1 & \cdots & \rho_{x_2x_N}(q) \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{x_1x_N}(q) & \cdots & \rho_{x_{N-1}x_N}(q) & 1 \end{bmatrix} \tag{9.57}$$

is a symmetric matrix, and

$$\rho_{x_i x_j}(q) = \frac{E\{x_i[k - f_j(q)]x_j[k - f_i(q)]\}}{\sqrt{E\{x_i^2(k)\}E\{x_j^2(k)\}}}, \quad i, j = 1, 2, \cdots, N, \tag{9.58}$$

is the cross-correlation coefficient between $x_i[k - f_j(q)]$ and $x_j[k - f_i(q)]$.

The $\widetilde{\mathbf{R}}(q)$ matrix given in (9.57) has the following properties:

$$\text{(a)} \qquad 0 < \det\left[\widetilde{\mathbf{R}}(q)\right] \leq 1, \tag{9.59}$$

$$\text{(b)} \qquad \det\left[\widetilde{\mathbf{R}}(q)\right] \leq \frac{J_{q,1;\min}}{E[x_1^2(k)]} \leq 1. \tag{9.60}$$

Proofs: Since $\mathbf{R}(q)$ is symmetric and is supposed to be positive definite, it is clear that $\det\left[\mathbf{R}(q)\right] > 0$, which implies that $\det\left[\widetilde{\mathbf{R}}(q)\right] > 0$. We can easily check that:

$$\det\left[\widetilde{\mathbf{R}}(q)\right] = \det\left(\widetilde{\mathbf{R}}_{q,1:N}\right) \leq \det\left(\widetilde{\mathbf{R}}_{q,2:N}\right) \leq \cdots \leq 1. \tag{9.61}$$

As a result, $0 < \det\left[\widetilde{\mathbf{R}}(q)\right] \leq 1$.

To show Property (b), let us define

$$\underline{\mathbf{a}}_q = \left[\begin{array}{cc} a_{q,1} & \mathbf{a}_q^T \end{array}\right]^T. \tag{9.62}$$

Then, for $a_{q,1} = -1$, the forward prediction error signal defined in (9.37) can be rewritten as

$$e_1(k, q) = -\mathbf{x}_{1:N}^T \underline{\mathbf{a}}_q. \tag{9.63}$$

Continuing, the criterion shown in (9.38) can be expressed as

$$J_{q,1} = E\left\{e_1^2(k, q)]\right\} + \kappa(\mathbf{u}_1^T \underline{\mathbf{a}}_q + 1), \tag{9.64}$$

where κ is a Lagrange multiplier introduced to force $a_{q,1}$ to have value -1. It is then easily shown that:

$$J_{q,1;\min} = \frac{1}{\mathbf{u}_1^T \mathbf{R}^{-1}(q)\mathbf{u}_1}. \tag{9.65}$$

In this case, with (9.55), (9.65) becomes:

$$\begin{aligned} J_{q,1;\min} &= \frac{E\{x_1^2(k)\}}{\mathbf{u}_1^T \widetilde{\mathbf{R}}^{-1}(q)\mathbf{u}_1} \\ &= E\{x_1^2(k)\} \frac{\det\left[\widetilde{\mathbf{R}}(q)\right]}{\det\left(\widetilde{\mathbf{R}}_{q,2:N}\right)}. \end{aligned} \tag{9.66}$$

Using (9.61), it is clear that Property (b) is verified.

9.7.6 Multichannel Cross-Correlation Coefficient

From the previous analysis, we can see that the determinant of the spatial correlation matrix is related to the minimum mean-squared error and to the power of the signals. In the two-channel case, it is easy to see that the cross-correlation coefficient between the two signals $x_1(k)$ and $x_2(k)$ is linked to the determinant of the corresponding spatial correlation matrix:

$$\rho^2_{x_1 x_2}(q) = 1 - \det\left(\widetilde{\mathbf{R}}_{q,1:2}\right). \tag{9.67}$$

By analogy to the cross-correlation coefficient definition between two random signals, we define the multichannel cross-correlation coefficient (MCCC) among the signals $x_n,\ n = 1, 2, \cdots, N$, as:

$$\begin{aligned} \rho^2_{1:N}(q) &= 1 - \det\left(\widetilde{\mathbf{R}}_{q,1:N}\right) \\ &= 1 - \det\left[\widetilde{\mathbf{R}}(q)\right]. \end{aligned} \tag{9.68}$$

Basically, the coefficient $\rho_{1:N}(q)$ will measure the amount of correlation among all the channels. This coefficient possesses the following properties:

(a) $0 \leq \rho^2_{1:N}(q) \leq 1$ (the case $\rho^2_{1:N}(q) = 1$ happens when matrix $\mathbf{R}(q)$ is nonnegative definite).
(b) If two or more signals are perfectly correlated, then $\rho^2_{1:N}(q) = 1$.
(c) If all the processes are completely uncorrelated with each other, then $\rho^2_{1:N}(q) = 0$.
(d) If one of the signals is completely uncorrelated with the $N - 1$ other signals, then this coefficient will measure the correlation among those $N - 1$ remaining signals.

The proof of these properties is straightforward. We shall leave it as an exercise for the reader.

9.7.7 Time Delay Estimation Using MCCC

Obviously, the multichannel cross-correlation coefficient $\rho^2_{1:N}(q)$ can be used for time delay estimation in the following way:

$$\begin{aligned} \hat{\tau}_{12} &= \arg\max_q \left(\rho^2_{1:N}(q)\right) \\ &= \arg\max_q \left\{1 - \det\left[\widetilde{\mathbf{R}}(q)\right]\right\} \\ &= \arg\min_q \det\left[\widetilde{\mathbf{R}}(q)\right]. \end{aligned} \tag{9.69}$$

It is clear that (9.69) is equivalent to (9.50). As we mentioned in our earlier statements, this algorithm can be treated as a generalization of the cross-correlation method from the two-channel to the multichannel cases. Figure 9.1

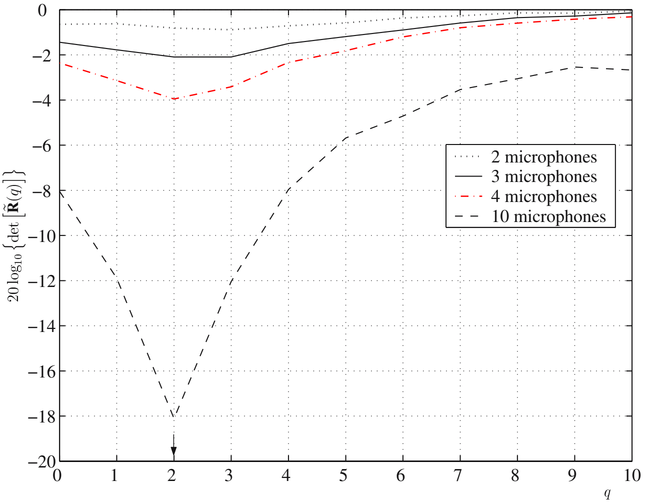


Fig. 9.1. Comparison of det $\left[\widetilde{\mathbf{R}}(q)\right]$ for different numbers of microphones.

shows an example where we use an equi-spaced linear array consisting of 10 microphone sensors. A loudspeaker is located in the far field, playing back a speech signal recorded from a female speaker. The sensors' outputs are corrupted by white Gaussian noise with SNR $= -5$ dB. Signals are sampled at 16 kHz. The true TDOA between Sensors 1 and 2 is $\tau_{12} = 2$ (samples). When only two sensors are used, we have $\hat{\tau}_{12} = 3$ (samples), which is a wrong TDOA estimate. When more than 3 sensors are used, we see that $\hat{\tau}_{12} = 2$ (samples), which is the same as the true TDOA. It can be seen that as the number of microphones increases, the valley of the cost function is better defined, which will enable an easier search for the extremum. This example demonstrates the effectiveness of the MCC approach in taking advantage of the redundant information provided by multiple sensor to improve TDE against noise.

It is worth pointing out that a pre-filtering (or pre-whitening) process can be applied to the observation signals before computing the MCCC. In this case, this multichannel correlation algorithm can be viewed as a generalized version of the GCC method.

The multichannel correlation method for time delay estimation is summarized in Table 9.7.

9.8 Adaptive Multichannel Time Delay Estimation

In the previous section, we have shown that TDE in adverse environments can be improved by using multiple sensors and taking advantage of the redundancy. The more the number of sensors we use, the better the TDE performance will be. However, since the multichannel cross-correlation method

Table 9.7. The multichannel cross-correlation method for time delay estimation.

```
Parameter:   $\hat{\tau}_{12}$
Estimation:  For $t = 0, 1, \cdots$
               (a) Obtain a frame of observation signals at time instant $t$:
                   $\{x_n(t), x_n(t+1), \cdots, x_n(t+K-1)\}$, $n = 1, 2, \cdots, N$
               (b) Pre-filtering the observation signals if needed
               (c) For $q = -\tau_{\max}, -\tau_{\max}+1, \cdots, \tau_{\max}$
                   (1) Estimate the spatial correlation matrix
                       $\widetilde{\mathbf{R}}(q) = \begin{bmatrix} 1 & \rho_{x_1x_2}(q) & \cdots & \rho_{x_1x_N}(q) \\ \rho_{x_1x_2}(q) & 1 & \cdots & \rho_{x_2x_N}(q) \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{x_1x_N}(q) & \cdots & \rho_{x_{N-1}x_N}(q) & 1 \end{bmatrix}$
                   (2) Estimate the multichannel cross-correlation coefficient
                       $\rho^2_{1:N}(q) = 1 - \det\left[\widetilde{\mathbf{R}}(q)\right]$
               (d) Obtain the time delay
                   $\hat{\tau}_{12} = \arg\max_q \left[\rho^2_{1:N}(q)\right]$
```

is derived based on the single-path propagation model, it has a fundamental weakness in its ability to deal with reverberation. In order to achieve a reasonable TDE performance in heavily reverberant environments, we may have to use a large number of receivers, which will increase both the cost and the size of the system. In this section, we discuss another multichannel algorithm based on the reverberation signal model, which can be viewed as an extension of the AED algorithm (see Sect. 9.6) from the two-channel to multichannel cases.

As shown in (9.32), the AED algorithm basically obtains the estimates of the two channel impulse responses by minimizing the following error signal:

$$e(k+1) = \mathbf{x}_1^T(k+1)\hat{\mathbf{h}}_2(k) - \mathbf{x}_2^T(k+1)\hat{\mathbf{h}}_1(k). \quad (9.70)$$

In order for the channel impulse responses to be uniquely determined, it requires that the polynomials formed from $\mathbf{h}_1$ and $\mathbf{h}_2$ do not share any common zeros. In room acoustic environments, channel impulse responses are usually very long, particularly when reverberation is strong. As a consequence, it is very likely that there are some common zeros between the two channels. One way to circumvent this common-zero problem is to use multiple sensors (channels). In the same acoustical environment, it would be less likely for all channels to share a common zero.

By analogy to the error signal defined in the AED algorithm, we can define the error signal between the ith and jth channels at time $k+1$ as:

$$e_{ij}(k+1) = \mathbf{x}_i^T(k+1)\hat{\mathbf{h}}_j(k) - \mathbf{x}_j^T(k+1)\hat{\mathbf{h}}_j(k), \quad i, j = 1, 2, \cdots, N. \quad (9.71)$$

Table 9.8. The multichannel adaptive algorithm for time delay estimation.

```
Parameters:  $\hat{\tau}_{ij}$, $i, j = 1, 2, \cdots, N$ and $i \neq j$
             $\hat{\mathbf{h}}_n$, $n = 1, 2, \cdots, N$
Estimation:  Initialize $\hat{\mathbf{h}}_n$, $n = 1, 2, \ldots, N$
             For $k = 0, 1, \cdots$
                (a) Estimate $\hat{\mathbf{h}}_n(k)$ using any one of the blind SIMO
                    identification algorithms described in Chap. 6
                    such as the FNMCLMS algorithm
                (b) Obtain the time delays based on the estimated
                    channel impulse responses:
                    $\hat{\tau}_{ij} = \arg\max_l |\hat{h}_{j,l}| - \arg\max_l |\hat{h}_{i,l}|$
                    $i, j = 1, 2, \cdots, N$, and $i \neq j$
```

Assuming that these error signals are equally important, we now define a cost function as follows:

$$\chi(k+1) = \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} e_{ij}^2(k+1), \tag{9.72}$$

where we exclude the cases of $e_{ii}(k) = 0$ $(i = 1, 2, \cdots, N)$ and count the $e_{ij}(k) = -e_{ij}(k)$ pair only once. With this definition of multichannel error signal, it follows immediately that many adaptive algorithms described in Chap. 6 can be exploited to estimate the channel impulse responses $\mathbf{h}_1, \mathbf{h}_2, \cdots, \mathbf{h}_N$. The TDOA estimate between any two channels can then be determined as

$$\hat{\tau}_{ij} = \arg\max_l |\hat{h}_{j,l}| - \arg\max_l |\hat{h}_{i,l}|. \tag{9.73}$$

The adaptive multichannel algorithm for time delay estimation is summarized in Table 9.8.

9.9 Acoustic Source Localization

In acoustic environments, the source location information plays an important role for applications such as automatic camera tracking for video-conferencing and beamformer steering for suppressing noise and reverberation. Estimation of source location, which is often called source-localization problem, has been of considerable interest for decades. A common method to obtain the estimate of source location is based on TDOA measurements as described previously.

Given the array geometry and TDOA measurements, the source-localization problem can be formulated mathematically as follows. The array consists of N microphones located at positions

$$\boldsymbol{\gamma}_n \stackrel{\triangle}{=} \begin{bmatrix} x_n & y_n & z_n \end{bmatrix}^T, \ n = 1, \cdots, N, \tag{9.74}$$

in Cartesian coordinates (see Fig. 9.2). The first microphone ($n = 1$) is regarded as the reference and is placed at the origin of the coordinate system, i.e. $\boldsymbol{\gamma}_1 = [0, 0, 0]^T$. The acoustic source is located at $\boldsymbol{\gamma}_\mathrm{s} \stackrel{\triangle}{=} [x_\mathrm{s}, y_\mathrm{s}, z_\mathrm{s}]^T$. The distances from the origin to the nth microphone and the source are denoted by ζ_n and ζ_s, respectively, where

$$\zeta_n \stackrel{\triangle}{=} \|\boldsymbol{\gamma}_n\| = \sqrt{x_n^2 + y_n^2 + z_n^2}, \quad n = 1, \cdots, N, \tag{9.75}$$

$$\zeta_\mathrm{s} \stackrel{\triangle}{=} \|\boldsymbol{\gamma}_\mathrm{s}\| = \sqrt{x_\mathrm{s}^2 + y_\mathrm{s}^2 + z_\mathrm{s}^2}. \tag{9.76}$$

The distance between the source and the nth microphone is denoted by

$$\eta_n \stackrel{\triangle}{=} \|\boldsymbol{\gamma}_n - \boldsymbol{\gamma}_\mathrm{s}\| = \sqrt{(x_n - x_\mathrm{s})^2 + (y_n - y_\mathrm{s})^2 + (z_n - z_\mathrm{s})^2}\ . \tag{9.77}$$

The difference in the distances of microphones n and j from the source is given by

$$d_{nj} \stackrel{\triangle}{=} \eta_n - \eta_j, \quad n, j = 1, \cdots, N. \tag{9.78}$$

This difference is usually termed the *range difference*. It is proportional to the time difference of arrival τ_{nj}. If the speed of sound is c, then

$$d_{nj} = c \cdot \tau_{nj}. \tag{9.79}$$

The speed of sound (in m/s) can be estimated from the air temperature t_air (in degrees Celsius) according to the following approximate (first-order) formula,

$$c \approx 331 + 0.610 \cdot t_\mathrm{air}. \tag{9.80}$$

The localization problem is then to estimate $\boldsymbol{\gamma}_\mathrm{s}$ given the set of $\boldsymbol{\gamma}_n$ and τ_{nj}. Note that there are $N(N-1)/2$ distinct TDOA estimates τ_{nj}, which exclude the case $n = j$ and count the $\tau_{nj} = -\tau_{jn}$ pair only once. However, in the absence of noise, the space spanned by these TDOA estimates is $(N-1)$-dimensional. Any $N-1$ linearly independent TDOAs determine all of the others. In a noisy environment, the TDOA redundancy can be used to improve the accuracy of the source localization algorithms, but this would increase their computational complexity. For simplicity and also without loss of generality, we choose $\tau_{n1}, n = 2, \cdots, N$ as the basis for this $\mathbb{R}^{N-1}$ space.

9.10 Measurement Model and Cramèr-Rao Lower Bound

When the source localization problem is examined using estimation theory, the measurements of the range differences are modeled by:

$$d_{n1} = g_n(\boldsymbol{\gamma}_\mathrm{s}) + \epsilon_n, \ n = 2, \cdots, N, \tag{9.81}$$

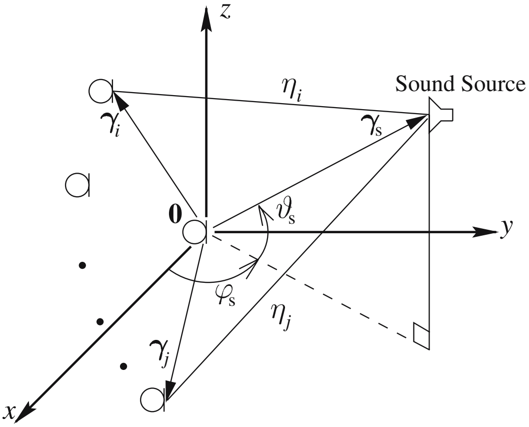


Fig. 9.2. Spatial diagram illustrating variables defined in the source-localization problem.

where

$$g_n(\boldsymbol{\gamma}_\mathrm{s}) = \|\boldsymbol{\gamma}_n - \boldsymbol{\gamma}_\mathrm{s}\| - \|\boldsymbol{\gamma}_\mathrm{s}\|,$$

and the ϵ_n's are measurement errors. In a vector form, such an additive measurement error model becomes,

$$\mathbf{d} = \mathbf{g}(\boldsymbol{\gamma}_\mathrm{s}) + \boldsymbol{\epsilon}, \tag{9.82}$$

where

$$\begin{aligned}
\mathbf{d} &= \begin{bmatrix} d_{21} & d_{31} & \cdots & d_{N1} \end{bmatrix}^T, \\
\mathbf{g}(\boldsymbol{\gamma}_\mathrm{s}) &= \begin{bmatrix} g_2(\boldsymbol{\gamma}_\mathrm{s}) & g_3(\boldsymbol{\gamma}_\mathrm{s}) & \cdots & g_N(\boldsymbol{\gamma}_\mathrm{s}) \end{bmatrix}^T, \\
\boldsymbol{\epsilon} &= \begin{bmatrix} \epsilon_2 & \epsilon_3 & \cdots & \epsilon_N \end{bmatrix}^T.
\end{aligned}$$

Further, we postulate that the additive measurement errors have mean zero and are independent of the range difference observation, as well as the source location $\boldsymbol{\gamma}_\mathrm{s}$. For a continuous-time estimator, the corrupting noise, as indicated in [154], is jointly Gaussian distributed. The probability density function (PDF) of $\mathbf{d}$ conditioned on $\boldsymbol{\gamma}_\mathrm{s}$ is subsequently given by,

$$p(\mathbf{d}|\boldsymbol{\gamma}_\mathrm{s}) = \frac{\exp\left\{-\frac{1}{2}\left[\mathbf{d} - \mathbf{g}(\boldsymbol{\gamma}_\mathrm{s})\right]^T \mathbf{C}_{\boldsymbol{\epsilon}}^{-1} \left[\mathbf{d} - \mathbf{g}(\boldsymbol{\gamma}_\mathrm{s})\right]\right\}}{\sqrt{(2\pi)^N \det(\mathbf{C}_{\boldsymbol{\epsilon}})}}, \tag{9.83}$$

where $\mathbf{C}_{\boldsymbol{\epsilon}}$ is the covariance matrix of $\boldsymbol{\epsilon}$. Note that $\mathbf{C}_{\boldsymbol{\epsilon}}$ is independent of $\boldsymbol{\gamma}_\mathrm{s}$ by assumption. Since digital equipment is used to sample the microphone waveforms and estimate the TDOAs, the error introduced by discrete-time processing also has to be taken into account. When this is done, the measurement error is no longer Gaussian and is more properly modeled as a mixture of a Gaussian noise and a noise that is uniformly distributed over $[-T_{\mathrm{s}}c/2, T_{\mathrm{s}}c/2]$, where T_{s} is the sampling period. As an example, for a digital source location estimator with an 8 KHz sampling rate operating at room temperature (25 degrees Celsius, i.e. $c \approx 346.25$ meters per second), the maximum error in range difference estimates due to sampling is about ±2.164 cm, which leads to considerable errors in the location estimate, especially when the source is far from the microphone array.

Under the measurement model (9.82), we are now faced with the parameter estimation problem of extracting the source location information from the mismeasured range differences or the equivalent TDOAs. For an *unbiased* estimator, a Cramèr-Rao lower bound (CRLB) can be placed on the variance of each estimated coordinate of the source location. However, since the range difference function $\mathbf{g}(\boldsymbol{\gamma}_{\mathrm{s}})$ in the measurement model is nonlinear in the parameters under estimation, it is very difficult (or even impossible) to find an unbiased estimator that is mathematically simple and attains the CRLB. The CRLB is usually used as a benchmark against which the statistical efficiency of any unbiased estimators can be compared.

In general, without any assumptions made about the PDF of the measurement error $\boldsymbol{\epsilon}$, the CRLB of the ith ($i = 1, 2, 3$) parameter variance is found as the $[i, i]$ element of the inverse of the Fisher information matrix defined by [200]:

$$[\mathbf{I}(\boldsymbol{\gamma}_{\mathrm{s}})]_{ij} \triangleq -E\left[\frac{\partial^2 \ln p(\mathbf{d}|\boldsymbol{\gamma}_{\mathrm{s}})}{\partial \gamma_{\mathrm{s},i} \partial \gamma_{\mathrm{s},j}}\right], \tag{9.84}$$

where the three parameters of $\boldsymbol{\gamma}_{\mathrm{s}}$, i.e., $\gamma_{\mathrm{s},1}$, $\gamma_{\mathrm{s},2}$, and $\gamma_{\mathrm{s},3}$, are respectively x, y, and z coordinates of the source location.

In the case of a Gaussian measurement error, the Fisher information matrix turns into [68]

$$\mathbf{I}(\boldsymbol{\gamma}_{\mathrm{s}}) = \left[\frac{\partial \mathbf{g}(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial \boldsymbol{\gamma}_{\mathrm{s}}}\right]^T \mathbf{C}_{\boldsymbol{\epsilon}}^{-1} \left[\frac{\partial \mathbf{g}(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial \boldsymbol{\gamma}_{\mathrm{s}}}\right], \tag{9.85}$$

where $\partial \mathbf{g}(\boldsymbol{\gamma}_{\mathrm{s}})/\partial \boldsymbol{\gamma}_{\mathrm{s}}$ is an $(N-1) \times 3$ Jacobian matrix defined as,

$$\begin{aligned}\frac{\partial \mathbf{g}(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial \boldsymbol{\gamma}_{\mathrm{s}}} &= \begin{bmatrix} \frac{\partial g_2(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial x_{\mathrm{s}}} & \frac{\partial g_2(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial y_{\mathrm{s}}} & \frac{\partial g_2(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial z_{\mathrm{s}}} \\ \frac{\partial g_3(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial x_{\mathrm{s}}} & \frac{\partial g_3(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial y_{\mathrm{s}}} & \frac{\partial g_3(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial z_{\mathrm{s}}} \\ \vdots & \vdots & \vdots \\ \frac{\partial g_N(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial x_{\mathrm{s}}} & \frac{\partial g_N(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial y_{\mathrm{s}}} & \frac{\partial g_N(\boldsymbol{\gamma}_{\mathrm{s}})}{\partial z_{\mathrm{s}}} \end{bmatrix} \\ &= \begin{bmatrix} (\mathbf{u}_2 - \mathbf{u}_1)^T \\ (\mathbf{u}_3 - \mathbf{u}_1)^T \\ \vdots \\ (\mathbf{u}_N - \mathbf{u}_1)^T \end{bmatrix},\end{aligned} \tag{9.86}$$

and

$$\mathbf{u}_n = \frac{\boldsymbol{\gamma}_{\mathrm{s}} - \boldsymbol{\gamma}_n}{\|\boldsymbol{\gamma}_{\mathrm{s}} - \boldsymbol{\gamma}_n\|} = \frac{\boldsymbol{\gamma}_{\mathrm{s}} - \boldsymbol{\gamma}_n}{\eta_n},\ n = 1, 2, \cdots, N, \tag{9.87}$$

is the normalized vector of unit length pointing from the nth microphone to the sound source.

9.11 Algorithm Overview

There is a rich literature of source localization techniques that use the additive measurement error model given in the previous section. Important distinctions between these methods include likelihood-based versus least-squares and linear approximation versus direct numerical optimization, as well as iterative versus closed-form algorithms.

In early research of source localization with passive sensor arrays, the maximum likelihood (ML) principle was widely utilized [154], [313], [286], [75] because of the proven asymptotic consistency and efficiency of an ML estimator (MLE). However, the number of microphones in an array for camera pointing or beamformer steering in multimedia communication systems is always limited, which makes acoustic source localization a finite-sample rather than a large-sample problem. Moreover, ML estimators require additional assumptions about the distributions of the measurement errors. One approach is to invoke the central limit theorem and assumes a Gaussian approximation, which makes the likelihood function easy to formulate. Although a Gaussian error was justified by Hahn and Tretter [154] for continuous-time processing, it can be difficult to verify and the MLE is no longer optimal when sampling introduces additional errors in discrete-time processing. To compute the solution to the MLE, a linear approximation and iterative numerical techniques have to be used because of the nonlinearity of the hyperbolic equations. The Newton-Raphson iterative method [12], the Gauss-Newton method [121], and the least-mean-square (LMS) algorithm are among possible choices. But for these iterative approaches, selecting a good initial guesstimate to avoid a local minimum is difficult and convergence to the optimal solution cannot be guaranteed. Therefore, an ML-based estimator is usually difficult to implement in real-time source-localization systems.

For real-time applications, closed-form estimators have also gained wider attention. Of the closed-form estimators, triangulation is the most straightforward [308]. However, with triangulation it is difficult to take advantage of extra sensors and the TDOA redundancy. Nowadays most closed-form algorithms exploit a least-squares principle, which makes no additional assumption about the distribution of measurement errors. To construct a least-squares estimator, one needs to define an error function based on the measured TDOAs. Different error functions will result in different estimators with different complexity and performance. Schmidt showed that the TDOAs to three sensors whose positions are known provide a straight line of possible source locations in two dimensions and a plane in three dimensions. By intersecting the lines/planes specified by different sensor triplets, he obtained an estimator called plane intersection. Another closed-form estimator, termed spherical intersection (SX), employed a spherical LS criterion [267]. The SX algorithm is mathematically simple, but requires an *a priori* solution for the source range, which may not exist or may not be unique in the presence of measurement errors. Based on the same criterion, Smith and Abel [279] proposed the spherical interpolation (SI) method, which also solved for the source range, again in the LS sense. Although the SI method has less bias, it is not efficient and it has a large standard deviation relative to the Cramèr-Rao lower bound (CRLB). With the SI estimator, the source range is a byproduct that is assumed to be independent of the location coordinates. Chan and Ho [68] improved the SI estimation with a second LS estimator that accommodates the information redundancy from the SI estimates and updates the squares of the coordinates. We shall refer to this method as the quadratic-correction least-squares (QCLS) approach. In the QCLS estimator, the covariance matrix of measurement errors is used. But this information can be difficult to properly assume or accurately estimate, which results in a performance degradation in practice. When the SI estimate is analyzed and the quadratic correction is derived in the QCLS estimation procedure, perturbation approaches are employed and, presumptively, the magnitude of measurement errors has to be small. It has been indicated in [69] that the QCLS estimator yields an unbiased solution with a small standard deviation that is close to the CRLB at a moderate noise level. But when noise is practically strong, its bias is considerable and its variance could no longer approach the CRLB according to our Monte-Carlo simulations. Recently a linear-correction least-squares (LCLS) algorithm has been proposed in [176]. This method applies the additive measurement error model and employs the technique of Lagrange multipliers. It makes no assumption on the covariance matrix of measurement errors.

In the rest of this chapter, we will discuss various source-localization algorithms in detail and evaluate them in terms of estimation accuracy and efficiency, computational complexity, implementation flexibility, and adaptation capabilities to different and varying environments.

9.12 Maximum Likelihood Estimator

The measurement model for the source localization problem was investigated and the CRLB for any unbiased estimator was determined in Sect. 9.10. Since the measurement model is highly nonlinear, an efficient estimator that attains the CRLB may not exist or might be impossible to find even if it does exist. In practice, the maximum likelihood estimator is often used since it has the well-proven advantage of asymptotic efficiency for a large sample space.

To apply the maximum likelihood principle, the statistical characteristics of the measurements need to be known or properly assumed prior to any processing. From the central limit theorem and also for mathematical simplicity, the measurement error is usually modeled as Gaussian and the likelihood function is given by (9.83), which is considered as a function of the source position $\boldsymbol{\gamma}_\mathrm{s}$ under estimation.

Since the exponential function is monotonically increasing, the MLE is equivalent to minimizing a (log-likelihood) cost function defined as,

$$\mathcal{L}(\boldsymbol{\gamma}_\mathrm{s}) \overset{\triangle}{=} [\mathbf{d} - \mathbf{g}(\boldsymbol{\gamma}_\mathrm{s})]^T \, \mathbf{C}_{\boldsymbol{\epsilon}}^{-1} \, [\mathbf{d} - \mathbf{g}(\boldsymbol{\gamma}_\mathrm{s})] \, . \tag{9.88}$$

Direct estimation of the minimizer is generally not practical. If the noise signals at different microphones are assumed to be uncorrelated, the covariance matrix is diagonal:

$$\mathbf{C}_{\boldsymbol{\epsilon}} = \mathrm{diag}(\sigma_2^2, \sigma_2^2, \cdots, \sigma_N^2), \tag{9.89}$$

where σ_n^2 $(n = 2, 3, \cdots, N)$ is the variance of ϵ_n, and the cost function (9.88) becomes,

$$\mathcal{L}(\boldsymbol{\gamma}_\mathrm{s}) = \sum_{n=2}^{N} \frac{[d_{n1} - g_n(\boldsymbol{\gamma}_\mathrm{s})]^2}{\sigma_n^2}. \tag{9.90}$$

Among other approaches, the steepest descent algorithm can be used to find $\hat{\boldsymbol{\gamma}}_{\mathrm{s,MLE}}$ iteratively with

$$\hat{\boldsymbol{\gamma}}_\mathrm{s}(k+1) = \hat{\boldsymbol{\gamma}}_\mathrm{s}(k) - \frac{1}{2}\mu \bigtriangledown \mathcal{L}[\hat{\boldsymbol{\gamma}}_\mathrm{s}(k)], \tag{9.91}$$

where μ is the step size.

The foregoing MLE can be determined and is asymptotically optimal for this problem only if its two assumptions (Gaussian and uncorrelated measurement noise) hold. However, this is not the case in practice as discussed in Sect. 9.10. Furthermore, the number of microphones in an array for camera pointing or beamformer steering is always limited, which makes the source localization a finite-sample rather than a large-sample problem. In addition, the cost function (9.90) is generally not strictly convex. In order to avoid a local minimum with the steepest descent algorithm, we need to select a good initial guesstimate of the source location, which is difficult to do in practice, and convergence of the iterative algorithm to the desired solution cannot be guaranteed.

9.13 Least-Squares Estimators

Two limitations of the MLE are that probabilistic assumptions have to be made about the measured range differences and that the iterative algorithm to find the solution is computationally intensive. An alternative method is the well-known least-squares estimator (LSE). The LSE makes no probabilistic assumptions about the data and hence can be applied to the source localization problem in which a precise statistical characterization of the data is hard to determine. Furthermore, an LSE usually produces a closed-form estimate that is desirable in real-time applications.

9.13.1 Least-Squares Error Criteria

In the LS approach, we attempt to minimize a squared error function that is zero in the absence of noise and model inaccuracies. Different error functions can be defined for closeness from the assumed (noiseless) signal based on hypothesized parameters to the observed data. When these are applied, different LSEs can be derived. For the source localization problem two LS error criteria can be constructed.

Hyperbolic LS Error Function

The first LS error function is defined as the difference between the observed range difference and that generated by a signal model depending upon the unknown parameters. Such an error function is routinely used in many LS estimators

$$\mathbf{e}_{\mathrm{h}}(\boldsymbol{\gamma}_{\mathrm{s}}) \overset{\triangle}{=} \mathbf{d} - \mathbf{g}(\boldsymbol{\gamma}_{\mathrm{s}}), \tag{9.92}$$

and the corresponding LS criterion is given by

$$J_{\mathrm{h}} = \mathbf{e}_{\mathrm{h}}^T \mathbf{e}_{\mathrm{h}} = [\mathbf{d} - \mathbf{g}(\boldsymbol{\gamma}_{\mathrm{s}})]^T [\mathbf{d} - \mathbf{g}(\boldsymbol{\gamma}_{\mathrm{s}})]. \tag{9.93}$$

In the source localization problem, an observed range difference d_{n1} defines a hyperboloid in 3-D space. All points lying on such a hyperboloid are potential source locations and all have the same range difference d_{n1} to the two microphones n and 1. Therefore, a sound source that is located by minimizing the hyperbolic LS error criterion (9.93) has the shortest distance to all hyperboloids associated with different microphone pairs and specified by the estimated range differences.

In (9.92), the signal model $\mathbf{g}(\boldsymbol{\gamma}_{\mathrm{s}})$ consists of a set of hyperbolic functions. Since they are nonlinear, minimizing (9.93) leads to a mathematically intractable solution as N gets large. Moreover, the hyperbolic function is very sensitive to noise, especially for far-field sources. As a result, it is rarely used in practice.

When the statistical characteristics of the corrupting noise are unknown, uncorrelated *white* Gaussian noise is one reasonable assumption. In this case, it is not surprising that the hyperbolic LSE and the MLE minimize (maximize) similar criteria.

Spherical LS Error Function

The second LS criterion is based on the errors found in the distances from a hypothesized source location to the microphones. In the absence of measurement errors, the correct source location is preferably at the intersection of a group of spheres centered at the microphones. When measurement errors are present, the best estimate of the source location would be the point that yields the shortest distance to those spheres defined by the range differences and the hypothesized source range.

Consider the distance η_n from the nth microphone to the source. From the definition of the range difference (9.78) and the fact that $\eta_1 = \zeta_\mathrm{s}$, we have:

$$\hat{\eta}_n = \zeta_\mathrm{s} + d_{n1}, \tag{9.94}$$

where $\hat{\eta}_n$ denotes an observation based on the measured range difference. From the inner product, we can derive the true value for η_n^2, the square of the noise-free distance generated by a spherical signal model:

$$\eta_n^2 = \|\boldsymbol{\gamma}_n - \boldsymbol{\gamma}_\mathrm{s}\|^2 = \zeta_n^2 - 2\boldsymbol{\gamma}_n^T\boldsymbol{\gamma}_\mathrm{s} + \zeta_\mathrm{s}^2. \tag{9.95}$$

The spherical LS error function is then defined as the difference between the measured and hypothesized values

$$\begin{aligned} e_{\mathrm{sp},n}(\boldsymbol{\gamma}_\mathrm{s}) &\triangleq \frac{1}{2}\left(\hat{\eta}_n^2 - \eta_n^2\right) \qquad (9.96)\\ &= \boldsymbol{\gamma}_n^T\boldsymbol{\gamma}_\mathrm{s} + d_{n1}\zeta_\mathrm{s} - \frac{1}{2}(\zeta_n^2 - d_{n1}^2), \quad n = 2, 3, \cdots, N. \end{aligned}$$

Putting the N errors together and writing them in a vector form gives,

$$\mathbf{e}_\mathrm{sp}(\mathbf{r}_\mathrm{s}) = \mathbf{A}\boldsymbol{\theta} - \mathbf{b}_\mathrm{r}, \tag{9.97}$$

where

$$\mathbf{A} \triangleq \left[\,\mathbf{S}\,\middle|\,\mathbf{d}\,\right], \quad \mathbf{S} \triangleq \begin{bmatrix} x_2 & y_2 & z_2 \\ x_3 & y_3 & z_3 \\ & \vdots & \\ x_N & y_N & z_N \end{bmatrix},$$

$$\boldsymbol{\theta} \triangleq \begin{bmatrix} x_\mathrm{s} \\ y_\mathrm{s} \\ z_\mathrm{s} \\ \zeta_\mathrm{s} \end{bmatrix}, \quad \mathbf{b}_\mathrm{r} \triangleq \frac{1}{2}\begin{bmatrix} \zeta_2^2 - d_{21}^2 \\ \zeta_3^2 - d_{31}^2 \\ \vdots \\ \zeta_N^2 - d_{N1}^2 \end{bmatrix},$$

and $\left[\,\mathbf{S}\,\middle|\,\mathbf{d}\,\right]$ indicates that $\mathbf{S}$ and $\mathbf{d}$ are stacked side-by-side. The corresponding LS criterion is then given by:

$$J_\mathrm{sp} = \mathbf{e}_\mathrm{sp}^T\mathbf{e}_\mathrm{sp} = [\mathbf{A}\boldsymbol{\theta} - \mathbf{b}_\mathrm{r}]^T[\mathbf{A}\boldsymbol{\theta} - \mathbf{b}_\mathrm{r}]. \tag{9.98}$$

In contrast to the hyperbolic error function (9.92), the spherical error function (9.97) is linear in $\boldsymbol{\gamma}_\mathrm{s}$ given ζ_s and vice versa. Therefore, the computational complexity to find a solution will *not* dramatically increase as N gets large.

9.13.2 Spherical Intersection (SX) Estimator

The SX source location estimator employs the spherical error and solves the problem in two steps [267]. It first finds the least-squares solution for $\boldsymbol{\gamma}_\mathrm{s}$ in terms of ζ_s,

$$\boldsymbol{\gamma}_\mathrm{s} = \mathbf{S}^\dagger(\mathbf{b}_\mathrm{r} - \zeta_\mathrm{s}\mathbf{d}), \tag{9.99}$$

where

$$\mathbf{S}^\dagger = \left(\mathbf{S}^T\mathbf{S}\right)^{-1}\mathbf{S}^T$$

is the pseudo-inverse of matrix $\mathbf{S}$. Then, substituting (9.99) into the constraint $\zeta_\mathrm{s}^2 = \boldsymbol{\gamma}_\mathrm{s}^T\boldsymbol{\gamma}_\mathrm{s}$ yields a quadratic equation as follows

$$\zeta_\mathrm{s}^2 = \left[\mathbf{S}^\dagger(\mathbf{b}_\mathrm{r} - \zeta_\mathrm{s}\mathbf{d})\right]^T \left[\mathbf{S}^\dagger(\mathbf{b}_\mathrm{r} - \zeta_\mathrm{s}\mathbf{d})\right]. \tag{9.100}$$

After expansion, it becomes

$$\alpha_1\zeta_\mathrm{s}^2 + \alpha_2\zeta_\mathrm{s} + \alpha_3 = 0, \tag{9.101}$$

where

$$\alpha_1 = 1 - \|\mathbf{S}^\dagger\mathbf{d}\|^2,\ \alpha_2 = 2\mathbf{b}_\mathrm{r}^T\mathbf{S}^{\dagger^T}\mathbf{S}^\dagger\mathbf{d},\ \alpha_3 = -\|\mathbf{S}^\dagger\mathbf{b}_\mathrm{r}\|^2.$$

The valid (real, positive) root is taken as an estimate of the source range ζ_s and is then substituted into (9.99) to calculate the SX estimate $\hat{\boldsymbol{\gamma}}_\mathrm{s,SX}$ of the source location.

In the SX estimation procedure, the solution of the quadratic equation (9.101) for the source range ζ_s is required. This solution must be a positive value by all means. If a real positive root is not available, the SX solution does not *exist*. On the contrary, if both of the roots are real and greater than 0, then the SX solution is not *unique*. In both cases, the SX source location estimator fails to produce a reliable estimate, which is not desirable for a real-time implementation.

9.13.3 Spherical Interpolation (SI) Estimator

In order to overcome the drawback of the SX algorithm, a spherical interpolation estimator was proposed in [2] which attempts to relax the restriction $\zeta_\mathrm{s} = \|\boldsymbol{\gamma}_\mathrm{s}\|$ by estimating ζ_s in the least-squares sense.

To begin, we substitute the least-squares solution (9.99) into the original spherical equation $\mathbf{A}\boldsymbol{\theta} = \mathbf{b}_\mathrm{r}$ to obtain

$$\zeta_\mathrm{s}\mathbf{P}_{\mathbf{S}^\perp}\mathbf{d} = \mathbf{P}_{\mathbf{S}^\perp}\mathbf{b}_\mathrm{r}, \tag{9.102}$$

where

$$\mathbf{P}_{\mathbf{S}^\perp} \overset{\triangle}{=} \mathbf{I}_{N\times N} - \mathbf{S}\mathbf{S}^\dagger, \tag{9.103}$$

and $\mathbf{I}_{N\times N}$ is an $N \times N$ identity matrix. Matrix $\mathbf{P}_{\mathbf{S}^\perp}$ is a projection matrix that projects a vector, when multiplied by the matrix, onto a space that is orthogonal to the column space of $\mathbf{S}$. Such a projection matrix is symmetric (i.e. $\mathbf{P}_{\mathbf{S}^\perp} = \mathbf{P}_{\mathbf{S}^\perp}^T$) and idempotent (i.e. $\mathbf{P}_{\mathbf{S}^\perp} = \mathbf{P}_{\mathbf{S}^\perp} \cdot \mathbf{P}_{\mathbf{S}^\perp}$). Then the least-squares solution to (9.102) is given by

$$\hat{\zeta}_{\mathrm{s,SI}} = \frac{\mathbf{d}^T \mathbf{P}_{\mathbf{S}^\perp} \mathbf{b}_{\mathrm{r}}}{\mathbf{d}^T \mathbf{P}_{\mathbf{S}^\perp} \mathbf{d}}. \tag{9.104}$$

Substituting this solution into (9.99) yields the SI estimate

$$\hat{\gamma}_{\mathrm{s,SI}} = \mathbf{S}^\dagger \left[\mathbf{I}_{N\times N} - \left(\frac{\mathbf{d}\mathbf{d}^T \mathbf{P}_{\mathbf{S}^\perp}}{\mathbf{d}^T \mathbf{P}_{\mathbf{S}^\perp} \mathbf{d}} \right) \right] \mathbf{b}_{\mathrm{r}}. \tag{9.105}$$

In practice, the SI estimator performs better, but is computationally a little bit more complex, than the SX estimator.

9.13.4 Linear-Correction Least-Squares Estimator

Finding the LSE based on the spherical error criterion (9.98) is a linear minimization problem, i.e.,

$$\hat{\boldsymbol{\theta}}_{\mathrm{LSE}} = \arg\min_{\boldsymbol{\theta}} \ (\mathbf{A}\boldsymbol{\theta} - \mathbf{b}_{\mathrm{r}})^T (\mathbf{A}\boldsymbol{\theta} - \mathbf{b}_{\mathrm{r}}) \tag{9.106}$$

subject to a quadratic constraint

$$\boldsymbol{\theta}^T \boldsymbol{\Xi} \boldsymbol{\theta} = 0, \tag{9.107}$$

where $\boldsymbol{\Xi} \overset{\triangle}{=} \mathrm{diag}(1, 1, 1, -1)$ is a diagonal and orthonormal matrix.

For such a constrained minimization problem, the technique of Lagrange multipliers will be used and the source location is determined by minimizing the Lagrangian

$$\begin{aligned} \mathcal{L}(\boldsymbol{\theta}, \kappa) &= J_{\mathrm{sp}} + \kappa \boldsymbol{\theta}^T \boldsymbol{\Xi} \boldsymbol{\theta} \\ &= (\mathbf{A}\boldsymbol{\theta} - \mathbf{b}_{\mathrm{r}})^T (\mathbf{A}\boldsymbol{\theta} - \mathbf{b}_{\mathrm{r}}) + \kappa \boldsymbol{\theta}^T \boldsymbol{\Xi} \boldsymbol{\theta}, \end{aligned}$$

where κ is a Lagrange multiplier. Expanding this expression yields

$$\mathcal{L}(\boldsymbol{\theta}, \kappa) = \boldsymbol{\theta}^T (\mathbf{A}^T \mathbf{A} + \kappa \boldsymbol{\Xi}) \boldsymbol{\theta} - 2\mathbf{b}_{\mathrm{r}}^T \mathbf{A} \boldsymbol{\theta} + \mathbf{b}_{\mathrm{r}}^T \mathbf{b}_{\mathrm{r}}. \tag{9.108}$$

Necessary conditions for minimizing (9.108) can be obtained by taking the gradient of $\mathcal{L}(\boldsymbol{\theta}, \kappa)$ with respect to $\boldsymbol{\theta}$ and equating the result to zero. This produces:

$$\frac{\partial \mathcal{L}(\boldsymbol{\theta}, \kappa)}{\partial \boldsymbol{\theta}} = 2 \left(\mathbf{A}^T \mathbf{A} + \kappa \boldsymbol{\Xi} \right) \boldsymbol{\theta} - 2\mathbf{A}^T \mathbf{b}_{\mathrm{r}} = \mathbf{0}. \tag{9.109}$$

Solving for $\boldsymbol{\theta}$ yields the constrained least squares estimate

$$\hat{\boldsymbol{\theta}} = \left(\mathbf{A}^T \mathbf{A} + \kappa \boldsymbol{\Xi} \right)^{-1} \mathbf{A}^T \mathbf{b}_{\mathrm{r}}, \tag{9.110}$$

where κ is yet to be determined.

In order to find κ, we can impose the quadratic constraint directly by substituting (9.110) into (9.107), which leads to

$$\mathbf{b}_{\mathrm{r}}^T \mathbf{A} \left(\mathbf{A}^T \mathbf{A} + \kappa \boldsymbol{\Xi}\right)^{-1} \boldsymbol{\Xi} \left(\mathbf{A}^T \mathbf{A} + \kappa \boldsymbol{\Xi}\right)^{-1} \mathbf{A}^T \mathbf{b}_{\mathrm{r}} = 0. \tag{9.111}$$

With eigenvalue analysis, the matrix $\mathbf{A}^T \mathbf{A} \boldsymbol{\Xi}$ can be decomposed as

$$\mathbf{A}^T \mathbf{A} \boldsymbol{\Xi} = \mathbf{U} \boldsymbol{\Lambda} \mathbf{U}^{-1}, \tag{9.112}$$

where $\boldsymbol{\Lambda} = \mathrm{diag}(\lambda_1, \cdots, \lambda_4)$ and $\lambda_i,\ i = 1, \cdots, 4$, are the eigenvalues of the matrix $\mathbf{A}^T \mathbf{A} \boldsymbol{\Xi}$. Substituting (9.112) into (9.111), we may rewrite the constraint as:

$$\mathbf{p}^T (\boldsymbol{\Lambda} + \kappa \mathbf{I})^{-2} \mathbf{q} = 0, \tag{9.113}$$

where

$$\begin{aligned} \mathbf{p} &= \mathbf{U}^T \boldsymbol{\Xi} \mathbf{A}^T \mathbf{b}_{\mathrm{r}}, \\ \mathbf{q} &= \mathbf{U}^T \mathbf{A}^T \mathbf{b}_{\mathrm{r}}. \end{aligned}$$

Define a function of the Lagrange multiplier as follows

$$\begin{aligned} f(\kappa) &\stackrel{\triangle}{=} \mathbf{p}^T (\boldsymbol{\Lambda} + \kappa \mathbf{I})^{-2} \mathbf{q} \\ &= \sum_{i=1}^{4} \frac{p_i q_i}{(\kappa + \lambda_i)^2}. \end{aligned} \tag{9.114}$$

This is a polynomial of degree eight and because of its complexity numerical methods need to be used for root searching. Since the root of (9.114) for κ is not unique, a two-step procedure will be followed such that the desired source location could be found.

Unconstrained Spherical Least Squares Estimator

In the first step, we assume that x_{s}, y_{s}, z_{s}, and ζ_{s} are mutually independent or equivalently disregard the quadratic constraint (9.107) in purpose. Then the LS solution minimizing (9.98) for $\boldsymbol{\theta}$ (the source location as well as its range) is given by

$$\hat{\boldsymbol{\theta}}_1 = \mathbf{A}^\dagger \mathbf{b}_{\mathrm{r}}, \tag{9.115}$$

where

$$\mathbf{A}^\dagger = \left(\mathbf{A}^T \mathbf{A}\right)^{-1} \mathbf{A}^T$$

is the pseudo-inverse of the matrix $\mathbf{A}$.

A good parameter estimator first and foremost needs to be unbiased. For such an unconstrained spherical least squares estimator, the bias and covariance matrix can be approximated by using the following perturbation analysis method.

When measurement errors are present in the range differences, $\mathbf{A}$, $\mathbf{b}_\mathrm{r}$, and the parameter estimate $\hat{\boldsymbol{\theta}}_1$ deviate from their true values and can be expressed as:

$$\mathbf{A} = \mathbf{A}^\mathrm{t} + \triangle\mathbf{A},\ \mathbf{b}_\mathrm{r} = \mathbf{b}_\mathrm{r}^\mathrm{t} + \varDelta\mathbf{b}_\mathrm{r},\ \hat{\boldsymbol{\theta}}_1 = \boldsymbol{\theta}^\mathrm{t} + \triangle\boldsymbol{\theta}, \tag{9.116}$$

where variables with superscript t denote the true values which also satisfy

$$\boldsymbol{\theta}^\mathrm{t} = \mathbf{A}^{\mathrm{t}^\dagger}\mathbf{b}_\mathrm{r}^\mathrm{t}. \tag{9.117}$$

If the magnitudes of the perturbations are small, the second-order errors are insignificant compared to their first-order counterparts and therefore can be neglected for simplicity, which then yields:

$$\triangle\mathbf{A} = \left[\,\mathbf{0}\,\middle|\,\boldsymbol{\epsilon}\,\right],\quad \triangle\mathbf{b}_\mathrm{r} \approx -\mathbf{d}^\mathrm{t} \odot \boldsymbol{\epsilon}, \tag{9.118}$$

where $\odot$ denotes the Schur (element-by-element) product. Substituting (9.116) into (9.115) gives,

$$\left(\mathbf{A}^\mathrm{t} + \triangle\mathbf{A}\right)^T \left(\mathbf{A}^\mathrm{t} + \triangle\mathbf{A}\right)\left(\boldsymbol{\theta}^\mathrm{t} + \triangle\boldsymbol{\theta}\right) = \left(\mathbf{A}^\mathrm{t} + \triangle\mathbf{A}\right)^T \left(\mathbf{b}_\mathrm{r}^\mathrm{t} + \triangle\mathbf{b}_\mathrm{r}\right). \tag{9.119}$$

Retaining only the linear perturbation terms and using (9.117) and (9.118) produces:

$$\triangle\boldsymbol{\theta} \approx -\mathbf{A}^{\mathrm{t}^\dagger}\mathbf{D}\boldsymbol{\epsilon}, \tag{9.120}$$

where

$$\mathbf{D} \overset{\triangle}{=} \mathrm{diag}(\eta_2, \eta_3, \cdots, \eta_N)$$

is a diagonal matrix. Since the measurement error $\boldsymbol{\epsilon}$ in the range differences has zero mean, $\hat{\boldsymbol{\theta}}_1$ is an unbiased estimate of $\boldsymbol{\theta}^\mathrm{t}$ when the small error assumption holds:

$$E\{\triangle\boldsymbol{\theta}\} \approx E\left\{-\mathbf{A}^{t^\dagger}\mathbf{D}\boldsymbol{\epsilon}\right\} = \mathbf{0}_{4\times 1}. \tag{9.121}$$

The covariance matrix of $\triangle\boldsymbol{\theta}$ is then found as,

$$\mathbf{C}_{\triangle\boldsymbol{\theta}} = E\{\triangle\boldsymbol{\theta}\triangle\boldsymbol{\theta}^T\} = \mathbf{A}^{\mathrm{t}^\dagger}\mathbf{D}\mathbf{C}_{\boldsymbol{\epsilon}}\mathbf{D}\mathbf{A}^{\mathrm{t}^{\dagger^T}}, \tag{9.122}$$

where $\mathbf{C}_{\boldsymbol{\epsilon}}$ is known or is properly assumed *a priori*. Theoretically, the covariance matrix $\mathbf{C}_{\triangle\boldsymbol{\theta}}$ cannot be calculated since it contains true values. Nevertheless, it can be approximated by using the values in $\hat{\boldsymbol{\theta}}_1$ with sufficient accuracy, as suggested by our numerical studies.

In the first unconstrained spherical LS estimate (9.115), the range information is redundant because of the independence assumption on the source location and range. If that information is simply discarded, the source location estimate is the same as the SI estimate but with less computational complexity [175]. To demonstrate this, we first write (9.115) into a block form as

$$\hat{\boldsymbol{\theta}}_1 = \begin{bmatrix} \mathbf{S}^T\mathbf{S} & \mathbf{S}^T\mathbf{d} \\ \mathbf{d}^T\mathbf{S} & \mathbf{d}^T\mathbf{d} \end{bmatrix}^{-1} \begin{bmatrix} \mathbf{S}^T \\ \mathbf{d}^T \end{bmatrix} \mathbf{b}_{\mathrm{r}}. \tag{9.123}$$

It can easily be shown that:

$$\begin{bmatrix} \mathbf{S}^T\mathbf{S} & \mathbf{S}^T\mathbf{d} \\ \mathbf{d}^T\mathbf{S} & \mathbf{d}^T\mathbf{d} \end{bmatrix}^{-1} = \begin{bmatrix} \mathbf{Q} & \mathbf{v} \\ \mathbf{v}^T & \aleph \end{bmatrix}, \tag{9.124}$$

where

$$\begin{aligned} \mathbf{v} &= -\left(\mathbf{S}^T\mathbf{S} - \frac{\mathbf{S}^T\mathbf{d}\mathbf{d}^T\mathbf{S}}{\mathbf{d}^T\mathbf{d}}\right)^{-1} \frac{\mathbf{S}^T\mathbf{d}}{\mathbf{d}^T\mathbf{d}}, \\ \mathbf{Q} &= \left(\mathbf{S}^T\mathbf{S}\right)^{-1} \left[\mathbf{I} - \left(\mathbf{S}^T\mathbf{d}\right)\mathbf{v}^T\right], \\ \aleph &= \frac{1 - (\mathbf{d}^T\mathbf{S})\mathbf{v}}{\mathbf{d}^T\mathbf{d}}. \end{aligned}$$

Next, we define another projection matrix $\mathbf{P}_{\mathbf{d}^\perp}$ associated with the $\mathbf{d}$-orthogonal space:

$$\mathbf{P}_{\mathbf{d}^\perp} \triangleq \mathbf{I} - \frac{\mathbf{d}\mathbf{d}^T}{\mathbf{d}^T\mathbf{d}}, \tag{9.125}$$

and find

$$\mathbf{v} = -\left(\mathbf{S}^T\mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\right)^{-1} \frac{\mathbf{S}^T\mathbf{d}}{\mathbf{d}^T\mathbf{d}}, \tag{9.126}$$

$$\mathbf{Q} = \left(\mathbf{S}^T\mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\right)^{-1}. \tag{9.127}$$

Substituting (9.124) together with (9.126) and (9.127) into (9.123) yields the unconstrained spherical LS estimate for source coordinates,

$$\hat{\boldsymbol{\gamma}}_{\mathrm{s},1} = \left(\mathbf{S}^T\mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\right)^{-1} \mathbf{S}^T\mathbf{P}_{\mathbf{d}^\perp}\mathbf{b}_{\mathrm{r}}, \tag{9.128}$$

which is the minimizer of

$$J_1(\boldsymbol{\gamma}_{\mathrm{s}}) = \|\mathbf{P}_{\mathbf{d}^\perp}\mathbf{b}\mathbf{b}_{\mathrm{r}} - \mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\boldsymbol{\gamma}_{\mathrm{s}}\|^2, \tag{9.129}$$

or the least-squares solution to the linear equation

$$\mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\boldsymbol{\gamma}_{\mathrm{s}} = \mathbf{P}_{\mathbf{d}^\perp}\mathbf{b}_{\mathrm{r}}. \tag{9.130}$$

In fact, the first unconstrained spherical LS estimator tries to approximate the projection of the observation vector $\mathbf{b}_{\mathrm{r}}$ with the projections of the column vectors of the microphone location matrix $\mathbf{S}$ onto the $\mathbf{d}$-orthogonal space. The source location estimate is the coefficient vector associated with the *best* approximation. Clearly from (9.130), this estimation procedure is the generalization of the plane intersection (PI) method proposed in [268].

By using the Sherman-Morrison formula [229]

$$\left(\mathbf{A}+\mathbf{x}\mathbf{y}^T\right)^{-1}=\mathbf{A}^{-1}-\frac{\mathbf{A}^{-1}\mathbf{x}\mathbf{y}^T\mathbf{A}^{-1}}{1+\mathbf{y}^T\mathbf{A}^{-1}\mathbf{x}}, \tag{9.131}$$

we can expand the item in (9.128) as

$$\left(\mathbf{S}^T\mathbf{P}_{\mathbf{d}^\perp}\mathbf{S}\right)^{-1}=\left[\mathbf{S}^T\mathbf{S}-\left(\frac{\mathbf{S}^T\mathbf{d}}{\mathbf{d}^T\mathbf{d}}\right)(\mathbf{S}^T\mathbf{d})^T\right]^{-1},$$

and finally can show that the unconstrained spherical LS estimate (9.128) is equivalent to the SI estimate (9.105), i.e. $\hat{\gamma}_{\mathrm{s},1}\equiv\hat{\gamma}_{\mathrm{s,SI}}$.

Although the unconstrained spherical LS and the SI estimators are mathematically equivalent, they are quite different in computational efficiency due to different approaches to the source localization problem. The complexities of the SI and unconstrained spherical LS estimators are in $\mathcal{O}\left(N^3\right)$ and $\mathcal{O}(N)$, respectively. In comparison, the unconstrained spherical LS estimator reduces the complexity of the SI estimator by a factor of N^2, which is significant when N is large (more microphones are used).

Linear Correction

In the previous subsection, we developed the unconstrained spherical LS estimator (USLSE) for source localization and demonstrated that it is mathematically equivalent to the SI estimator but with less computational complexity. Although the USLSE/SI estimates can be accurate as indicated in [175] among others, it is helpful to exploit the redundancy of source range for improving the statistical efficiency (i.e., to reduce the variance of source location estimates) of the overall estimation procedure. Therefore, in the second step, we intend to correct the USLS estimate $\hat{\boldsymbol{\theta}}_1$ to make a better estimate $\hat{\boldsymbol{\theta}}_2$ of $\boldsymbol{\theta}$. This new estimate should be in the neighborhood of $\hat{\boldsymbol{\theta}}_1$ and should obey the constraint (9.107). We expect that the corrected estimate would still be unbiased and would have a smaller variance.

To begin, we substitute $\hat{\boldsymbol{\theta}}_1=\boldsymbol{\theta}^{\mathrm{t}}+\triangle\boldsymbol{\theta}$ into (9.109) and expand the expression to find

$$\mathbf{A}^T\mathbf{A}\hat{\boldsymbol{\theta}}_1+\kappa\Xi\hat{\boldsymbol{\theta}}_1-(\mathbf{A}^T\mathbf{A}+\kappa\Xi)\triangle\boldsymbol{\theta}=\mathbf{A}^T\mathbf{b}_{\mathrm{r}}. \tag{9.132}$$

Combined with (9.115), (9.132) becomes

$$(\mathbf{A}^T\mathbf{A}+\kappa\Xi)\triangle\boldsymbol{\theta}=\kappa\Xi\hat{\boldsymbol{\theta}}_1, \tag{9.133}$$

and hence

$$\triangle\boldsymbol{\theta}=\kappa\left(\mathbf{A}^T\mathbf{A}\right)^{-1}\Xi\boldsymbol{\theta}^{\mathrm{t}}. \tag{9.134}$$

Substituting (9.134) into $\hat{\boldsymbol{\theta}}_1=\boldsymbol{\theta}^{\mathrm{t}}+\triangle\boldsymbol{\theta}$ yields

$$\hat{\boldsymbol{\theta}}_1=\left[\mathbf{I}+\kappa\left(\mathbf{A}^T\mathbf{A}\right)^{-1}\Xi\right]\boldsymbol{\theta}^{\mathrm{t}}. \tag{9.135}$$

Solving for $\boldsymbol{\theta}^{\mathrm{t}}$ produces the corrected estimate $\hat{\boldsymbol{\theta}}_2$ and also the final output of the linear-correction least-squares (LCLS) estimator:

$$\hat{\boldsymbol{\theta}}_2 = \left[\mathbf{I} + \kappa \left(\mathbf{A}^T\mathbf{A}\right)^{-1} \boldsymbol{\Xi}\right]^{-1} \hat{\boldsymbol{\theta}}_1. \tag{9.136}$$

Equation (9.136) suggests how the second-step processing updates the source location estimate based on the first unconstrained spherical least squares result, or equivalently the SI estimate. If the regularity condition [224]

$$\lim_{i\to\infty} \left(\kappa(\mathbf{A}^T\mathbf{A})^{-1}\boldsymbol{\Xi}\right)^i = \mathbf{0} \tag{9.137}$$

is satisfied, then the estimate $\hat{\boldsymbol{\theta}}_2$ can be expanded in a Neumann series:

$$\begin{aligned} \hat{\boldsymbol{\theta}}_2 &= \left[\mathbf{I} + \left(-\kappa \left(\mathbf{A}^T\mathbf{A}\right)^{-1} \boldsymbol{\Xi}\right) + \left(-\kappa \left(\mathbf{A}^T\mathbf{A}\right)^{-1} \boldsymbol{\Xi}\right)^2 + \cdots\right] \hat{\boldsymbol{\theta}}_1 \\ &= \hat{\boldsymbol{\theta}}_1 + \sum_{i=1}^{\infty} \left[-\kappa \left(\mathbf{A}^T\mathbf{A}\right)^{-1} \boldsymbol{\Xi}\right]^i \hat{\boldsymbol{\theta}}_1, \end{aligned} \tag{9.138}$$

where the second term is the linear correction. Equation (9.137) implies that in order to avoid divergence, the Lagrange multiplier κ should be small. In addition, κ needs to be determined carefully such that $\hat{\boldsymbol{\theta}}_2$ obeys the quadratic constraint (9.107).

Because the function $f(\kappa)$ is smooth near $\kappa = 0$ (corresponding to the neighborhood of $\hat{\boldsymbol{\theta}}_1$), as suggested by numerical experiments, the secant method [254] can be used to determine its desired root. Two reasonable initial points can be chosen as:

$$\kappa_0 = 0, \quad \kappa_1 = \beta, \tag{9.139}$$

where the small number β is dependent on the array geometry. Five iterations should be sufficient to give an accurate approximation to the root.

The idea of exploiting the relationship between a sound source's range and its location coordinates to improve the estimation efficiency of the SI estimator was first suggested by Chan and Ho in [68] with a quadratic correction. Accordingly, they constructed a quadratic data model for $\hat{\boldsymbol{\theta}}_1$.

$$\hat{\boldsymbol{\theta}}_1 \odot \hat{\boldsymbol{\theta}}_1 = \mathbf{T}(\boldsymbol{\gamma}_{\mathrm{s}} \odot \boldsymbol{\gamma}_{\mathrm{s}}) + \mathbf{n}, \tag{9.140}$$

where

$$\mathbf{T} = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 1 \end{bmatrix}$$

is a constant matrix, and $\mathbf{n}$ is the corrupting noise. In contrast to the linear correction technique based on the Lagrange multiplier, the quadratic counterpart needs to know the covariance matrix $\mathbf{C}_{\boldsymbol{\epsilon}}$ of measurement errors in the range differences *a priori*. In a real-time digital source localization system, a poorly estimated $\mathbf{C}_{\boldsymbol{\epsilon}}$ will lead to performance degradation. In addition, the quadratic-correction least squares estimation procedure uses the perturbation approaches to linearly approximate $\triangle\boldsymbol{\theta}$ and $\mathbf{n}$ in (9.116) and (9.140), respectively. Therefore, the approximations of their corresponding covariance matrices $\mathbf{C}_{\triangle\boldsymbol{\theta}}$ and $\mathbf{C}_{\mathbf{n}}$ can be good only when the noise level is low. When noise is at a practically high level, the quadratic-correction least squares estimate has a large bias and a high variance. Furthermore, since the true value of the source location which is necessary for calculating $\mathbf{C}_{\triangle\boldsymbol{\theta}}$ and $\mathbf{C}_{\mathbf{n}}$ cannot be known theoretically, the estimated source location has to be utilized for approximation. It was suggested in [68] that several iterations in the second correction stage would improve estimation accuracy. However, while the bias is suppressed after iterations, the estimate is closer to the SI solution and the variance is boosted, as demonstrated in [176]. Finally, the direct solutions of the quadratic-correction least-squares estimator are the squares of the source location coordinates $(\boldsymbol{\gamma}_{\mathrm{s}} \odot \boldsymbol{\gamma}_{\mathrm{s}})$. In 3-D space, these correspond to 8 positions, which introduce decision ambiguities. Other physical criteria, such as the domain of interest, were suggested but these are hard to define in practical situations, particularly when one of the source coordinates is close to zero.

In comparison, the linear-correction method updates the source location estimate of the first unconstrained spherical LS estimator without making any assumption about the error covariance matrix and without resort to a linear approximation. Even though we need to find a small root of function (9.114) for the Lagrange multiplier κ that satisfies the regularity condition (9.137), the function $f(\kappa)$ is smooth around zero and the solution can be easily determined using the secant method. The linear-correction method achieves a relatively better balance between computational complexity and estimation accuracy.

9.14 Example System Implementation

Acoustic source localization systems are not necessarily complicated and need not use computationally powerful and consequently expensive devices for running in real time, as the implementation described briefly in this section demonstrates. The real-time acoustic source localization system with passive microphone arrays for video camera steering in teleconferencing environments was developed by the authors at Bell Laboratories. Figure 9.3 shows a signal-flow diagram of the system.

This system is based on a personal computer that is powered by an Intel Pentium® III 1 GHz general-purposed processor and that runs a Microsoft Windows® operation system. Sonorus AD/24 converter and STUDI/O® digital audio interface card are employed to simultaneously capture multiple microphone signals. The camera is a Sony EVI-D30 with pan, tilt, and zoom capabilities. These motions can be harmoniously performed at the same time by separate motors, providing good coverage of a normal conference room.

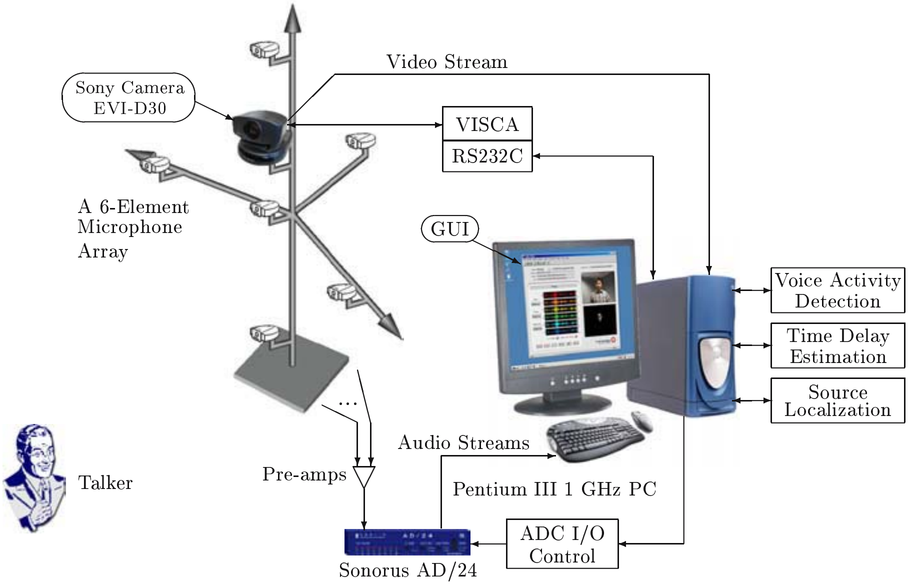


Fig. 9.3. Illustration of the real-time acoustic source localization system for video camera steering.

The host computer drives the camera via two layers of protocols, namely the RS232C serial control protocol and the Video System Control Architecture (VISCA$^{®}$) protocol. The focus of the video camera is updated four times a second and the video stream is fed into the computer through a video capture card at a rate of 30 frames per second.

The microphone array uses six Lucent Speech Tracker Directional$^{®}$ hypercardioid microphones, as illustrated in Fig. 9.4. The frequency response of these microphones is 200-6000 Hz and beyond 4 kHz there is negligible energy. Therefore microphone signals are sampled at 8 kHz and a one-stage pre-amplifier with the fixed gain 37 dB is used prior to sampling. The reference microphone 1 is located at the center (the origin of the coordinate) and the rest microphones are in the same distance of 40 cm from the reference.

The empirical bias and standard deviation data in Figs. 9.5 and 9.6 show the results of two source localization examples using four different source localization algorithms where TDOA is estimated using the AED method (a more comprehensive numerical study can be found in [176]). In the graphs of standard deviation, the CRLBs are also plotted. For the QCLS algorithm, the true value of the source location needs to be known to calculate the covariance matrix of the first-stage SI estimate. But this knowledge is practically inaccessible and the estimated source location has to be used for approximation. It is suggested in [68] that several iterations in the second correction stage could improve the estimation accuracy. In the following, we refer to the one

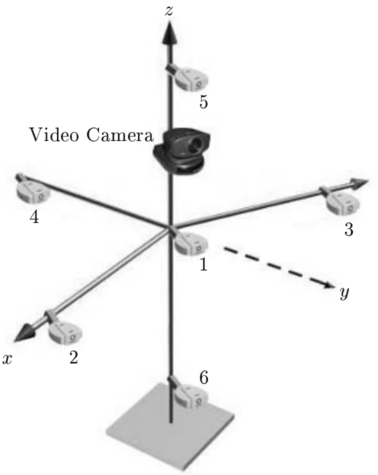


Fig. 9.4. Microphone array of the acoustic source localization system for video camera steering.

without iterations as the QCLS-i estimator and the other with iterations as the QCLS-ii estimator.

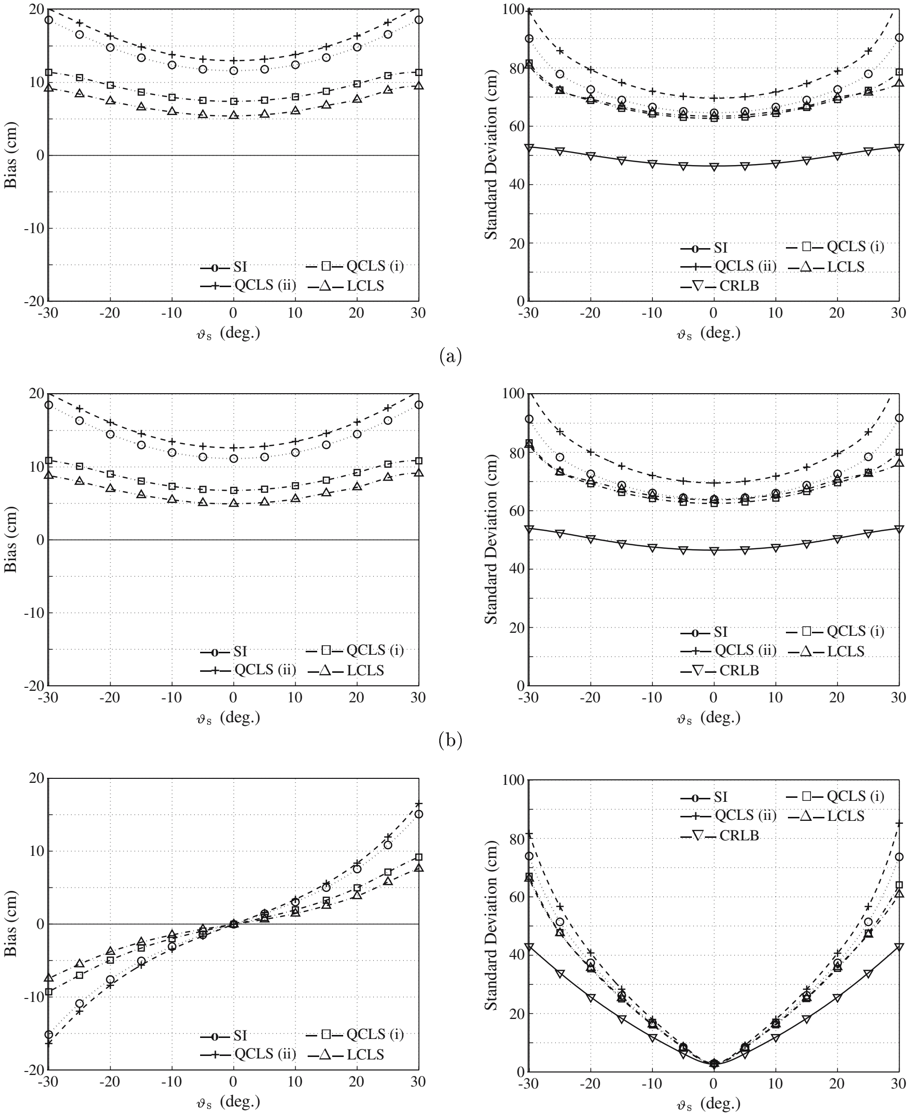
The microphone array designed for the real-time system presented above was used in these examples. As illustrated in Fig. 9.4, the six microphones are located at (distances in centimeters):

$$\begin{aligned} \gamma_1 &= (0,0,0), & \gamma_2 &= (40,0,0), & \gamma_3 &= (-40,0,0), \\ \gamma_4 &= (0,0,40), & \gamma_5 &= (0,-40,0), & \gamma_6 &= (0,0,-40). \end{aligned} \tag{9.141}$$

For such an array, the value of β in (9.139) was empirically set as $\beta = 1$. The source was positioned 300 cm away from the array with a fixed azimuth angle $\varphi_\mathrm{s} = 45°$ and varying elevation angles ϑ_s. At each location, the empirical bias and standard deviation of each estimator were obtained by averaging the results of 2000-trial Monte-Carlo runs.

In the first example, errors in time delay estimates are i.i.d. Gaussian with zero mean and 1 cm standard deviation. As seen clearly from Fig. 9.5, the QCLS-i estimator has the largest bias. Performing several iterations in the second stage can effectively reduce the estimation bias, but the solution is more like an SI estimate and the variance is boosted. In terms of standard deviation, all correction estimators perform better than the SI estimator (without correction). Among these four studied LS estimators, the QCLS-ii and the LCLS achieve the lowest standard deviation and their values approach the CRLS at most source locations.

In the second example, measurement errors are mutually dependent and their covariance matrix is given by [68]:

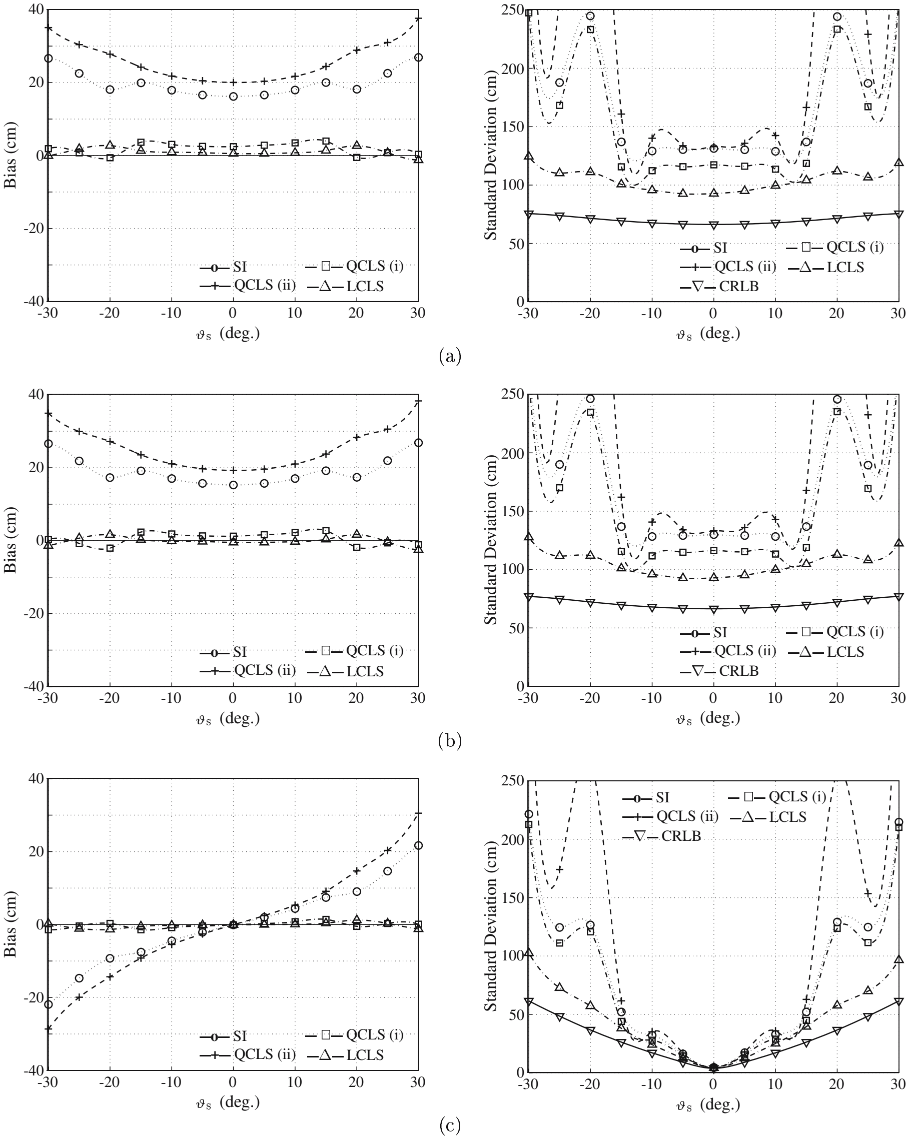


(a)

(b)

(c)

Fig. 9.5. Comparisons of empirical bias and standard deviation among the SI, QCLS-i, QCLS-ii, and LCLS estimators with zero mean i.i.d. Gaussian errors of standard deviation $\sigma_\epsilon = 1$ cm. (a) Estimators of x_s, (b) estimators of y_s, (c) estimators of z_s.



(a)

(b)

(c)

Fig. 9.6. Comparisons of empirical bias and standard deviation among the SI, QCLS-i, QCLS-ii, and LCLS estimators with zero mean *colored* Gaussian errors of standard deviation $\sigma_\epsilon = 1$ cm. (a) Estimators of x_s, (b) estimators of y_s, (c) estimators of z_s.

$$\mathbf{C}_{\boldsymbol{\epsilon}} = \frac{\sigma_\epsilon^2}{2} \begin{bmatrix} 2 & 1 & \cdots & 1 \\ 1 & 2 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 2 \end{bmatrix}, \tag{9.142}$$

where again $\sigma_\epsilon = 1$ cm. For a more realistic simulation, all estimators are provided with no information of the error distribution. From Fig. 9.6, we see that the performance of each estimator deteriorates because errors are no longer independent. At such a noise level, the linear approximation used by the QCLS estimators is inaccurate and the estimation procedure fails. However, the LCLS estimation procedure makes no assumption about $\mathbf{C}_{\boldsymbol{\epsilon}}$ and does not depend on a linear approximation. It produces an estimate whose bias and variance are always the smallest.

9.15 Summary

This chapter consists of two parts. The first one (from Sect. 9.1 to Sect. 9.8) was devoted to the time-delay-estimation problem in room acoustic environments. Starting from the problem formulation, we summarized the state of the art of the TDE algorithms ranging from the cross correlation based methods to the blind system identification based techniques. Since reverberation is the most difficult effect to cope with, we paid significant attention to the robustness of TDE against this effect. Fundamentally, there are three approaches to improving the robustness of TDE. When we have access to some *a priori* knowledge about the distortion sources, this *a priori* knowledge can be used to improve TDE. In case that reverberation is the most contaminant distortions, blind system identification techniques would be a good choice for TDE. If we have an array of sensors, we can improve TDE by taking advantage of the redundant information provided by multiple sensors.

The second part discussed the acoustic source localization techniques, which will play a significant role in the next-generation multimedia communication systems. The problem was postulated from a perspective of the estimation theory and the Cramèr-Rao lower bound for unbiased location estimators was derived. After an insightful review of conventional approaches ranging from maximum likelihood to least squares estimators, we presented a recently developed linear-correction least-squares algorithm that is more robust to measurement errors and that is more computationally as well as statistically efficient. As an example, we presented an acoustic source localization system for video camera steering in teleconferencing, which used a cheap Intel Pentium® III general-purposed processor.