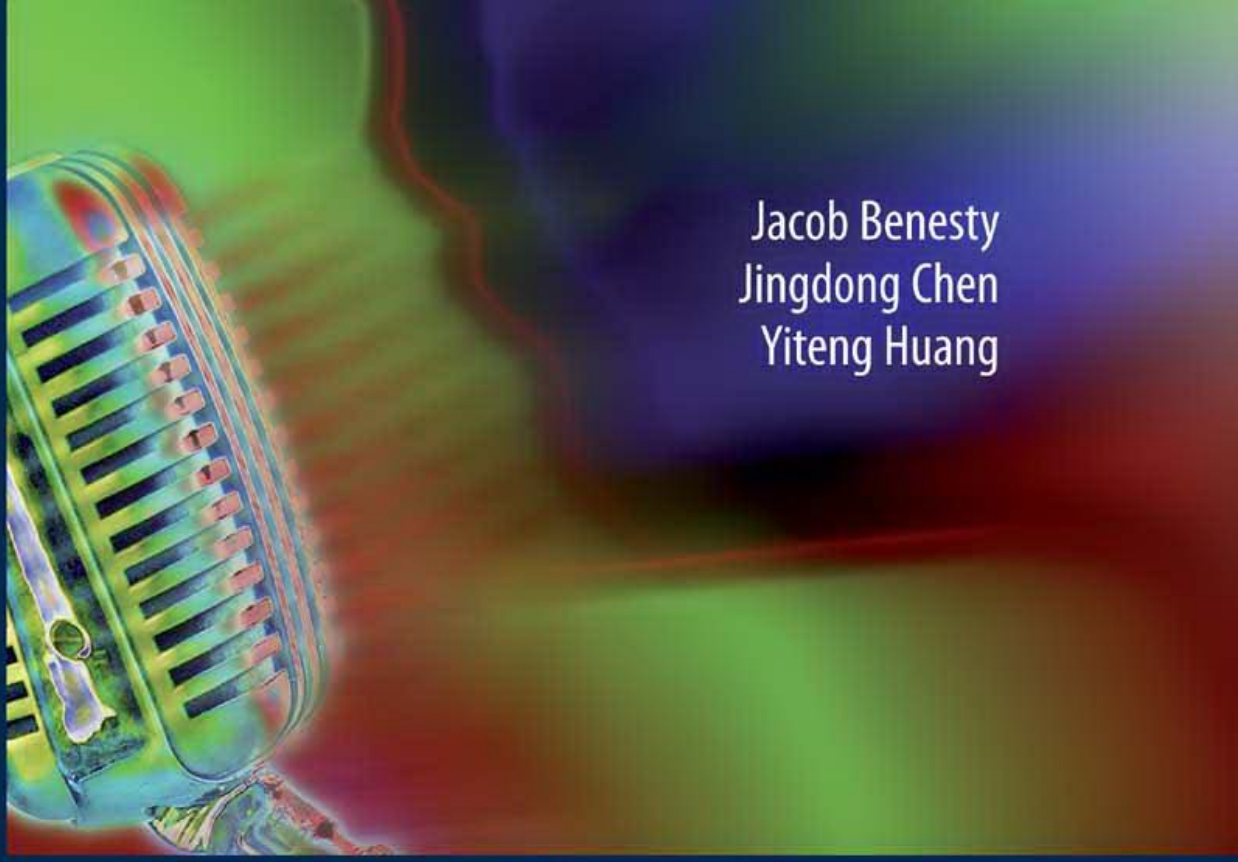


Jacob Benesty
Jingdong Chen
Yiteng Huang

SPRINGER TOPICS IN SIGNAL PROCESSING

Microphone Array Signal Processing

Springer

Springer Topics in Signal Processing

Volume 1

Springer Topics in Signal Processing

Edited by J. Benesty and W. Kellermann

Vol. 1: Benesty, J.; Chen, J.; Huang, Y.
Microphone Array Signal Processing
250 p. 2008 [978-3-540-78611-5]

Jacob Benesty · Jingdong Chen · Yiteng Huang

Microphone Array Signal Processing

Springer

Jacob Benesty
INRS-EMT, University of Quebec
800 de la Gauchetiere Ouest
Montreal, QC, H5A 1K6
Canada

Yiteng Huang
WeVoice, Inc.
9 Sylvan Dr.
Bridgewater, NJ, 08807
USA

Jingdong Chen
Bell Labs, Alcatel-Lucent
600 Mountain Ave.
Murray Hill, NJ, 07974
USA

ISBN 978-3-540-78611-5 e-ISBN 978-3-540-78612-2

DOI 10.1007/978-3-540-78612-2

Springer Topics in Signal Processing ISSN 1866-2609

Library of Congress Control Number: 2008922312

Coverdesign: WMXDesign GmbH, Heidelberg

Printed on acid-free paper

9 8 7 6 5 4 3 2 1

springer.com

Preface

In the past few years we have written and edited several books in the area of acoustic and speech signal processing. The reason behind this endeavor is that there were almost no books available in the literature when we first started while there was (and still is) a real need to publish manuscripts summarizing the most useful ideas, concepts, results, and state-of-the-art algorithms in this important area of research. According to all the feedback we have received so far, we can say that we were right in doing this. Recently, several other researchers have followed us in this journey and have published interesting books with their own visions and perspectives.

The idea of writing a book on *Microphone Array Signal Processing* comes from discussions we have had with many colleagues and friends. As a consequence of these discussions, we came up with the conclusion that, again, there is an urgent need for a monograph that carefully explains the theory and implementation of microphone arrays. While there are many manuscripts on antenna arrays from a narrowband perspective (narrowband signals and narrowband processing), the literature is quite scarce when it comes to sensor arrays explained from a truly broadband perspective. Many algorithms for speech applications were simply borrowed from narrowband antenna arrays. However, a direct application of narrowband ideas to broadband speech processing may not be necessarily appropriate and can lead to many misunderstandings. Therefore, the main objective of this book is to derive and explain the most fundamental algorithms from a strict broadband (signals and/or processing) viewpoint. Thanks to the approach taken here, new concepts come in light that have the great potential of solving several and very difficult problems encountered in acoustic and speech applications.

This book is especially written for graduate students and research engineers who work on microphone arrays. Our goal is to make the area of microphone array signal processing theory and application available in a complete and self-contained text. We attempt to explain the main ideas in a clear and rigorous way so that the reader can have a pretty good idea of the po-

tentials, opportunities, challenges, and limitations of microphone array signal processing. We hope that the reader will find it useful and inspiring.

Finally, we would like to thank Christoph Baumann, Petra Jantzen, and Carmen Wolf from Springer (Germany) for their wonderful help in the preparation and publication of this manuscript. Working with them is always a pleasure and a wonderful experience.

Montréal, QC/ Murray Hill, NJ/ Bridgewater, NJ

Jacob Benesty
Jingdong Chen
Yiteng Huang

Contents

1

Introduction

1.1 Microphone Array Signal Processing

A microphone array consists of a set of microphones positioned in a way that the spatial information is well captured. To make an analogy with wireless communications, we can talk about spatial diversity. This diversity, represented by the acoustic impulse responses from a radiating source to the sensors, can be understood and exploited in different ways as will be explained throughout this book. These acoustic channels, modeled as finite impulse response (FIR) filters, are usually not identical; the most problematic situation is when the FIR filters share common zeroes.

The rich information available thanks to the diversity needs to be processed. Then, the main objective of *microphone array signal processing* is the estimation of some parameters or the extraction of some signals of interest, depending on the application, by using the spatio-temporal (and possibly frequency) information available at the output of the microphone array. Although the particular case of a single-microphone system is also covered (in Chapter 2 in the context of optimal filtering and in Chapter 6 in the context of noise reduction in the frequency domain), the major focus of this book is on the use of a multiple-sensor system since it allows more flexibility to solve many important practical problems.

Depending on the nature of the applications, the geometry of the microphone array may play an important role in the formulation of the processing algorithms. For example, in source localization the array geometry must be known in order to be able to localize a source properly; moreover, sometimes a regular geometry will even simplify the problem of estimation, that is why uniform linear and circular arrays are often used [148]. Today, these two geometries dominate the market but we see more and more sophisticated three-dimensional spherical arrays as they can better capture the sound field [163], [164]. However, in some other crucial problems such as noise reduction or source separation, the geometry of the array may have little (or no) importance depending on the algorithm. In this case, we may say that we have

a multiple microphone system instead of a microphone array. It is not necessary to distinguish the two situations since it will become quite obvious in the context.

The problems encountered in microphone arrays may look easy to tackle because similar problems have been tackled for a long period of time in narrowband antenna arrays. But this is quite deceiving. Actually, microphone arrays work differently than antenna arrays for applications such as radar and sonar for the following reasons [105], [215]:

- speech is a wideband signal,
- reverberation of the room (or multipath) is high,
- environments and signals are highly non-stationary,
- noise can have the same spectral characteristics as the desired speech signal,
- the number of sensors is usually restricted, and
- the human ear has an extremely wide dynamic range (as much as 120 dB for normal hearing) and is very sensitive to weak tails of the channel impulse responses. As a result, the length of the modeling filters is very long (thousands of samples are not uncommon).

For these main reasons, we should not be surprised that for some problems, many existing algorithms do not perform well.

A large number of algorithms for microphone array processing were borrowed or generalized (in a very simple manner) from narrowband array processing [51]. The advantage of this demarche is that most algorithms conceived for decades in antenna arrays can be extended without much efforts. The drawback, though, is that none of these algorithms are tailored to work in real acoustic environments. As a result, performances are often very limited. Simply put, microphone arrays require broadband processing. This is the approach taken, in general, in this book.

The main problems that have the potential to be solved with microphone arrays are

- noise reduction,
- echo reduction,
- dereverberation,
- localization of a single source,
- estimation of the number of sources,
- localization of multiple sources,
- source separation, and
- cocktail party.

Most of these problems are depicted in Fig. 1.1 where all the signals picked up by the microphones pass through some filters that need to be optimized according to one of the above-mentioned problems we want to solve [20].

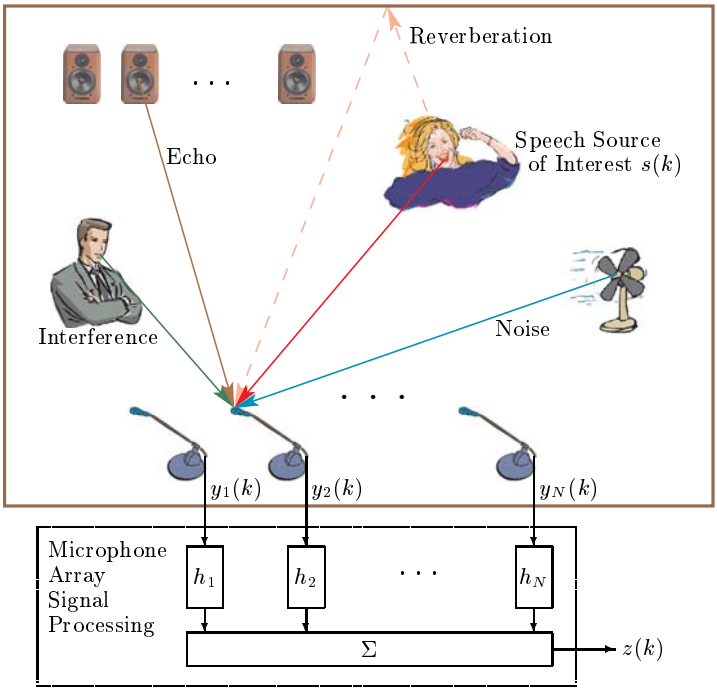


Fig. 1.1. Microphone array signal processing.

The aim of a noise reduction algorithm is to estimate a desired speech signal from its corrupted observations that are due to the effects of an unwanted additive noise. Many techniques based on a single microphone already exist [16], [154], [156]. The main problem, tough, with all these single-channel algorithms is that they distort the speech signal [41], [42]. While the speech quality may be improved, the speech intelligibility is degraded. However, with a microphone array, we should be able to reduce (at least in theory) the noise without affecting much the speech signal.

In hands-free communications the acoustic coupling between loudspeakers and microphones, associated with the overall delay, would produce echoes that would make real-time conversations very difficult [10], [29], [84], [98], [99], [121]. Furthermore, the acoustic system could become very instable. It was believed that a microphone array would be able to significantly reduce the level of echoes by directing the array towards the source of interest and putting nulls towards the loudspeakers. Unfortunately, this idea even though very attractive and elegant, does not work in practice and the acoustic echo cancellation approach [10] to this problem is still the best, and by far, solution today.

In a room and in a hands-free context, the signals that are picked by microphones from a talker contain not only the direct-path signals, but also

attenuated and delayed replicas of the source signal due to reflections from boundaries and objects in this room. This multipath propagation effect introduces echoes and spectral distortions into the observation signals, termed as reverberation, which may severely deteriorate the source signal causing quality and intelligibility degradation. Therefore, dereverberation is required to improve the intelligibility of the speech signal [125]. Great efforts have been going on for the last four decades to find practical solutions with a microphone array.

In acoustic environments, the source location information plays an important role for applications such as automatic camera tracking for videoconferencing and beamformer steering for suppressing noise and reverberation. Estimation of the source location, which is often called source-localization problem, has been of considerable interest for decades [26], [117], [175], [222]. Two or three dimensional microphone arrays are required to estimate the angle of arrival or the position in Cartesian coordinates of a source. For the two related problems of estimating the number of sources and localizing multiple sources, several interesting algorithms exist for narrowband signals; however, researchers have just started to investigate these problems for broadband sources.

In source separation with multiple microphones, we try to separate different signals coming, at the same time, from different directions. All the approaches are blind in nature since we have no access to neither the acoustic channels nor the source signals. Independent component analysis (ICA) [127] is the most widely used tool for the blind source separation (BSS) problem, since it takes fully advantage of the independence of the source signals. While most of the algorithms based on ICA work very well when the signals are mixed instantaneously, they do not perform that well in a reverberant (convolutive) environment. Although much progress has been made recently, it is still not clear how and to what degree this can be useful in speech and acoustic applications. Since the literature is already very rich in ICA [159] (see also references in [182]), we will not discuss BSS in this book from this perspective.

It has been known for some time that humans have the ability of focusing on one particular voice or sound amid a cacophony of distracting conversations or background noise. This interesting psychoacoustic phenomenon is referred to as the *cocktail party effect* [45], [46]. One of the important observations from the psychoacoustic experiments of [45] and [46] is that spatial hearing plays a very important role. Our perception of speech remarkably benefits from spatial hearing. This ability is mainly attributed to the fact that we have two ears. This is intuitively justified by our daily experience and can be further demonstrated simply by observing the difference in understanding between using both ears and with either ear covered when listening in an enclosed space where there are multiple speakers at the same time [125]. While humans with a normal hearing and some brain processing can effectively handle this cocktail party problem with not much effort, it is still very tricky with microphone array signal processing. This is the mother of all challenges in this area of

research and until today we still do not have a clear idea how to solve this problem.

All the aforementioned problems are very difficult to solve no matter the size, the geometry, or the number of elements of the array. Sometimes, a specific geometry of the array or an increase in the number of microphones can give a more accurate solution to an estimation problem. However, the gain may be limited or even negligible. Then, some fundamental questions arise: how do we exploit the spatial information? How far can we go to solving a specific problem? What are the appropriate models? Where are the limits and why? Can we go beyond the spatial information and if so how?

Our objective in this book is not to expose different and state-of-the-art solutions to all the problems explained previously but rather to give a general framework, important tools, and signal models that will help readers understand how to process multiple microphone signals intuitively yet rigorously.

To conclude this section, let us briefly mention the typical applications of microphone arrays:

- teleconferencing,
- multi-party telecommunications,
- hands-free acoustic human-machine interfaces,
- dialogue systems,
- computer games,
- command-and-control interfaces,
- dictation systems,
- high-quality audio recordings,
- acoustic surveillance (security and monitoring),
- acoustic scene analysis,
- sensor network technology.

We see that the number of applications is enormous and growing every day. Clearly, the market is still waiting for good microphone array solutions before that such systems can be widely deployed.

1.2 Organization of the Book

This book contains ten chapters (including this one). We tried to cover the most important topics of microphone array signal processing, from a fresh perspective, in the next nine chapters. Each chapter builds up important concepts so the reader can follow the ideas from the basic theory to practical applications. Although the chapters are coherently tied to each other, the material was written so that each chapter can be studied almost independently.

Linear optimal filters play a fundamental role in many applications of signal processing including microphone arrays. The concepts behind optimal filtering are easy to understand and are important for the rest of this book. Chapter 2 studies the Wiener, Frost, and Kalman filters. It also develops the

concept of the Pearson correlation coefficient as an alternative to the mean-square error (MSE). This development leads to many interesting results.

Conventional beamforming methods for spatial filtering in narrowband antenna arrays are very well established. In Chapter 3, we discuss the most well-known techniques using a simple propagation signal model and in the context of signal enhancement. The philosophy behind the broadband beamforming, which is of more interest with speech signals, is also introduced.

The linearly constrained minimum variance (LCMV) filter is extremely popular in antenna arrays. This optimal filter is quite powerful thanks to all the constraints that can be adjoined to the cost function from which it is derived. Chapter 4 shows how the LCMV filter can be used in room acoustic environments, for noise reduction and dereverberation, by using three different signal models.

Chapter 5 is dedicated to the problem of noise reduction with multiple microphones. Several classical methods are derived in the multichannel case within a unique framework. All important aspects of speech enhancement such as the levels of noise reduction and speech distortion are discussed.

Chapter 6 is concerned with the noncausal (frequency-domain) Wiener filter and its application to noise reduction. Both the single- and multi-channel cases are developed. Many fundamental aspects in the context of speech enhancement are derived to help the reader better understand how frequency-domain algorithms work, especially with multiple microphones.

In Chapter 7, the desired and interference sources on the one hand and the microphone signals on the other hand are treated as a multiple-input multiple-output (MIMO) system. A general framework based on the MIMO channel impulse responses is then developed for analyzing beamforming performance for source extraction, dereverberation, and interference suppression.

Chapter 8 is a continuation of Chapter 7. It is shown how the two problems of interference sources and reverberation can be separated in a distinguishable manner in a two-step approach. The conditions for that are also clearly demonstrated. Thanks to this separation we better understand the interactions between source separation and dereverberation.

Chapter 9 concerns the important problem of direction-of-arrival (DOA) and time-difference-of-arrival (TDOA) estimation. The focus is more on the TDOA estimation algorithms since the problem of the DOA estimation is essentially the same as the TDOA estimation. Many algorithms are developed: from the classical ones such as the cross-correlation method to more modern and new methods such as the minimum entropy technique. The principles for TDOA estimation of multiple sources are also discussed.

Chapter 10 concludes this book with a discussion on some unaddressed problems.

2

Classical Optimal Filtering

2.1 Introduction

In his landmark manuscript on extrapolation, interpolation and smoothing of stationary time series [234], Norbert Wiener was one of the first researchers to treat the filtering problem of estimating a process corrupted by additive noise. The optimum estimate that he derived, required the solution of an integral equation known as the Wiener-Hopf equation [233]. Soon after Wiener published his work, Levinson formulated the same problem in discrete time [152]. Levinson's contribution has had a great impact on the field. Indeed, thanks to him, Wiener's ideas have become more accessible to many engineers and, as a result, more practical. A very nice overview of linear filtering theory and the history of the different discoveries in this area can be found in [136].

The Wiener filter has been used in a very large number of applications thanks to its simple formulation and its effectiveness. However, this optimal filter is not adequate for nonstationary signals. Moreover, in many situations it distorts the signal of interest as explained later in this chapter.

In 1960, R. E. Kalman published his famous paper describing a recursive solution to the discrete-data linear filtering problem [137]. This so-called Kalman filter is based on the fact that the desired signal follows a state model and, in contrast to the Wiener filter, it is tailored to work well in nonstationary environments. Another merit of this sequential filter is that, if the modeling is correct, the desired signal will not be distorted.

This chapter is dedicated to the study of three important filters often encountered in microphone arrays: Wiener, linearly constrained minimum variance, and Kalman filters. We also propose a new alternative to the mean-square error (MSE) criterion (used to derive the Wiener filter) based on the Pearson correlation coefficient and show why it may be more convenient to use in general.

2.2 Wiener Filter

Consider a zero-mean clean speech signal $x(k)$ contaminated by a zero-mean noise process $v(k)$ [white or colored but uncorrelated with $x(k)$], so that the noisy speech signal, at the discrete time sample k is

$$y(k) = x(k) + v(k). \tag{2.1}$$

Assuming that all signals are stationary, our objective in this section is to find an optimal estimate of $x(k)$ in the Wiener sense [234].

Define the error signal between the clean speech sample at time k and its estimate

$$\begin{aligned} e(k) &= x(k) - z(k) \\ &= x(k) - \mathbf{h}^T \mathbf{y}(k), \end{aligned} \tag{2.2}$$

where

$$\mathbf{h} = \begin{bmatrix} h_0 \; h_1 \; \cdots \; h_{L-1} \end{bmatrix}^T$$

is a finite impulse response (FIR) filter of length L, superscript ^{T} denotes transpose of a vector or a matrix,

$$\mathbf{y}(k) = \begin{bmatrix} y(k) \; y(k-1) \; \cdots \; y(k-L+1) \end{bmatrix}^T$$

is a vector containing the L most recent samples of the observation signal $y(k)$, and

$$z(k) = \mathbf{h}^T \mathbf{y}(k) \tag{2.3}$$

is the output of the filter $\mathbf{h}$.

We now can write the MSE criterion [103]:

$$\begin{aligned} J(\mathbf{h}) &= E\left[e^2(k)\right] \\ &= \mathbf{h}^T \mathbf{R}_{yy} \mathbf{h} - 2\mathbf{r}_{yx}^T \mathbf{h} + \sigma_x^2, \end{aligned} \tag{2.4}$$

where $E[\cdot]$ denotes mathematical expectation,

$$\mathbf{R}_{yy} = E\left[\mathbf{y}(k)\mathbf{y}^T(k)\right] \tag{2.5}$$

is the correlation matrix, assumed to be full rank, of the observation signal $y(k)$,

$$\mathbf{r}_{yx} = E\left[\mathbf{y}(k)x(k)\right] \tag{2.6}$$

is the cross-correlation vector between the noisy and clean speech signals, and $\sigma_x^2 = E\left[x^2(k)\right]$ is the variance of the signal $x(k)$. Then the optimal Wiener filter is obtained as follows

$$\begin{aligned}\mathbf{h}_{\mathrm{W}} &= \arg\min_{\mathbf{h}} J(\mathbf{h}) \\ &= \mathbf{R}_{yy}^{-1}\mathbf{r}_{yx}. \end{aligned} \tag{2.7}$$

However, $x(k)$ is unobservable; as a result, an estimation of $\mathbf{r}_{yx}$ may seem difficult to obtain. But

$$\begin{aligned}\mathbf{r}_{yx} &= E\left[\mathbf{y}(k)x(k)\right] \\ &= E\left\{\mathbf{y}(k)\left[y(k)-v(k)\right]\right\} \\ &= E\left[\mathbf{y}(k)y(k)\right] - E\left\{\left[\mathbf{x}(k)+\mathbf{v}(k)\right]v(k)\right\} \\ &= E\left[\mathbf{y}(k)y(k)\right] - E[\mathbf{v}(k)v(k)] \\ &= \mathbf{r}_{yy}-\mathbf{r}_{vv}. \end{aligned} \tag{2.8}$$

Now $\mathbf{r}_{yx}$ depends on the correlation vectors $\mathbf{r}_{yy}$ and $\mathbf{r}_{vv}$. The vector $\mathbf{r}_{yy}$ (which is also the first column of $\mathbf{R}_{yy}$) can be easily estimated during speech and noise periods while $\mathbf{r}_{vv}$ can be estimated during noise-only intervals.

Consider the particular filter

$$\mathbf{h}_1 = \begin{bmatrix}1 & 0 & \cdots & 0\end{bmatrix}^T \tag{2.9}$$

of length L. The corresponding MSE is

$$\begin{aligned}J(\mathbf{h}_1) &= E\left\{\left[x(k)-\mathbf{h}_1^T\mathbf{y}(k)\right]^2\right\} \\ &= E\left\{\left[x(k)-y(k)\right]^2\right\} \\ &= E\left\{v^2(k)\right\} = \sigma_v^2. \end{aligned} \tag{2.10}$$

This means that the observed signal $y(k)$ will pass the filter $\mathbf{h}_1$ unaltered (no noise reduction).

Using (2.8) and the fact that $\mathbf{h}_1 = \mathbf{R}_{yy}^{-1}\mathbf{r}_{yy}$, we obtain another form of the Wiener filter [15]:

$$\begin{aligned}\mathbf{h}_{\mathrm{W}} &= \mathbf{h}_1 - \mathbf{R}_{yy}^{-1}\mathbf{r}_{vv} \\ &= \left[\mathbf{I}-\mathbf{R}_{yy}^{-1}\mathbf{R}_{vv}\right]\mathbf{h}_1 \\ &= \left[\frac{\mathbf{I}}{\mathrm{SNR}}+\tilde{\mathbf{R}}_{vv}^{-1}\tilde{\mathbf{R}}_{xx}\right]^{-1}\tilde{\mathbf{R}}_{vv}^{-1}\tilde{\mathbf{R}}_{xx}\mathbf{h}_1, \end{aligned} \tag{2.11}$$

where

$$\mathrm{SNR} = \frac{\sigma_x^2}{\sigma_v^2} \tag{2.12}$$

is the input signal-to-noise ratio (SNR), $\mathbf{I}$ is the identity matrix, and

$$\begin{aligned}\tilde{\mathbf{R}}_{xx} &= \frac{\mathbf{R}_{xx}}{\sigma_x^2} = \frac{E\left[\mathbf{x}(k)\mathbf{x}^T(k)\right]}{\sigma_x^2}, \\ \tilde{\mathbf{R}}_{vv} &= \frac{\mathbf{R}_{vv}}{\sigma_v^2} = \frac{E\left[\mathbf{v}(k)\mathbf{v}^T(k)\right]}{\sigma_v^2}. \end{aligned}$$

We have

$$\lim_{\text{SNR}\to\infty} \mathbf{h}_{\text{W}} = \mathbf{h}_1, \tag{2.13}$$

$$\lim_{\text{SNR}\to 0} \mathbf{h}_{\text{W}} = \mathbf{0}_{L\times 1}, \tag{2.14}$$

where $\mathbf{0}_{L\times 1}$ has the same length as $\mathbf{h}_{\text{W}}$ and consists of all zeros. The minimum MSE (MMSE) is

$$\begin{aligned} J(\mathbf{h}_{\text{W}}) &= \sigma_x^2 - \mathbf{r}_{yx}^T \mathbf{h}_{\text{W}} \\ &= \sigma_v^2 - \mathbf{r}_{vv}^T \mathbf{R}_{yy}^{-1} \mathbf{r}_{vv} \\ &= \mathbf{r}_{vv}^T \mathbf{h}_{\text{W}} \\ &= \mathbf{h}_1^T \left(\mathbf{R}_{vv} - \mathbf{R}_{vv} \mathbf{R}_{yy}^{-1} \mathbf{R}_{vv} \right) \mathbf{h}_1. \end{aligned} \tag{2.15}$$

We see clearly from (2.15) that $J(\mathbf{h}_{\text{W}}) < J(\mathbf{h}_1)$; therefore, noise reduction is possible.

The normalized MMSE is

$$\begin{aligned} \tilde{J}(\mathbf{h}_{\text{W}}) &= \frac{J(\mathbf{h}_{\text{W}})}{J(\mathbf{h}_1)} \\ &= \frac{J(\mathbf{h}_{\text{W}})}{\sigma_v^2}, \end{aligned} \tag{2.16}$$

and $0 < \tilde{J}(\mathbf{h}_{\text{W}}) < 1$.

The optimal estimation of the clean speech, $x(k)$, in the Wiener sense, is then

$$\begin{aligned} z_{\text{W}}(k) &= \mathbf{h}_{\text{W}}^T \mathbf{y}(k) \\ &= y(k) - \mathbf{r}_{vv}^T \mathbf{R}_{yy}^{-1} \mathbf{y}(k). \end{aligned} \tag{2.17}$$

Therefore, the variance of this estimated signal is

$$\begin{aligned} E\left[z_{\text{W}}^2(k) \right] &= \mathbf{h}_{\text{W}}^T \mathbf{R}_{yy} \mathbf{h}_{\text{W}} \\ &= \mathbf{h}_{\text{W}}^T \mathbf{R}_{xx} \mathbf{h}_{\text{W}} + \mathbf{h}_{\text{W}}^T \mathbf{R}_{vv} \mathbf{h}_{\text{W}}, \end{aligned} \tag{2.18}$$

which is the sum of two terms. The first one is the power of the attenuated clean speech and the second one is the power of the residual noise (always greater than zero). While noise reduction is feasible with the Wiener filter, expression (2.18) shows that the price to pay for this is also a reduction of the clean speech; this contributes to speech distortion.

We define the noise-reduction factor (with the Wiener filter) as [15]

$$\begin{aligned} \xi_{\text{nr}}\left(\mathbf{h}_{\text{W}}\right) &= \frac{\mathbf{h}_1^T \mathbf{R}_{vv} \mathbf{h}_1}{\mathbf{h}_{\text{W}}^T \mathbf{R}_{vv} \mathbf{h}_{\text{W}}} \\ &= \frac{\mathbf{h}_1^T \mathbf{R}_{vv} \mathbf{h}_1}{\mathbf{h}_1^T \mathbf{R}_{xx} \mathbf{R}_{yy}^{-1} \mathbf{R}_{vv} \mathbf{R}_{yy}^{-1} \mathbf{R}_{xx} \mathbf{h}_1} \end{aligned} \tag{2.19}$$

and the speech-distortion index as [15]

$$\upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right) = \frac{E\left\{\left[x(k) - \mathbf{h}_{\mathrm{W}}^{T}\mathbf{x}(k)\right]^{2}\right\}}{\sigma_{x}^{2}} = \frac{\left(\mathbf{h}_{1} - \mathbf{h}_{\mathrm{W}}\right)^{T}\mathbf{R}_{xx}\left(\mathbf{h}_{1} - \mathbf{h}_{\mathrm{W}}\right)}{\mathbf{h}_{1}^{T}\mathbf{R}_{xx}\mathbf{h}_{1}}. \tag{2.20}$$

The noise-reduction factor is always greater than 1; the higher the value of $\xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right)$, the more the noise is reduced. Also

$$\lim_{\mathrm{SNR}\to 0} \xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right) = \infty, \tag{2.21}$$

$$\lim_{\mathrm{SNR}\to \infty} \xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right) = 1. \tag{2.22}$$

The speech-distortion index is always between 0 and 1 for the Wiener filter. Also

$$\lim_{\mathrm{SNR}\to 0} \upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right) = 1, \tag{2.23}$$

$$\lim_{\mathrm{SNR}\to \infty} \upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right) = 0. \tag{2.24}$$

So when $\upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right)$ is close to 1, the speech signal is highly distorted and when $\upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right)$ is near 0, the speech signal is lowly distorted. Therefore, we see that for low SNRs the Wiener filter can have a disastrous effect on the speech signal.

As shown in [77], the two symmetric matrices $\mathbf{R}_{xx}$ and $\mathbf{R}_{vv}$ can be jointly diagonalized if $\mathbf{R}_{vv}$ is positive definite. So we have

$$\mathbf{R}_{xx} = \mathbf{B}^{T}\mathbf{\Lambda}\mathbf{B}, \tag{2.25}$$

$$\mathbf{R}_{vv} = \mathbf{B}^{T}\mathbf{B}, \tag{2.26}$$

$$\mathbf{R}_{yy} = \mathbf{B}^{T}\left[\mathbf{I} + \mathbf{\Lambda}\right]\mathbf{B}, \tag{2.27}$$

where $\mathbf{B}$ is a full rank square matrix but not necessarily orthogonal, and the diagonal matrix

$$\mathbf{\Lambda} = \mathrm{diag}\left[\lambda_{1}\ \lambda_{2}\ \cdots\ \lambda_{L}\right] \tag{2.28}$$

are the eigenvalues of the matrix $\mathbf{R}_{vv}^{-1}\mathbf{R}_{xx}$ with $\lambda_{1} \geq \lambda_{2} \geq \cdots \geq \lambda_{L} \geq 0$. Substituting (2.25)–(2.27) into (2.19), we obtain

$$\xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right) = \frac{\sum_{l=1}^{L} b_{l1}^{2}}{\sum_{l=1}^{L} \frac{\lambda_{l}^{2}}{(1+\lambda_{l})^{2}} b_{l1}^{2}}, \tag{2.29}$$

where the elements $b_{l1},\ l = 1, 2, \ldots, L$, form the first column of $\mathbf{B}$ and satisfy $\sum_{l=1}^{L} b_{l1}^{2} = \sigma_{v}^{2}$.

Also with the matrix decomposition in (2.25)–(2.27), the input SNR can be expressed as

$$\begin{aligned}\text{SNR} &= \frac{\mathbf{h}_1^T \mathbf{R}_{xx} \mathbf{h}_1}{\mathbf{h}_1^T \mathbf{R}_{vv} \mathbf{h}_1} \\ &= \frac{\sum_{l=1}^{L} \lambda_l b_{l1}^2}{\sum_{l=1}^{L} b_{l1}^2}. \end{aligned} \tag{2.30}$$

Using (2.30), we can rewrite (2.29) as

$$\begin{aligned}\xi_{\text{nr}}(\mathbf{h}_{\text{W}}) &= \frac{1}{\text{SNR}} \cdot \frac{\sum_{l=1}^{L} \lambda_l b_{l1}^2}{\sum_{l=1}^{L} \frac{\lambda_l^2}{(1+\lambda_l)^2} b_{l1}^2} \\ &= \frac{1}{\text{SNR}} \cdot \frac{\sum_{l=1}^{L} \frac{(1+\lambda_l)^2}{(1+\lambda_l)^2} \lambda_l b_{l1}^2}{\sum_{l=1}^{L} \frac{\lambda_l^2}{(1+\lambda_l)^2} b_{l1}^2} \\ &= \frac{1}{\text{SNR}} \cdot \left[\frac{\sum_{l=1}^{L} \frac{\lambda_l + \lambda_l^3}{(1+\lambda_l)^2} b_{l1}^2}{\sum_{l=1}^{L} \frac{\lambda_l^2}{(1+\lambda_l)^2} b_{l1}^2} + 2 \right]. \end{aligned} \tag{2.31}$$

Using the fact that $\lambda_l + \lambda_l^3 \geq \lambda_l^3$, we easily deduce from (2.31) that

$$\xi_{\text{nr}}(\mathbf{h}_{\text{W}}) \geq \frac{1}{\text{SNR}} \cdot \left[\frac{\sum_{l=1}^{L} \frac{\lambda_l^3}{(1+\lambda_l)^2} b_{l1}^2}{\sum_{l=1}^{L} \frac{\lambda_l^2}{(1+\lambda_l)^2} b_{l1}^2} + 2 \right]. \tag{2.32}$$

We can prove the following inequality (see the proof after the proposition in the next page):

$$\frac{\sum_{l=1}^{L} \frac{\lambda_l^3}{(1+\lambda_l)^2} b_{l1}^2}{\sum_{l=1}^{L} \frac{\lambda_l^2}{(1+\lambda_l)^2} b_{l1}^2} \geq \frac{\sum_{l=1}^{L} \lambda_l b_{l1}^2}{\sum_{l=1}^{L} b_{l1}^2} = \text{SNR}, \tag{2.33}$$

where equality holds if and only if all the λ_l's corresponding to the nonzero b_{l1} are equal, with $l = 1, 2, \ldots, L$. It follows immediately that [41], [42]

$$\xi_{\text{nr}}(\mathbf{h}_{\text{W}}) \geq \frac{\text{SNR} + 2}{\text{SNR}} \geq 1. \tag{2.34}$$

It can be checked from (2.34) that the lower bound of the noise-reduction factor is a monotonically decreasing function of the SNR. It approaches infinity when SNR comes close to 0 and tends to 1 as SNR approaches infinity. This indicates that more noise reduction can be achieved with the Wiener filter as the SNR decreases, which is, of course, desirable since as SNR drops, there will be more noise to be eliminated.

The upper bound of the speech-distortion index can be derived using the eigenvalue decomposition given in (2.25)–(2.27). Indeed, substituting (2.25)–(2.27) into (2.20), we get [41], [42]

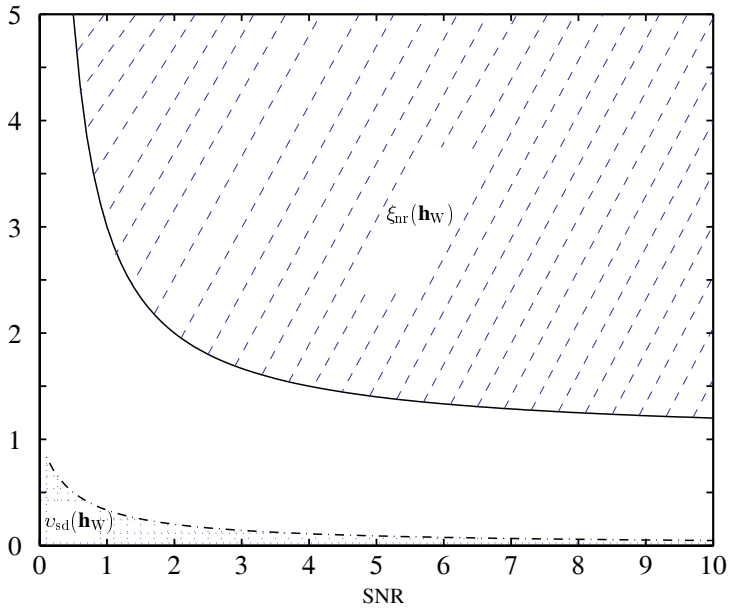


Fig. 2.1. Illustration of the areas where $\xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{W}})$ and $\upsilon_{\mathrm{sd}}(\mathbf{h}_{\mathrm{W}})$ take their values as a function of the input SNR. $\xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{W}})$ can take any value above the solid line while $\upsilon_{\mathrm{sd}}(\mathbf{h}_{\mathrm{W}})$ can take any value under the dotted line.

$$\begin{aligned}\upsilon_{\mathrm{sd}}(\mathbf{h}_{\mathrm{W}}) &= \frac{\sum_{l=1}^{L}\frac{\lambda_l}{(1+\lambda_l)^2}b_{l1}^2}{\sum_{l=1}^{L}\lambda_i b_{l1}^2} \\ &\leq \frac{\sum_{l=1}^{L}\frac{\lambda_l}{1+2\lambda_l}b_{l1}^2}{\sum_{l=1}^{L}\frac{\lambda_l+2\lambda_l^2}{1+2\lambda_l}b_{l1}^2} \\ &\leq \frac{1}{2\cdot\mathrm{SNR}+1}, \end{aligned} \tag{2.35}$$

where we have used the following inequality:

$$\frac{\sum_{l=1}^{L}\frac{\lambda_l^2}{1+2\lambda_l}b_{l1}^2}{\sum_{l=1}^{L}\frac{\lambda_l}{1+2\lambda_l}b_{l1}^2} \geq \frac{\sum_{l=1}^{L}\lambda_l b_{l1}^2}{\sum_{l=1}^{L}b_{l1}^2} = \mathrm{SNR}. \tag{2.36}$$

This inequality can be proved by induction.

Figure 2.1 illustrates the lower bound of the noise-reduction factor [eq. (2.34)] and the upper bound of the speech-distortion index [eq. (2.35)], both as a function of the input SNR.

From the previous analysis, we see that the Wiener filter achieves noise reduction at the price of speech attenuation. Therefore, the noise-reduction factor on its own may not be a satisfactory measure. In fact, the most relevant measure is the output SNR defined as

$$\mathrm{SNR}\left(\mathbf{h}_{\mathrm{W}}\right)=\frac{\mathbf{h}_{\mathrm{W}}^{T}\mathbf{R}_{xx}\mathbf{h}_{\mathrm{W}}}{\mathbf{h}_{\mathrm{W}}^{T}\mathbf{R}_{vv}\mathbf{h}_{\mathrm{W}}}, \tag{2.37}$$

and if, indeed, $\mathrm{SNR}\left(\mathbf{h}_{\mathrm{W}}\right) > \mathrm{SNR}$ then this will indicate that the Wiener filter has a real impact in reducing the noise comparatively to the speech. A key question is then whether the Wiener filter can improve the SNR. To answer this question, we give the following proposition [15], [41], [42].

Proposition. With the optimal Wiener filter given in (2.7), the output SNR [eq. (2.37)] is always greater than or at least equal to the input SNR [eq. (2.12)].

Proof. If the noise $v(k)$ is zero, the Wiener filter is equal to $\mathbf{h}_1$ and has no effect on the speech signal. Applying the matrix decomposition [eqs. (2.25)–(2.27)] in (2.37), the output SNR can be rewritten as

$$\mathrm{SNR}\left(\mathbf{h}_{\mathrm{W}}\right)=\frac{\sum_{l=1}^{L}\frac{\lambda_l^3}{(\lambda_l+1)^2}b_{l1}^2}{\sum_{l=1}^{L}\frac{\lambda_l^2}{(\lambda_l+1)^2}b_{l1}^2}. \tag{2.38}$$

Then it follows that

$$\frac{\mathrm{SNR}\left(\mathbf{h}_{\mathrm{W}}\right)}{\mathrm{SNR}}=\frac{\sum_{l=1}^{L}b_{l1}^2\cdot\sum_{l=1}^{L}\frac{\lambda_l^3}{(\lambda_l+1)^2}b_{l1}^2}{\sum_{l=1}^{L}\lambda_l b_{l1}^2\cdot\sum_{l=1}^{L}\frac{\lambda_l^2}{(\lambda_l+1)^2}b_{l1}^2}. \tag{2.39}$$

Since all the sums $\sum_{l=1}^{L}\frac{\lambda_l^3}{(1+\lambda_l)^2}b_{l1}^2$, $\sum_{l=1}^{L}\frac{\lambda_l^2}{(1+\lambda_l)^2}b_{l1}^2$, $\sum_{l=1}^{L}\lambda_l b_{l1}^2$, and $\sum_{l=1}^{L}b_{l1}^2$ are non-negative numbers, as long as we can show that the inequality

$$\sum_{l=1}^{L}\frac{\lambda_l^3}{(1+\lambda_l)^2}b_{l1}^2\sum_{l=1}^{L}b_{l1}^2\geq\sum_{l=1}^{L}\frac{\lambda_l^2}{(1+\lambda_l)^2}b_{l1}^2\sum_{l=1}^{L}\lambda_l b_{l1}^2 \tag{2.40}$$

holds, then $\mathrm{SNR}\left(\mathbf{h}_{\mathrm{W}}\right)\geq\mathrm{SNR}$. Now we prove this inequality by way of induction.

- Basic step: if $L=2$,

$$\sum_{l=1}^{2}\frac{\lambda_l^3}{(1+\lambda_l)^2}b_{l1}^2\sum_{l=1}^{2}b_{l1}^2=\frac{\lambda_1^3}{(1+\lambda_1)^2}b_{11}^4+\frac{\lambda_2^3}{(1+\lambda_2)^2}b_{21}^4+\left[\frac{\lambda_1^3}{(1+\lambda_1)^2}+\frac{\lambda_2^3}{(1+\lambda_2)^2}\right]b_{11}^2b_{21}^2.$$

  Since $\lambda_l\geq 0$, it is trivial to show that

$$\frac{\lambda_1^3}{(1+\lambda_1)^2}+\frac{\lambda_2^3}{(1+\lambda_2)^2}\geq\frac{\lambda_1^2\lambda_2}{(1+\lambda_1)^2}+\frac{\lambda_1\lambda_2^2}{(1+\lambda_2)^2},$$

  where "=" holds when $\lambda_1=\lambda_2$. Therefore

$$\sum_{l=1}^{2} \frac{\lambda_l^3}{(1+\lambda_l)^2} b_{l1}^2 \sum_{l=1}^{2} b_{l1}^2 \geq \frac{\lambda_1^3}{(1+\lambda_1)^2} b_{11}^4 + \frac{\lambda_2^3}{(1+\lambda_2)^2} b_{21}^4 + \left[\frac{\lambda_1^2 \lambda_2}{(1+\lambda_1)^2} + \frac{\lambda_1 \lambda_2^2}{(1+\lambda_2)^2} \right] b_{11}^2 b_{21}^2 = \sum_{l=1}^{2} \frac{\lambda_l^2}{(1+\lambda_l)^2} b_{l1}^2 \sum_{l=1}^{2} \lambda_l b_{l1}^2,$$

so the property is true for $L = 2$, where "=" holds when any one of b_{11} and b_{21} is equal to 0 (note that b_{11} and b_{21} cannot be zero at the same time since $\mathbf{B}$ is invertible) or when $\lambda_1 = \lambda_2$.

- Inductive step: assume that the property is true for $L = P$, i.e.,

$$\sum_{l=1}^{P} \frac{\lambda_l^3}{(1+\lambda_l)^2} b_{l1}^2 \sum_{l=1}^{P} b_{l1}^2 \geq \sum_{l=1}^{P} \frac{\lambda_l^2}{(1+\lambda_l)^2} b_{l1}^2 \sum_{l=1}^{P} \lambda_l b_{l1}^2.$$

We must prove that it is also true for $L = P + 1$. As a matter of fact,

$$\sum_{l=1}^{P+1} \frac{\lambda_l^3}{(1+\lambda_l)^2} b_{l1}^2 \sum_{l=1}^{P+1} b_{l1}^2 = \left[\sum_{l=1}^{P} \frac{\lambda_l^3}{(1+\lambda_l)^2} b_{l1}^2 + \frac{\lambda_{P+1}^3}{(1+\lambda_{P+1})^2} b_{P+11}^2 \right] \times \left[\sum_{l=1}^{P} b_{l1}^2 + b_{P+11}^2 \right]$$
$$= \left[\sum_{l=1}^{P} \frac{\lambda_l^3}{(1+\lambda_l)^2} b_{l1}^2 \right] \left[\sum_{l=1}^{P} b_{l1}^2 \right] + \frac{\lambda_{P+1}^3}{(1+\lambda_{P+1})^2} b_{P+11}^4 + \sum_{l=1}^{P} \left[\frac{\lambda_l^3}{(1+\lambda_l)^2} + \frac{\lambda_{P+1}^3}{(1+\lambda_{P+1})^2} \right] b_{l1}^2 b_{P+11}^2.$$

Using the induction hypothesis, and also the fact that

$$\frac{\lambda_l^3}{(1+\lambda_l)^2} + \frac{\lambda_{P+1}^3}{(1+\lambda_{P+1})^2} \geq \frac{\lambda_l^2 \lambda_{P+1}}{(1+\lambda_l)^2} + \frac{\lambda_l \lambda_{P+1}^2}{(1+\lambda_{P+1})^2},$$

we get

$$\sum_{l=1}^{P+1} \frac{\lambda_l^3}{(1+\lambda_l)^2} b_{l1}^2 \sum_{l=1}^{P+1} b_{l1}^2 \geq \sum_{l=1}^{P} \frac{\lambda_l^2}{(1+\lambda_l)^2} b_{l1}^2 \sum_{l=1}^{P} \lambda_l b_{l1}^2 + \frac{\lambda_{P+1}^3}{(1+\lambda_{P+1})^2} b_{P+11}^4 + \sum_{l=1}^{P} \left[\frac{\lambda_l^2 \lambda_{P+1}}{(1+\lambda_l)^2} + \frac{\lambda_l \lambda_{P+1}^2}{(1+\lambda_{P+1})^2} \right] b_{l1}^2 b_{P+11}^2$$
$$= \sum_{l=1}^{P+1} \frac{\lambda_l^2}{(1+\lambda_l)^2} b_{l1}^2 \sum_{l=1}^{P+1} \lambda_l b_{l1}^2,$$

where "=" holds when all the λ_l's corresponding to the nonzeroes b_{l1} are equal, with $l = 1, 2, \ldots, P+1$. That completes the proof.

Even though it can improve the SNR, the Wiener filter does not maximize the output SNR. As a matter of fact, (2.37) is the well-known generalized Rayleigh quotient. So the filter that maximizes the output SNR is the eigenvector corresponding to the maximum eigenvalue of the matrix $\mathbf{R}_{vv}^{-1}\mathbf{R}_{xx}$ (see Section 2.5). However, this filter typically gives rise to large speech distortion.

The more general multichannel Wiener filter for noise reduction is studied in Chapter 5.

2.3 Frost Filter

The linearly constrained minimum variance (LCMV) filter [76], that we will also call the Frost filter, can be seen as a particular form of the Wiener filter.

2.3.1 Algorithm

In many practical situations, we do not have access to the reference signal and sometimes this reference does not even exist. As a result, the error signal as defined in (2.2) is meaningless.

If we consider the reference signal $x(k)$ to be zero, the MSE criterion [eq. (2.4)] becomes

$$J(\mathbf{h}) = \mathbf{h}^T \mathbf{R}_{yy} \mathbf{h}, \tag{2.41}$$

and the minimization of $J(\mathbf{h})$ with respect to $\mathbf{h}$ leads to the obvious solution $\mathbf{h} = \mathbf{0}_{L\times 1}$. Fortunately in many applications, constraints on the filter $\mathbf{h}$ that have the following form

$$\mathbf{C}^T \mathbf{h} = \mathbf{u} \tag{2.42}$$

are available, where $\mathbf{C}$ is the constraint matrix of size $L \times L_{\mathrm{c}}$ and

$$\mathbf{u} = \begin{bmatrix} u_0 \; u_1 \; \cdots \; u_{L_{\mathrm{c}}-1} \end{bmatrix}^T$$

is a vector of length L_{c} containing some chosen numbers.

This time to find the optimal filter, we need to solve the optimization problem

$$\min_{\mathbf{h}} J(\mathbf{h}) \quad \text{subject to} \quad \mathbf{C}^T \mathbf{h} = \mathbf{u}. \tag{2.43}$$

Using Lagrange multipliers to adjoin the constraints to the cost function, we easily find the Frost filter [76]

$$\mathbf{h}_{\mathrm{F}} = \mathbf{R}_{yy}^{-1} \mathbf{C} \left(\mathbf{C}^T \mathbf{R}_{yy}^{-1} \mathbf{C} \right)^{-1} \mathbf{u}. \tag{2.44}$$

It is important to observe that, in order for this filter to exist, the correlation matrix $\mathbf{R}_{yy}$ must be invertible and $\mathbf{C}$ must have full column rank, which implies that $L_{\mathrm{c}} \leq L$. In the rest, we assume that the rank of $\mathbf{C}$ is equal to L_{c}. The solution for the particular case of $L_{\mathrm{c}} = L$ is directly obtained from (2.42): $\mathbf{h}_{\mathrm{F}} = \left(\mathbf{C}^T\right)^{-1} \mathbf{u}$, which does not depend on the observation signal anymore. For the case $L_{\mathrm{c}} = 1$, the constraint matrix $\mathbf{C}$ becomes a constraint vector $\mathbf{c}$ and the solution has a similar form to the minimum variance distortionless response (MVDR) filter [35], [149]:

$$\mathbf{h}_{\mathrm{F}} = u_0 \frac{\mathbf{R}_{yy}^{-1}\mathbf{c}}{\mathbf{c}^T\mathbf{R}_{yy}^{-1}\mathbf{c}}. \tag{2.45}$$

2.3.2 Generalized Sidelobe Canceller Structure

The generalized sidelobe canceller (GSC) structure solves exactly the same problem as the LCMV approach by dividing the filter vector $\mathbf{h}_{\mathrm{F}}$ into two components operating on orthogonal subspaces [31], [54], [94], [230]:

$$\mathbf{h}_{\mathrm{F}} = \mathbf{f} - \mathbf{B}_{\mathrm{c}}\mathbf{w}_{\mathrm{GSC}}, \tag{2.46}$$

where

$$\mathbf{f} = \mathbf{C}\left(\mathbf{C}^T\mathbf{C}\right)^{-1}\mathbf{u} \tag{2.47}$$

is the minimum-norm solution of $\mathbf{C}^T\mathbf{f} = \mathbf{u}$, $\mathbf{B}_{\mathrm{c}}$ is the so-called blocking matrix that spans the nullspace of $\mathbf{C}^T$, i.e.,

$$\mathbf{C}^T\mathbf{B}_{\mathrm{c}} = \mathbf{0}_{L_{\mathrm{c}}\times(L-L_{\mathrm{c}})}, \tag{2.48}$$

and $\mathbf{w}_{\mathrm{GSC}}$ is a weighting vector derived as explained below. The size of $\mathbf{B}_{\mathrm{c}}$ is $L \times (L - L_{\mathrm{c}})$, where $L - L_{\mathrm{c}}$ is the dimension of the nullspace of $\mathbf{C}^T$. Therefore, the length of the vector $\mathbf{w}_{\mathrm{GSC}}$ is $L - L_{\mathrm{c}}$. The blocking matrix is not unique and the most obvious choice is the following:

$$\mathbf{B}_{\mathrm{c}} = \begin{bmatrix} \mathbf{I}_{(L-L_{\mathrm{c}})\times(L-L_{\mathrm{c}})} \\ \mathbf{0}_{L_{\mathrm{c}}\times(L-L_{\mathrm{c}})} \end{bmatrix} - \mathbf{C}\left(\mathbf{C}^T\mathbf{C}\right)^{-1}\mathbf{C}^T \begin{bmatrix} \mathbf{I}_{(L-L_{\mathrm{c}})\times(L-L_{\mathrm{c}})} \\ \mathbf{0}_{L_{\mathrm{c}}\times(L-L_{\mathrm{c}})} \end{bmatrix}. \tag{2.49}$$

To obtain the filter $\mathbf{w}_{\mathrm{GSC}}$, the GSC approach is used, which is formulated as the following unconstrained optimization problem

$$\min_{\mathbf{w}} \left(\mathbf{f} - \mathbf{B}_{\mathrm{c}}\mathbf{w}\right)^T \mathbf{R}_{yy} \left(\mathbf{f} - \mathbf{B}_{\mathrm{c}}\mathbf{w}\right), \tag{2.50}$$

and the solution is

$$\mathbf{w}_{\mathrm{GSC}} = \left(\mathbf{B}_{\mathrm{c}}^T\mathbf{R}_{yy}\mathbf{B}_{\mathrm{c}}\right)^{-1}\mathbf{B}_{\mathrm{c}}^T\mathbf{R}_{yy}\mathbf{f}. \tag{2.51}$$

Define the error signal between the outputs of the two filters $\mathbf{f}$ and $\mathbf{B}_{\mathrm{c}}\mathbf{w}$:

$$e(k) = \mathbf{y}^T(k)\mathbf{f} - \mathbf{y}^T(k)\mathbf{B}_{\mathrm{c}}\mathbf{w}, \tag{2.52}$$

it is easy to see that the minimization of $E\left[e^2(k)\right]$ with respect to $\mathbf{w}$ is equivalent to (2.50).

Now we need to check if indeed the two filters LCMV and GSC are equivalent, i.e.

$$\mathbf{u}^T \left(\mathbf{C}^T\mathbf{R}_{yy}^{-1}\mathbf{C}\right)^{-1}\mathbf{C}^T\mathbf{R}_{yy}^{-1} = \mathbf{f}^T\left[\mathbf{I} - \mathbf{R}_{yy}\mathbf{B}_{\mathrm{c}}\left(\mathbf{B}_{\mathrm{c}}^T\mathbf{R}_{yy}\mathbf{B}_{\mathrm{c}}\right)^{-1}\mathbf{B}_{\mathrm{c}}^T\right]. \tag{2.53}$$

For that, we are going to follow the elegant proof given in [28].

The matrix in brackets on the right-hand side of (2.53) can be rewritten as

$$\left[\mathbf{I} - \mathbf{R}_{yy}\mathbf{B}_{\mathrm{c}}\left(\mathbf{B}_{\mathrm{c}}^T\mathbf{R}_{yy}\mathbf{B}_{\mathrm{c}}\right)^{-1}\mathbf{B}_{\mathrm{c}}^T\right] = \mathbf{R}_{yy}^{1/2}\left(\mathbf{I} - \mathbf{P}_1\right)\mathbf{R}_{yy}^{-1/2}, \tag{2.54}$$

where

$$\mathbf{P}_1 = \mathbf{R}_{yy}^{1/2}\mathbf{B}_{\mathrm{c}}\left(\mathbf{B}_{\mathrm{c}}^T\mathbf{R}_{yy}\mathbf{B}_{\mathrm{c}}\right)^{-1}\mathbf{B}_{\mathrm{c}}^T\mathbf{R}_{yy}^{1/2} \tag{2.55}$$

is a projection operator onto the subspace spanned by the columns of $\mathbf{R}_{yy}^{1/2}\mathbf{B}_{\mathrm{c}}$. We have $\mathbf{B}_{\mathrm{c}}^T\mathbf{C} = \mathbf{B}_{\mathrm{c}}^T\mathbf{R}_{yy}^{1/2}\mathbf{R}_{yy}^{-1/2}\mathbf{C} = \mathbf{0}_{(L-L_{\mathrm{c}})\times L_{\mathrm{c}}}$. This implies that the rows of $\mathbf{B}_{\mathrm{c}}^T$ are orthogonal to the columns of $\mathbf{C}$ and the subspace spanned by the columns of $\mathbf{R}_{yy}^{1/2}\mathbf{B}_{\mathrm{c}}$ is orthogonal to the subspace spanned by the columns of $\mathbf{R}_{yy}^{-1/2}\mathbf{C}$. Since $\mathbf{B}_{\mathrm{c}}$ has a rank equal to $L - L_{\mathrm{c}}$ where L_{c} is the rank of $\mathbf{C}$, then the sum of the dimensions of the two subspaces is L and the subspaces are complementary. This means

$$\mathbf{P}_1 + \mathbf{P}_2 = \mathbf{I}, \tag{2.56}$$

where

$$\mathbf{P}_2 = \mathbf{R}_{yy}^{-1/2}\mathbf{C}\left(\mathbf{C}^T\mathbf{R}_{yy}^{-1}\mathbf{C}\right)^{-1}\mathbf{C}^T\mathbf{R}_{yy}^{-1/2}. \tag{2.57}$$

When this is substituted and the constraint $\mathbf{u}^T = \mathbf{f}^T\mathbf{C}$ is applied, (2.53) becomes

$$\begin{aligned}\mathbf{u}^T\left(\mathbf{C}^T\mathbf{R}_{yy}^{-1}\mathbf{C}\right)^{-1}\mathbf{C}^T\mathbf{R}_{yy}^{-1} &= \mathbf{f}^T\mathbf{R}_{yy}^{1/2}\mathbf{P}_2\mathbf{R}_{yy}^{-1/2}\\ &= \mathbf{f}^T\mathbf{R}_{yy}^{1/2}\left(\mathbf{I} - \mathbf{P}_1\right)\mathbf{R}_{yy}^{-1/2}\\ &= \mathbf{f}^T\left[\mathbf{I} - \mathbf{R}_{yy}\mathbf{B}_{\mathrm{c}}\left(\mathbf{B}_{\mathrm{c}}^T\mathbf{R}_{yy}\mathbf{B}_{\mathrm{c}}\right)^{-1}\mathbf{B}_{\mathrm{c}}^T\right].\end{aligned} \tag{2.58}$$

Hence, the LCMV and GSC filters are strictly equivalent.

2.3.3 Application to Linear Interpolation

In this subsection, the link between linear interpolation and the Frost filter is explained.

Linear interpolation is a straightforward generalization of forward and backward linear predictions. Indeed, in linear interpolation, we try to predict the value of the sample $y(k-i)$ from its past and future values [140], [183]. We define the interpolation error as

$$\begin{aligned} e_i(k) &= y(k-i) - \hat{y}(k-i) \\ &= y(k-i) - \sum_{l=0, l\neq i}^{L-1} h_{i,l} y(k-l) \\ &= \mathbf{h}_i^T \mathbf{y}(k),\ i = 0, 1, \ldots, L-1, \end{aligned} \tag{2.59}$$

where $\hat{y}(k-i)$ is the interpolated sample, and

$$\mathbf{h}_i = \left[-h_{i,0}\ -h_{i,1} \cdots h_{i,i} \cdots -h_{i,L-1} \right]^T$$

is a vector of length L containing the interpolation coefficients, with $h_{i,i} = 1$. The special cases $i = 0$ and $i = L-1$ correspond to the forward and backward prediction errors, respectively.

To find the optimal interpolator, we need to minimize the cost function

$$\begin{aligned} J_i(\mathbf{h}_i) &= E\left[e_i^2(k)\right] \\ &= \mathbf{h}_i^T \mathbf{R}_{yy} \mathbf{h}_i, \end{aligned} \tag{2.60}$$

subject to the constraint

$$\mathbf{c}_i^T \mathbf{h}_i = h_{i,i} = 1, \tag{2.61}$$

where

$$\mathbf{c}_i = \left[0\ 0 \cdots 0\ 1\ 0 \cdots 0 \right]^T$$

is the constraint vector of length L with its $(i+1)$th component equal to one and all others are equal to zero. By using a Lagrange multiplier, it is easy to see that the solution to this optimization problem is

$$\mathbf{R}_{yy} \mathbf{h}_{\mathrm{o},i} = E_i \mathbf{c}_i, \tag{2.62}$$

where

$$\begin{aligned} E_i &= \mathbf{h}_{\mathrm{o},i}^T \mathbf{R}_{yy} \mathbf{h}_{\mathrm{o},i} \\ &= \frac{1}{\mathbf{c}_i^T \mathbf{R}_{yy}^{-1} \mathbf{c}_i} \end{aligned} \tag{2.63}$$

is the interpolation error power. Hence

$$\mathbf{h}_{\mathrm{o},i} = \frac{\mathbf{R}_{yy}^{-1}\mathbf{c}_i}{\mathbf{c}_i^T\mathbf{R}_{yy}^{-1}\mathbf{c}_i}. \tag{2.64}$$

Comparing (2.64) with (2.44), it is clear that the optimal interpolator is a particular case of the Frost filter.

From (2.62) we find

$$\frac{\mathbf{h}_{\mathrm{o},i}}{E_i} = \mathbf{R}_{yy}^{-1}\mathbf{c}_i, \tag{2.65}$$

hence the $(i+1)$th column of $\mathbf{R}_{yy}^{-1}$ is $\mathbf{h}_{\mathrm{o},i}/E_i$. We can now see that $\mathbf{R}_{yy}^{-1}$ can be factorized as follows [12]:

$$\begin{aligned}\mathbf{R}_{yy}^{-1} &= \begin{bmatrix} 1 & -h_{\mathrm{o},1,0} & \cdots & -h_{\mathrm{o},L-1,0} \\ -h_{\mathrm{o},0,1} & 1 & \cdots & -h_{\mathrm{o},L-1,1} \\ \vdots & \vdots & \ddots & \vdots \\ -h_{\mathrm{o},0,L-1} & -h_{\mathrm{o},1,L-1} & \cdots & 1 \end{bmatrix} \begin{bmatrix} 1/E_0 & 0 & \cdots & 0 \\ 0 & 1/E_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1/E_{L-1} \end{bmatrix} \\ &= \mathbf{H}_{\mathrm{o}}^T\mathbf{D}_{\mathrm{e}}^{-1}. \end{aligned} \tag{2.66}$$

Furthermore, since $\mathbf{R}_{yy}^{-1}$ is a symmetric matrix, (2.66) can be written as

$$\begin{aligned}\mathbf{R}_{yy}^{-1} &= \begin{bmatrix} 1/E_0 & 0 & \cdots & 0 \\ 0 & 1/E_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1/E_{L-1} \end{bmatrix} \begin{bmatrix} 1 & -h_{\mathrm{o},0,1} & \cdots & -h_{\mathrm{o},0,L-1} \\ -h_{\mathrm{o},1,0} & 1 & \cdots & -h_{\mathrm{o},1,L-1} \\ \vdots & \vdots & \ddots & \vdots \\ -h_{\mathrm{o},L-1,0} & -h_{\mathrm{o},L-1,1} & \cdots & 1 \end{bmatrix} \\ &= \mathbf{D}_{\mathrm{e}}^{-1}\mathbf{H}_{\mathrm{o}}. \end{aligned} \tag{2.67}$$

Therefore, we deduce that

$$\frac{h_{\mathrm{o},i,l}}{E_i} = \frac{h_{\mathrm{o},l,i}}{E_l},\ i,l = 0,1,\ldots,L. \tag{2.68}$$

The first and last columns of $\mathbf{R}_{yy}^{-1}$ contain respectively the normalized forward and backward predictors and all the columns between contain the normalized interpolators.

We are now going to show how the condition number of the correlation matrix depends on the interpolators. The condition number of the matrix $\mathbf{R}_{yy}$ is defined as [89]

$$\chi\left[\mathbf{R}_{yy}\right] = \left\|\mathbf{R}_{yy}\right\| \left\|\mathbf{R}_{yy}^{-1}\right\|, \tag{2.69}$$

where $\|\cdot\|$ can be any matrix norm. Note that $\chi\left[\mathbf{R}_{yy}\right]$ depends on the underlying norm. Let us compute $\chi\left[\mathbf{R}_{yy}\right]$ using the Frobenius norm

$$\begin{aligned}\|\mathbf{R}_{yy}\|_{\mathrm{F}} &= \left\{\mathrm{tr}\left[\mathbf{R}_{yy}^T\mathbf{R}_{yy}\right]\right\}^{1/2} \\ &= \left\{\mathrm{tr}\left[\mathbf{R}_{yy}^2\right]\right\}^{1/2},\end{aligned} \tag{2.70}$$

and

$$\left\|\mathbf{R}_{yy}^{-1}\right\|_{\mathrm{F}} = \left\{\mathrm{tr}\left[\mathbf{R}_{yy}^{-2}\right]\right\}^{1/2}. \tag{2.71}$$

From (2.65), we have

$$\frac{\mathbf{h}_{\mathrm{o},i}^T\mathbf{h}_{\mathrm{o},i}}{E_i^2} = \mathbf{c}_i^T\mathbf{R}_{yy}^{-2}\mathbf{c}_i, \tag{2.72}$$

which implies that

$$\begin{aligned}\sum_{i=0}^{L-1}\frac{\mathbf{h}_{\mathrm{o},i}^T\mathbf{h}_{\mathrm{o},i}}{E_i^2} &= \sum_{i=0}^{L-1}\mathbf{c}_i^T\mathbf{R}_{yy}^{-2}\mathbf{c}_i \\ &= \mathrm{tr}\left[\mathbf{R}_{yy}^{-2}\right].\end{aligned} \tag{2.73}$$

Also, we can easily check that

$$\mathrm{tr}\left[\mathbf{R}_{yy}^2\right] = Lr_{yy}^2(0) + 2\sum_{l=1}^{L-1}(L-l)r_{yy}^2(l), \tag{2.74}$$

where $r_{yy}(l)$, $l = 0, 1, \ldots, L-1$, are the elements of the Toeplitz matrix $\mathbf{R}_{yy}$. Therefore, the square of the condition number of the correlation matrix associated with the Frobenius norm is

$$\chi_{\mathrm{F}}^2\left[\mathbf{R}_{yy}\right] = \left[Lr_{yy}^2(0) + 2\sum_{l=1}^{L-1}(L-l)r_{yy}^2(l)\right]\sum_{i=0}^{L-1}\frac{\mathbf{h}_{\mathrm{o},i}^T\mathbf{h}_{\mathrm{o},i}}{E_i^2}. \tag{2.75}$$

Some other interesting relations between the forward predictors and the condition number can be found in [17].

The LCMV filter for noise reduction and speech dereverberation is studied in Chapters 4, 5, and 7.

2.4 Kalman Filter

The Kalman filter [137], [138], [139] is a natural generalization of the Wiener filter [234] for nonstationary signals. The Kalman filter is also a sequential (recursive) MMSE estimator of a signal embedded in noise, where the signal is characterized by a state model.

We consider the observation signal

$$\begin{aligned} y(k) &= x(k) + v(k) \\ &= \mathbf{h}_1^T \mathbf{x}(k) + v(k), \end{aligned} \tag{2.76}$$

where $\mathbf{h}_1$ is defined in (2.9), $\mathbf{x}(k)$ is the state vector of length L, $v(k)$ is a zero-mean white Gaussian noise, and $\sigma_v^2(k) = E\left[v^2(k)\right]$. Note that now, the variance of the noise is allowed to vary with time.

We assume that the speech signal can be expressed as

$$x(k) = \sum_{l=1}^{L} a_l x(k-l) + v_x(k), \tag{2.77}$$

where a_l, $l = 1, 2, \ldots, L$, can be seen as the prediction coefficients of the signal $x(k)$, $v_x(k)$ is a zero-mean white Gaussian noise uncorrelated with $v(k)$, and $\sigma_{v_x}^2(k) = E\left[v_x^2(k)\right]$. Equation (2.77) is called the state model.

Using the state vector, we can rewrite (2.77) as

$$\mathbf{x}(k) = \mathbf{A}\mathbf{x}(k-1) + v_x(k)\mathbf{h}_1, \tag{2.78}$$

where

$$\mathbf{A} = \begin{bmatrix} a_1 & a_2 & \cdots & a_{L-1} & a_L \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix} \tag{2.79}$$

is the $L \times L$ (nonsingular) state transition matrix.

Given the equations

$$\mathbf{x}(k) = \mathbf{A}\mathbf{x}(k-1) + v_x(k)\mathbf{h}_1, \tag{2.80}$$

$$y(k) = \mathbf{h}_1^T \mathbf{x}(k) + v(k), \tag{2.81}$$

and assuming that $\mathbf{A}$, $\sigma_v^2(k)$, and $\sigma_{v_x}^2(k)$ are known, the objective of the Kalman filter is to find the optimal linear MMSE of $x(k)$. This can be done in two steps. In the following, we will borrow the nice and intuitive approach given in [102] to derive the Kalman filter.

Let $\hat{\mathbf{x}}(k|k-1)$ denote the linear MMSE estimator of $\mathbf{x}(k)$ at time k given the observations $y(1), y(2), \ldots, y(k-1)$. The corresponding state estimation error is

$$\mathbf{e}(k|k-1) = \mathbf{x}(k) - \hat{\mathbf{x}}(k|k-1), \tag{2.82}$$

and the error covariance matrix is

$$\mathbf{R}_{ee}(k|k-1) = E\left[\mathbf{e}(k|k-1)\mathbf{e}^T(k|k-1)\right]. \tag{2.83}$$

In the first step, no new observation is used. We would like to predict $\mathbf{x}(k)$ using the state equation (2.80). Since no new information is available, the best possible predictor is

$$\hat{\mathbf{x}}(k|k-1) = \mathbf{A}\hat{\mathbf{x}}(k-1|k-1). \tag{2.84}$$

The estimation error is

$$\begin{aligned}\mathbf{e}(k|k-1) &= \mathbf{x}(k) - \hat{\mathbf{x}}(k|k-1) \\ &= \mathbf{A}\mathbf{x}(k-1) + v_x(k)\mathbf{h}_1 - \mathbf{A}\hat{\mathbf{x}}(k-1|k-1) \\ &= \mathbf{A}\mathbf{e}(k-1|k-1) + v_x(k)\mathbf{h}_1. \end{aligned} \tag{2.85}$$

If we require that $E\left[\mathbf{e}(k-1|k-1)\right] = \mathbf{0}_{L\times 1}$ (this zero-mean condition states that there is no constant bias in the optimal linear estimation [82]), then $E\left[\mathbf{e}(k|k-1)\right] = \mathbf{0}_{L\times 1}$. Since $\mathbf{e}(k-1|k-1)$ is uncorrelated with $v_x(k)$, then

$$\mathbf{R}_{ee}(k|k-1) = \mathbf{A}\mathbf{R}_{ee}(k-1|k-1)\mathbf{A}^T + \sigma^2_{v_x}(k)\mathbf{h}_1\mathbf{h}_1^T. \tag{2.86}$$

This is the Riccati equation.

In the second step, the new observation, $y(k)$, is incorporated to estimate $\mathbf{x}(k)$. A linear estimate that is based on $\hat{\mathbf{x}}(k|k-1)$ and $y(k)$ has the form

$$\hat{\mathbf{x}}(k|k) = \mathbf{K}'(k)\hat{\mathbf{x}}(k|k-1) + \mathbf{k}(k)y(k), \tag{2.87}$$

where $\mathbf{K}'(k)$ and $\mathbf{k}(k)$ are some matrix and vector to be determined. The vector $\mathbf{k}(k)$ is called the Kalman gain. Now, the estimation error is

$$\begin{aligned}\mathbf{e}(k|k) &= \mathbf{x}(k) - \hat{\mathbf{x}}(k|k) \\ &= \mathbf{x}(k) - \mathbf{K}'(k)\hat{\mathbf{x}}(k|k-1) - \mathbf{k}(k)y(k) \\ &= \mathbf{x}(k) - \mathbf{K}'(k)\left[\mathbf{x}(k) - \mathbf{e}(k|k-1)\right] - \\ &\qquad \mathbf{k}(k)\left[\mathbf{h}_1^T\mathbf{x}(k) + v(k)\right] \\ &= \left[\mathbf{I} - \mathbf{K}'(k) - \mathbf{k}(k)\mathbf{h}_1^T\right]\mathbf{x}(k) + \\ &\qquad \mathbf{K}'(k)\mathbf{e}(k|k-1) - \mathbf{k}(k)v(k). \end{aligned} \tag{2.88}$$

Since $E\left[\mathbf{e}(k|k-1)\right] = \mathbf{0}_{L\times 1}$, then $E\left[\mathbf{e}(k|k)\right] = \mathbf{0}_{L\times 1}$ only if

$$\mathbf{K}'(k) = \mathbf{I} - \mathbf{k}(k)\mathbf{h}_1^T. \tag{2.89}$$

With this constraint, it follows that

$$\begin{aligned}\hat{\mathbf{x}}(k|k) &= \left[\mathbf{I} - \mathbf{k}(k)\mathbf{h}_1^T\right]\hat{\mathbf{x}}(k|k-1) + \mathbf{k}(k)y(k) \\ &= \hat{\mathbf{x}}(k|k-1) + \mathbf{k}(k)\left[y(k) - \mathbf{h}_1^T\hat{\mathbf{x}}(k|k-1)\right], \end{aligned} \tag{2.90}$$

and

$$\begin{aligned}\mathbf{e}(k|k) &= \mathbf{K}'(k)\mathbf{e}(k|k-1) - \mathbf{k}(k)v(k) \\ &= \left[\mathbf{I} - \mathbf{k}(k)\mathbf{h}_1^T\right]\mathbf{e}(k|k-1) - \mathbf{k}(k)v(k). \end{aligned} \tag{2.91}$$

Since $v(k)$ is uncorrelated with $v_x(k)$ and with $y(k-1)$, then $v(k)$ will be uncorrelated with $\mathbf{x}(k)$ and with $\hat{\mathbf{x}}(k|k-1)$; as a result $E\left[\mathbf{e}(k|k-1)v(k)\right] = \mathbf{0}_{L\times 1}$. Therefore, the error covariance matrix for $\mathbf{e}(k|k)$ is

$$\begin{aligned}\mathbf{R}_{ee}(k|k) &= E\left[\mathbf{e}(k|k)\mathbf{e}^T(k|k)\right] \\ &= \left[\mathbf{I} - \mathbf{k}(k)\mathbf{h}_1^T\right]\mathbf{R}_{ee}(k|k-1)\left[\mathbf{I} - \mathbf{k}(k)\mathbf{h}_1^T\right]^T + \\ &\quad \sigma_v^2(k)\mathbf{k}(k)\mathbf{k}^T(k). \end{aligned} \tag{2.92}$$

The final task is to find the Kalman gain vector, $\mathbf{k}(k)$, that minimizes the MSE

$$J(k) = \mathrm{tr}\left[\mathbf{R}_{ee}(k|k)\right]. \tag{2.93}$$

Differentiating $J(k)$ with respect to $\mathbf{k}(k)$, we get

$$\frac{\partial J(k)}{\partial \mathbf{k}(k)} = -2\left[\mathbf{I} - \mathbf{k}(k)\mathbf{h}_1^T\right]\mathbf{R}_{ee}(k|k-1)\mathbf{h}_1 + 2\sigma_v^2(k)\mathbf{k}(k), \tag{2.94}$$

and equating it to zero, we deduce the Kalman gain

$$\mathbf{k}(k) = \frac{\mathbf{R}_{ee}(k|k-1)\mathbf{h}_1}{\mathbf{h}_1^T\mathbf{R}_{ee}(k|k-1)\mathbf{h}_1 + \sigma_v^2(k)}. \tag{2.95}$$

We may simplify the expression for the error covariance matrix as follows

$$\begin{aligned}\mathbf{R}_{ee}(k|k) = &\left[\mathbf{I} - \mathbf{k}(k)\mathbf{h}_1^T\right]\mathbf{R}_{ee}(k|k-1) - \\ &\left\{\left[\mathbf{I} - \mathbf{k}(k)\mathbf{h}_1^T\right]\mathbf{R}_{ee}(k|k-1)\mathbf{h}_1 + \sigma_v^2(k)\mathbf{k}(k)\right\}\mathbf{k}^T(k), \end{aligned} \tag{2.96}$$

where, thanks to (2.94), the second term in (2.96) is equal to zero. Hence

$$\mathbf{R}_{ee}(k|k) = \left[\mathbf{I} - \mathbf{k}(k)\mathbf{h}_1^T\right]\mathbf{R}_{ee}(k|k-1). \tag{2.97}$$

The following steps summarize the Kalman filter [102]:

- **State Equation:**

$$\mathbf{x}(k) = \mathbf{A}\mathbf{x}(k-1) + v_x(k)\mathbf{h}_1$$

- **Observation Equation:**

$$y(k) = \mathbf{h}_1^T\mathbf{x}(k) + v(k)$$

- **Initialization:**

$$\begin{aligned}\hat{\mathbf{x}}(0|0) &= E\left[\mathbf{x}(0)\right]\\ \mathbf{R}_{ee}(0|0) &= E\left[\mathbf{x}(0)\mathbf{x}^T(0)\right]\end{aligned}$$

- **Computation:** For $k = 1, 2, \ldots$

$$\begin{aligned}\hat{\mathbf{x}}(k|k-1) &= \mathbf{A}\hat{\mathbf{x}}(k-1|k-1)\\ \mathbf{R}_{ee}(k|k-1) &= \mathbf{A}\mathbf{R}_{ee}(k-1|k-1)\mathbf{A}^T + \sigma_{v_x}^2(k)\mathbf{h}_1\mathbf{h}_1^T\\ \mathbf{k}(k) &= \frac{\mathbf{R}_{ee}(k|k-1)\mathbf{h}_1}{\mathbf{h}_1^T\mathbf{R}_{ee}(k|k-1)\mathbf{h}_1 + \sigma_v^2(k)}\\ \hat{\mathbf{x}}(k|k) &= \hat{\mathbf{x}}(k|k-1) + \mathbf{k}(k)\left[y(k) - \mathbf{h}_1^T\hat{\mathbf{x}}(k|k-1)\right]\\ \mathbf{R}_{ee}(k|k) &= \left[\mathbf{I} - \mathbf{k}(k)\mathbf{h}_1^T\right]\mathbf{R}_{ee}(k|k-1)\end{aligned}$$

Finally, the estimate of the speech sample, $x(k)$, at time k with the Kalman filter would be

$$z_{\mathrm{K}}(k) = \mathbf{h}_1^T\hat{\mathbf{x}}(k|k). \tag{2.98}$$

By analogy with the Wiener filter, we define the speech-distortion index for the Kalman filter as

$$\begin{aligned}\upsilon_{\mathrm{sd}}(k) &= \frac{E\left\{\left[x(k) - \mathbf{h}_1^T\hat{\mathbf{x}}(k|k)\right]^2\right\}}{\sigma_x^2(k)}\\ &= \frac{\mathbf{h}_1^T\mathbf{R}_{ee}(k|k)\mathbf{h}_1}{\sigma_x^2(k)}.\end{aligned} \tag{2.99}$$

When the Kalman filter converges, $\mathbf{R}_{ee}(k|k)$ will become smaller and smaller and so will be $\upsilon_{\mathrm{sd}}(k)$. Clearly, the Kalman filter has the potential to cause much less distortion than the Wiener filter.

2.5 A Viable Alternative to the MSE

With the MSE formulation, many desired properties of the optimal filters such as the SNR behavior cannot be seen. In this section, we present a new criterion based on the Pearson correlation coefficient (PCC). We show that the squared PCC has many appealing properties and can be used as an optimization cost function. Similar to the MSE, we can derive the Wiener filter and many other optimal or suboptimal filters with this new cost function. The clear advantage of using the squared PCC over the MSE is that the performance (particularly for the output SNR) of the resulting optimal filters can be easily analyzed.

2.5.1 Pearson Correlation Coefficient

Let x and y be two zero-mean real-valued random variables. The Pearson correlation coefficient (PCC) is defined as[1] [64], [181], [191]

$$\rho(x,y) = \frac{E[xy]}{\sigma_x \sigma_y}, \tag{2.100}$$

where $E[xy]$ is the cross-correlation between x and y, and $\sigma_x^2 = E\left[x^2\right]$ and $\sigma_y^2 = E\left[y^2\right]$ are the variances of the signals x and y, respectively. In the rest, it will be more convenient to work with the squared Pearson correlation coefficient (SPCC):

$$\rho^2(x,y) = \frac{E^2[xy]}{\sigma_x^2 \sigma_y^2}. \tag{2.101}$$

One of the most important properties of the SPCC is that

$$0 \leq \rho^2(x,y) \leq 1. \tag{2.102}$$

The SPCC gives an indication on the strength of the linear relationship between the two random variables x and y. If $\rho^2(x,y) = 0$, then x and y are said to be uncorrelated. The closer the value of $\rho^2(x,y)$ is to 1, the stronger the correlation between the two variables. If the two variables are independent, then $\rho^2(x,y) = 0$. But the converse is not true because the SPCC detects only *linear* dependencies between the two variables x and y. For a non-linear dependency, the SPCC may be equal to zero. However, in the special case when x and y are jointly normal, "independent" is equivalent to "uncorrelated."

2.5.2 Important Relations with the SPCC

In this subsection, we discuss many interesting properties regarding SPCCs among the four signals x, v, y, and z.

The SPCC between $x(k)$ and $y(k)$ [as defined in (2.1)] is

$$\begin{aligned} \rho^2(x,y) &= \frac{\sigma_x^2}{\sigma_y^2} \\ &= \frac{\mathrm{SNR}}{1+\mathrm{SNR}}, \end{aligned} \tag{2.103}$$

where $\sigma_y^2 = E\left[y^2(k)\right] = \sigma_x^2 + \sigma_v^2$ is the variance of the signal $y(k)$.

The SPCC between $x(k)$ and $z(k)$ [as defined in (2.3)] is

[1] This correlation coefficient is named after Karl Pearson who described many of its properties.

$$\rho^2(x,z) = \frac{\left(\mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}\right)^2}{\sigma_x^2\left(\mathbf{h}^T\mathbf{R}_{yy}\mathbf{h}\right)} = \frac{\left(\mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}\right)^2}{\sigma_x^2\left(\mathbf{h}^T\mathbf{R}_{xx}\mathbf{h}\right)} \cdot \frac{\text{SNR}(\mathbf{h})}{1+\text{SNR}(\mathbf{h})}, \tag{2.104}$$

where

$$\text{SNR}(\mathbf{h}) = \frac{\mathbf{h}^T\mathbf{R}_{xx}\mathbf{h}}{\mathbf{h}^T\mathbf{R}_{vv}\mathbf{h}} \tag{2.105}$$

is the output SNR for the filter $\mathbf{h}$.

Property 1. We have

$$\rho^2(x,z) = \rho^2\left(x,\mathbf{h}^T\mathbf{y}\right) = \rho^2\left(x,\mathbf{h}^T\mathbf{x}\right)\rho^2\left(\mathbf{h}^T\mathbf{x},\mathbf{h}^T\mathbf{y}\right), \tag{2.106}$$

where

$$\rho^2\left(x,\mathbf{h}^T\mathbf{x}\right) = \frac{\left(\mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}\right)^2}{\sigma_x^2\left(\mathbf{h}^T\mathbf{R}_{xx}\mathbf{h}\right)}, \tag{2.107}$$

and

$$\rho^2\left(\mathbf{h}^T\mathbf{x},\mathbf{h}^T\mathbf{y}\right) = \frac{\text{SNR}(\mathbf{h})}{1+\text{SNR}(\mathbf{h})}. \tag{2.108}$$

The SPCC $\rho^2\left(x,\mathbf{h}^T\mathbf{x}\right)$ can be viewed as a speech-distortion index. If $\mathbf{h} = \mathbf{h}_1$ (no speech distortion) then $\rho^2\left(x,\mathbf{h}^T\mathbf{x}\right) = 1$. The closer the value of $\rho^2\left(x,\mathbf{h}^T\mathbf{x}\right)$ is to 0, the more distorted the speech signal (except for a simple delay filter). The SPCC $\rho^2\left(\mathbf{h}^T\mathbf{x},\mathbf{h}^T\mathbf{y}\right)$ shows the SNR improvement, so it can be viewed as a noise-reduction index that reaches its maximum when $\text{SNR}(\mathbf{h})$ is maximized.

Property 1 is fundamental in the noise-reduction problem. It shows that the SPCC $\rho^2\left(x,\mathbf{h}^T\mathbf{y}\right)$, which is a cost function as explained later, is simply the product of two important indices reflecting noise reduction and speech distortion. In contrast, the MSE has a much more complex form with no real physical meaning in the speech enhancement context.

Property 2. We have

$$\rho^2\left(x,\mathbf{h}^T\mathbf{y}\right) \leq \frac{\text{SNR}(\mathbf{h})}{1+\text{SNR}(\mathbf{h})}, \tag{2.109}$$

with equality when $\mathbf{h} = \mathbf{h}_1$.

Property 3. We have

$$\rho^2\left(\mathbf{h}^T\mathbf{x}, y\right) = \rho^2\left(x, \mathbf{h}^T\mathbf{x}\right)\rho^2\left(x, y\right). \tag{2.110}$$

Property 4. We have

$$\rho^2\left(\mathbf{h}^T\mathbf{x}, y\right) \leq \frac{\mathrm{SNR}}{1+\mathrm{SNR}}, \tag{2.111}$$

with equality when $\mathbf{h} = \mathbf{h}_1$.

The SPCC between $v(k)$ and $y(k)$ [as defined in (2.1)] is

$$\begin{aligned}\rho^2\left(v, y\right) &= \frac{\sigma_v^2}{\sigma_y^2} \\ &= \frac{1}{1+\mathrm{SNR}}.\end{aligned} \tag{2.112}$$

The SPCC between $v(k)$ and $z(k)$ [as defined in (2.3)] is

$$\begin{aligned}\rho^2\left(v, z\right) &= \frac{\left(\mathbf{h}_1^T\mathbf{R}_{vv}\mathbf{h}\right)^2}{\sigma_v^2\left(\mathbf{h}^T\mathbf{R}_{yy}\mathbf{h}\right)} \\ &= \frac{\left(\mathbf{h}_1^T\mathbf{R}_{vv}\mathbf{h}\right)^2}{\sigma_v^2\left(\mathbf{h}^T\mathbf{R}_{vv}\mathbf{h}\right)} \cdot \frac{1}{1+\mathrm{SNR}(\mathbf{h})}.\end{aligned} \tag{2.113}$$

Property 5. We have

$$\rho^2\left(v, z\right) = \rho^2\left(v, \mathbf{h}^T\mathbf{y}\right) = \rho^2\left(v, \mathbf{h}^T\mathbf{v}\right) \cdot \rho^2\left(\mathbf{h}^T\mathbf{v}, \mathbf{h}^T\mathbf{y}\right), \tag{2.114}$$

where

$$\rho^2\left(v, \mathbf{h}^T\mathbf{v}\right) = \frac{\left(\mathbf{h}_1^T\mathbf{R}_{vv}\mathbf{h}\right)^2}{\sigma_v^2\left(\mathbf{h}^T\mathbf{R}_{vv}\mathbf{h}\right)}, \tag{2.115}$$

and

$$\rho^2\left(\mathbf{h}^T\mathbf{v}, \mathbf{h}^T\mathbf{y}\right) = \frac{1}{1+\mathrm{SNR}(\mathbf{h})}. \tag{2.116}$$

Property 6. We have

$$\rho^2\left(v, \mathbf{h}^T\mathbf{y}\right) \leq \frac{1}{1+\text{SNR}(\mathbf{h})}, \tag{2.117}$$

with equality when $\mathbf{h} = \mathbf{h}_1$.

Property 7. We have

$$\rho^2\left(\mathbf{h}^T\mathbf{v}, y\right) = \rho^2\left(v, \mathbf{h}^T\mathbf{v}\right) \cdot \rho^2\left(v, y\right). \tag{2.118}$$

Property 8. We have

$$\rho^2\left(\mathbf{h}^T\mathbf{v}, y\right) \leq \frac{1}{1+\text{SNR}}, \tag{2.119}$$

with equality when $\mathbf{h} = \mathbf{h}_1$.

Property 9. We have

$$\text{SNR}(\mathbf{h}) = \frac{\rho^2\left(\mathbf{h}^T\mathbf{x}, \mathbf{h}^T\mathbf{y}\right)}{\rho^2\left(\mathbf{h}^T\mathbf{v}, \mathbf{h}^T\mathbf{y}\right)}. \tag{2.120}$$

In the next subsection, we will see that the positive quantity $\rho^2\left(x, \mathbf{h}^T\mathbf{y}\right)$ can serve as a criterion to derive different optimal filters. Many of the properties shown here are relevant and will better help us understand the fundamental role of the SPCC in the application of speech enhancement.

2.5.3 Examples of Optimal Filters Derived from the SPCC

Intuitively, the problem of estimating the signal $x(k)$ from the observation signal $y(k)$ can be formulated as one of finding the filter that maximizes the SPCC $\rho^2\left(x, \mathbf{h}^T\mathbf{y}\right)$ in order to make the clean speech signal, $x(k)$, and the filter output signal, $z(k)$, correlated as much as possible. Furthermore, since the SPCC $\rho^2\left(x, \mathbf{h}^T\mathbf{y}\right)$ is the product of two other SPCCs $\rho^2\left(x, \mathbf{h}^T\mathbf{x}\right)$ and $\rho^2\left(\mathbf{h}^T\mathbf{x}, \mathbf{h}^T\mathbf{y}\right)$ (see Property 1), we can find other forms of optimal filters that maximize either one of these two SPCCs with or without constraints.

Speech Distortionless Filter.

As explained in the previous subsection, the SPCC $\rho^2\left(x, \mathbf{h}^T\mathbf{x}\right)$ is a speech-distortion index. This term is maximum (and equal to one) if $\mathbf{h} = \mathbf{h}_1$. Therefore, maximizing $\rho^2\left(x, \mathbf{h}^T\mathbf{x}\right)$ will lead to the filter $\mathbf{h}_1$. In this case, we have

$$\mathrm{SNR}(\mathbf{h}_1) = \mathrm{SNR}, \tag{2.121}$$

$$\rho^2\left(x, \mathbf{h}_1^T\mathbf{x}\right) = 1, \tag{2.122}$$

$$z_1(k) = y(k). \tag{2.123}$$

The filter $\mathbf{h}_1$ has no impact neither on the clean signal nor on the noise. In other words, $\mathbf{h} = \mathbf{h}_1$ will not distort the clean signal but will not improve the output SNR either.

Maximum SNR Filter.

It is easy to see that maximizing $\rho^2\left(\mathbf{h}^T\mathbf{x}, \mathbf{h}^T\mathbf{y}\right)$ is equivalent to maximizing the output SNR, $\mathrm{SNR}(\mathbf{h})$, which is also equivalent to solving the generalized eigenvalue problem:

$$\mathbf{R}_{xx}\mathbf{h} = \lambda\mathbf{R}_{vv}\mathbf{h}. \tag{2.124}$$

Assuming that $\mathbf{R}_{vv}^{-1}$ exists, the optimal solution to our problem is the eigenvector, $\mathbf{h}_{\max}$, corresponding to the maximum eigenvalue, $\lambda_{\max}$, of $\mathbf{R}_{vv}^{-1}\mathbf{R}_{xx}$. Hence

$$\mathrm{SNR}(\mathbf{h}_{\max}) = \lambda_{\max}, \tag{2.125}$$

$$\rho^2\left(\mathbf{h}_{\max}^T\mathbf{x}, \mathbf{h}_{\max}^T\mathbf{y}\right) = \frac{\lambda_{\max}}{1+\lambda_{\max}}, \tag{2.126}$$

$$z_{\max}(k) = \mathbf{h}_{\max}^T\mathbf{y}(k). \tag{2.127}$$

From this filter, we can deduce another interesting property of the SPCC.

Property 10. We have

$$\rho^2\left(x, \mathbf{h}_{\max}^T\mathbf{x}\right) = \frac{\mathrm{SNR}(\mathbf{h}_{\max})}{\mathrm{SNR}} \cdot \rho^2\left(v, \mathbf{h}_{\max}^T\mathbf{v}\right). \tag{2.128}$$

Since $\mathrm{SNR}(\mathbf{h}_{\max}) \geq \mathrm{SNR}(\mathbf{h}_1) = \mathrm{SNR}$, this implies that

$$\rho^2\left(x, \mathbf{h}_{\max}^T\mathbf{x}\right) \geq \rho^2\left(v, \mathbf{h}_{\max}^T\mathbf{v}\right), \tag{2.129}$$

which means that the filter $\mathbf{h}_{\max}$ yields less distortion to the clean speech signal, $x(k)$, than to the noise signal, $v(k)$.

Wiener Filter.

We are going to maximize the SPCC $\rho^2\left(x, \mathbf{h}^T\mathbf{y}\right)$. Indeed, if we differentiate this term with respect to $\mathbf{h}$, equate the result to zero, and assume that the matrices $\mathbf{R}_{xx}$ and $\mathbf{R}_{yy}$ are full rank, we easily obtain

$$\mathbf{R}_{xx}\mathbf{h}_1\left(\mathbf{h}^T\mathbf{R}_{yy}\mathbf{h}\right) = \left(\mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}\right)\mathbf{R}_{yy}\mathbf{h}. \tag{2.130}$$

If we look for the optimal filter, $\mathbf{h}_{\mathrm{W}}$, that satisfies the relation

$$\mathbf{h}_{\mathrm{W}}^T\mathbf{R}_{yy}\mathbf{h}_{\mathrm{W}} = \mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}_{\mathrm{W}}, \tag{2.131}$$

we find that

$$\mathbf{h}_{\mathrm{W}} = \mathbf{R}_{yy}^{-1}\mathbf{R}_{xx}\mathbf{h}_1, \tag{2.132}$$

which is the classical Wiener filter [125] also given in (2.7). We can check that, indeed, $\mathbf{h}_{\mathrm{W}}$ as given in (2.132) satisfies the relation (2.131) as well as (2.130). For the Wiener filter, we have the following properties.

Property 11. Maximizing the SPCC $\rho^2\left(x, \mathbf{h}^T\mathbf{y}\right)$ is equivalent to maximizing the variance, $E\left[z^2(k)\right]$, of the filter output signal, $z(k)$, subject to the constraint $\mathbf{h}^T\mathbf{R}_{yy}\mathbf{h} = \mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}$.

Property 12. We have

$$\rho^2\left(x, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right) = \frac{1}{\xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right)} \cdot \frac{1 + \mathrm{SNR}(\mathbf{h}_{\mathrm{W}})}{\mathrm{SNR}}. \tag{2.133}$$

This implies that

$$\xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right) \geq \frac{1 + \mathrm{SNR}(\mathbf{h}_{\mathrm{W}})}{\mathrm{SNR}}. \tag{2.134}$$

But using Properties 2 and 12, we deduce a better lower bound:

$$\xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right) \geq \frac{\left[1 + \mathrm{SNR}(\mathbf{h}_{\mathrm{W}})\right]^2}{\mathrm{SNR} \cdot \mathrm{SNR}(\mathbf{h}_{\mathrm{W}})} \geq \frac{1 + \mathrm{SNR}(\mathbf{h}_{\mathrm{W}})}{\mathrm{SNR}}. \tag{2.135}$$

Property 13. (Identical to the Proposition given in Section 2.2.) With the optimal Wiener filter given in (2.132), the output SNR is always greater than or at least equal to the input SNR.

Proof. Let us evaluate the SPCC between $y(k)$ and $\mathbf{h}_{\mathrm{W}}^T\mathbf{y}(k)$:

$$\begin{aligned}
\rho^2\left(y, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right) &= \frac{\left(\mathbf{h}_1^T\mathbf{R}_{yy}\mathbf{h}_{\mathrm{W}}\right)^2}{\sigma_y^2\left(\mathbf{h}_{\mathrm{W}}^T\mathbf{R}_{yy}\mathbf{h}_{\mathrm{W}}\right)} \\
&= \frac{\sigma_x^2}{\sigma_y^2} \cdot \frac{\sigma_x^2}{\mathbf{h}_{\mathrm{W}}^T\mathbf{R}_{xx}\mathbf{h}_1} \\
&= \frac{\rho^2(x, y)}{\rho^2\left(x, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right)}.
\end{aligned} \tag{2.136}$$

Therefore

$$\rho^2(x,y) = \rho^2\left(y, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right) \cdot \rho^2\left(x, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right) \leq \rho^2\left(x, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right). \tag{2.137}$$

Using (2.103) and Property 2 in the previous expression, we get

$$\frac{\mathrm{SNR}}{1+\mathrm{SNR}} \leq \frac{\mathrm{SNR}(\mathbf{h}_{\mathrm{W}})}{1+\mathrm{SNR}(\mathbf{h}_{\mathrm{W}})}. \tag{2.138}$$

Slightly reorganizing (2.138) gives

$$\frac{1}{1+\dfrac{1}{\mathrm{SNR}}} \leq \frac{1}{1+\dfrac{1}{\mathrm{SNR}(\mathbf{h}_{\mathrm{W}})}}, \tag{2.139}$$

which implies that

$$\frac{1}{\mathrm{SNR}} \geq \frac{1}{\mathrm{SNR}(\mathbf{h}_{\mathrm{W}})}. \tag{2.140}$$

As a result

$$\mathrm{SNR}(\mathbf{h}_{\mathrm{W}}) \geq \mathrm{SNR}. \tag{2.141}$$

That completes the proof.

This proof is amazingly simple and much easier to follow than the proof given in Section 2.2.

Property 14. We have

$$\frac{\left[1+\mathrm{SNR}(\mathbf{h}_{\mathrm{W}})\right]^2}{\mathrm{SNR}\cdot\mathrm{SNR}(\mathbf{h}_{\mathrm{W}})} \leq \xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{W}}) \leq \frac{(1+\mathrm{SNR})\left[1+\mathrm{SNR}(\mathbf{h}_{\mathrm{W}})\right]}{\mathrm{SNR}^2}, \tag{2.142}$$

or

$$\frac{1}{\rho^2\left(\mathbf{h}_{\mathrm{W}}^T\mathbf{v}, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right)\cdot\rho^2\left(\mathbf{h}_{\mathrm{W}}^T\mathbf{x}, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right)} \leq \mathrm{SNR}\cdot\xi_{\mathrm{nr}}(\mathbf{h}_{\mathrm{W}}) \leq \frac{1}{\rho^2(x,y)\cdot\rho^2\left(\mathbf{h}_{\mathrm{W}}^T\mathbf{v}, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right)}. \tag{2.143}$$

Proof. For the lower bound, see (2.135). The upper bound is easy to show by using Property 12 and (2.137).

Property 15. We have

$$\upsilon_{\mathrm{sd}}(\mathbf{h}_{\mathrm{W}}) = 1 - \rho^2\left(x, \mathbf{h}_{\mathrm{W}}^T\mathbf{x}\right)\cdot\left\{1 - \frac{1}{\left[1+\mathrm{SNR}(\mathbf{h}_{\mathrm{W}})\right]^2}\right\}. \tag{2.144}$$

This expression shows the link between the speech-distortion index, $\upsilon_{\rm sd}\left(\mathbf{h}_{\rm W}\right)$, and the SPCC $\rho^2\left(x,\mathbf{h}_{\rm W}^T\mathbf{x}\right)$. When $\rho^2\left(x,\mathbf{h}_{\rm W}^T\mathbf{x}\right)$ is high (resp. low), $\upsilon_{\rm sd}\left(\mathbf{h}_{\rm W}\right)$ is small (resp. large) and, as a result, the clean speech signal is lowly (resp. highly) distorted. We also have

$$\rho^2\left(x,\mathbf{h}_{\rm W}^T\mathbf{x}\right)\geq\frac{\rm SNR}{1+\rm SNR}\cdot\frac{1+\rm SNR(\mathbf{h}_{\rm W})}{\rm SNR(\mathbf{h}_{\rm W})},\tag{2.145}$$

so when the output SNR increases, the lower bound of the SPCC $\rho^2\left(x,\mathbf{h}_{\rm W}^T\mathbf{x}\right)$ decreases; as a consequence, the distortion of the clean speech likely increases.

Now we discuss the connection between maximizing the SPCC and minimizing the MSE. The MSE is

$$\begin{aligned}J(\mathbf{h})&=E\left[e^2(k)\right]\\&=\sigma_x^2+\mathbf{h}^T\mathbf{R}_{yy}\mathbf{h}-2\mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}\\&=\sigma_x^2\left[1+\frac{1}{\xi_{\rm nr}\left(\mathbf{h}\right)}\cdot\frac{1+\rm SNR(\mathbf{h})}{\rm SNR}-2\frac{\mathbf{h}^T\mathbf{R}_{xx}\mathbf{h}}{\mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}}\cdot\rho^2\left(x,\mathbf{h}^T\mathbf{x}\right)\right].\end{aligned}\tag{2.146}$$

Property 16. We have

$$\tilde{J}(\mathbf{h}_{\rm W})=\rm SNR\left[1-\rho^2\left(x,\mathbf{h}_{\rm W}^T\mathbf{y}\right)\right],\tag{2.147}$$

where $\tilde{J}(\mathbf{h}_{\rm W})$ is the normalized MMSE defined in (2.16). Therefore, as expected, the MSE is minimized when the SPCC is maximized.

Proof. Equation (2.147) can be easily verified by using Property 12, relation (2.131), and Property 1 in (2.146).

Property 17. We have

$$\frac{\rm SNR}{1+\rm SNR(\mathbf{h}_{\rm W})}\leq\tilde{J}(\mathbf{h}_{\rm W})\leq\frac{\rm SNR}{1+\rm SNR},\tag{2.148}$$

or

$$\rho^2\left(\mathbf{h}_{\rm W}^T\mathbf{v},\mathbf{h}_{\rm W}^T\mathbf{y}\right)\leq\frac{\tilde{J}(\mathbf{h}_{\rm W})}{\rm SNR}\leq\rho^2\left(v,y\right).\tag{2.149}$$

Proof. These bounds can be proven by using the bounds of $\rho^2\left(x,\mathbf{h}_{\rm W}^T\mathbf{y}\right)$ and (2.147).

Property 18. We have

$$\upsilon_{\rm sd}\left(\mathbf{h}_{\rm W}\right)=\frac{1}{\rm SNR}\left[\tilde{J}(\mathbf{h}_{\rm W})-\frac{1}{\xi_{\rm nr}\left(\mathbf{h}_{\rm W}\right)}\right].\tag{2.150}$$

Proof. See [41].

Property 19. We have

$$\frac{1}{\left[1+\mathrm{SNR}(\mathbf{h}_{\mathrm{W}})\right]^{2}} \leq \upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right) \leq \frac{1+\mathrm{SNR}(\mathbf{h}_{\mathrm{W}})-\mathrm{SNR}}{(1+\mathrm{SNR})\left[1+\mathrm{SNR}(\mathbf{h}_{\mathrm{W}})\right]}, \quad (2.151)$$

or

$$\rho^{4}\left(\mathbf{h}_{\mathrm{W}}^{T}\mathbf{v}, \mathbf{h}_{\mathrm{W}}^{T}\mathbf{y}\right) \leq \upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right) \leq \rho^{2}\left(v, y\right) \cdot \rho^{2}\left(\mathbf{h}_{\mathrm{W}}^{T}\mathbf{v}, \mathbf{h}_{\mathrm{W}}^{T}\mathbf{y}\right) + \rho^{2}\left(v, y\right) - \rho^{2}\left(\mathbf{h}_{\mathrm{W}}^{T}\mathbf{v}, \mathbf{h}_{\mathrm{W}}^{T}\mathbf{y}\right). \quad (2.152)$$

Proof. These bounds can be proven by using Properties 14, 17, and 18.

Property 20. From the MSE perspective, with the Wiener filter

$$\mathrm{SNR}(\mathbf{h}_{\mathrm{W}}) \geq \mathrm{SNR} \iff \xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right) > 1,\ \upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right) < 1. \quad (2.153)$$

Therefore, the measures $\xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right)$ and $\upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right)$ may be good indicators of the behavior of the Wiener filter except for at least the case when $\mathrm{SNR}(\mathbf{h}_{\mathrm{W}}) = \mathrm{SNR}$. In this scenario

$$\xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right) = \frac{\left(1+\mathrm{SNR}\right)^{2}}{\mathrm{SNR}^{2}} > 1, \quad (2.154)$$

$$\upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right) = \frac{1}{\left(1+\mathrm{SNR}\right)^{2}} > 0, \quad (2.155)$$

$$\mathbf{h}_{\mathrm{W}} = \frac{\mathrm{SNR}}{1+\mathrm{SNR}}\mathbf{h}_{1}. \quad (2.156)$$

This situation occurs when the signal $x(k)$ is not predictable (white random signal). This particular case shows a slight anomaly in the definitions (2.19) and (2.20) since noise reduction and speech distortion are possible while the output SNR is not improved at all. This is due to the fact that

$$\xi_{\mathrm{nr}}\left(c \cdot \mathbf{h}_{\mathrm{W}}\right) \neq \xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right), \quad (2.157)$$

$$\upsilon_{\mathrm{sd}}\left(c \cdot \mathbf{h}_{\mathrm{W}}\right) \neq \upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right), \quad (2.158)$$

for a constant $c \neq 0$ and $c \neq 1$.

Property 21. From the SPCC perspective, with the Wiener filter

$$\mathrm{SNR}(\mathbf{h}_{\mathrm{W}}) \geq \mathrm{SNR} \iff \rho^{2}\left(\mathbf{h}_{\mathrm{W}}^{T}\mathbf{x}, \mathbf{h}_{\mathrm{W}}^{T}\mathbf{y}\right) \geq \rho^{2}\left(x, y\right),\ \rho^{2}\left(x, \mathbf{h}_{\mathrm{W}}^{T}\mathbf{x}\right) \leq 1. \quad (2.159)$$

When $\mathrm{SNR}(\mathbf{h}_{\mathrm{W}}) = \mathrm{SNR}$, then

$$\rho^2\left(\mathbf{h}_{\mathrm{W}}^T\mathbf{x}, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right) = \rho^2\left(x, y\right), \tag{2.160}$$

$$\rho^2\left(x, \mathbf{h}_{\mathrm{W}}^T\mathbf{x}\right) = 1. \tag{2.161}$$

This time, the measures based on the SPCCs $\rho^2\left(\mathbf{h}_{\mathrm{W}}^T\mathbf{x}, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right)$ and $\rho^2\left(x, \mathbf{h}_{\mathrm{W}}^T\mathbf{x}\right)$ reflect accurately the output SNR, since when this latter is not improved the speech-distortion index $\rho^2\left(x, \mathbf{h}_{\mathrm{W}}^T\mathbf{x}\right)$ says that there is no speech distortion and the noise-reduction index $\rho^2\left(\mathbf{h}_{\mathrm{W}}^T\mathbf{x}, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right)$ says that there is no noise reduction indeed. The anomaly discussed above no longer exists in the context of the SPCC thanks to the properties:

$$\rho^2\left(c\cdot\mathbf{h}_{\mathrm{W}}^T\mathbf{x}, c\cdot\mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right) = \rho^2\left(\mathbf{h}_{\mathrm{W}}^T\mathbf{x}, \mathbf{h}_{\mathrm{W}}^T\mathbf{y}\right), \tag{2.162}$$

$$\rho^2\left(x, c\cdot\mathbf{h}_{\mathrm{W}}^T\mathbf{x}\right) = \rho^2\left(x, \mathbf{h}_{\mathrm{W}}^T\mathbf{x}\right), \tag{2.163}$$

for a constant $c \neq 0$.

Properties 20 and 21 show basically that the noise-reduction factor, $\xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{W}}\right)$, and the speech-distortion index, $\upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right)$, derived from the MSE formulation present a slight anomaly compared to the equivalent measures based on the SPCCs and derived from an SPCC criterion.

Trade-Off Filters.

It is also possible to derive other optimal filters that can control the trade-off between speech distortion and SNR improvement. For example, it can be more attractive to find a filter that minimizes the speech distortion while it guaranties a certain level of SNR improvement. Mathematically, this optimization problem can be written as follows:

$$\max_{\mathbf{h}} \rho^2\left(x, \mathbf{h}^T\mathbf{x}\right) \quad \text{subject to} \quad \mathrm{SNR}(\mathbf{h}) = \beta_1 \cdot \mathrm{SNR}, \tag{2.164}$$

where $\beta_1 > 1$. If we use a Lagrange multiplier, μ, to adjoin the constraint to the cost function, (2.164) can be rewritten as

$$\max_{\mathbf{h}} \mathcal{L}(\mathbf{h}, \mu), \tag{2.165}$$

with

$$\mathcal{L}(\mathbf{h}, \mu) = \frac{\left(\mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}\right)^2}{\sigma_x^2\left(\mathbf{h}^T\mathbf{R}_{xx}\mathbf{h}\right)} + \mu\left(\frac{\mathbf{h}^T\mathbf{R}_{xx}\mathbf{h}}{\mathbf{h}^T\mathbf{R}_{vv}\mathbf{h}} - \beta_1 \cdot \mathrm{SNR}\right). \tag{2.166}$$

Taking the gradient of $\mathcal{L}(\mathbf{h}, \mu)$ with respect to $\mathbf{h}$ and equating the result to zero, we get

$$\frac{2\sigma_x^2\left(\mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}\right)\left(\mathbf{h}^T\mathbf{R}_{xx}\mathbf{h}\right)\mathbf{R}_{xx}\mathbf{h}_1-2\sigma_x^2\left(\mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}\right)^2\mathbf{R}_{xx}\mathbf{h}}{\left(\sigma_x^2\cdot\mathbf{h}^T\mathbf{R}_{xx}\mathbf{h}\right)^2}+$$
$$\mu\frac{2\left(\mathbf{h}^T\mathbf{R}_{vv}\mathbf{h}\right)\mathbf{R}_{xx}\mathbf{h}-2\left(\mathbf{h}^T\mathbf{R}_{xx}\mathbf{h}\right)\mathbf{R}_{vv}\mathbf{h}}{\left(\mathbf{h}^T\mathbf{R}_{vv}\mathbf{h}\right)^2}=\mathbf{0}_{L\times 1}. \tag{2.167}$$

Now let's look for the optimal filter, $\mathbf{h}_{\mathrm{T}}$, that satisfies the relation

$$\mathbf{h}_1^T\mathbf{R}_{xx}\mathbf{h}_{\mathrm{T}}=\mathbf{h}_{\mathrm{T}}^T\mathbf{R}_{xx}\mathbf{h}_{\mathrm{T}}. \tag{2.168}$$

In this case, (2.167) becomes

$$\frac{\mathbf{R}_{xx}\mathbf{h}_1}{\sigma_x^2}-\frac{\mathbf{R}_{xx}\mathbf{h}_{\mathrm{T}}}{\sigma_x^2}+\mu\frac{\left(\mathbf{h}_{\mathrm{T}}^T\mathbf{R}_{vv}\mathbf{h}_{\mathrm{T}}\right)\mathbf{R}_{xx}\mathbf{h}_{\mathrm{T}}-\left(\mathbf{h}_{\mathrm{T}}^T\mathbf{R}_{xx}\mathbf{h}_{\mathrm{T}}\right)\mathbf{R}_{vv}\mathbf{h}_{\mathrm{T}}}{\left(\mathbf{h}_{\mathrm{T}}^T\mathbf{R}_{vv}\mathbf{h}_{\mathrm{T}}\right)^2}=\mathbf{0}_{L\times 1}. \tag{2.169}$$

Left-multiplying both sides of (2.169) by $\mathbf{h}_{\mathrm{T}}^T$, we can check that, indeed, the filter $\mathbf{h}_{\mathrm{T}}$ satisfies the relation (2.168). After some simple manipulations on (2.169), we find that

$$\mathbf{R}_{xx}\mathbf{h}_1-\mathbf{R}_{xx}\mathbf{h}_{\mathrm{T}}+\mu\mathrm{SNR}\xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{T}}\right)\mathbf{R}_{xx}\mathbf{h}_{\mathrm{T}}-\mu\beta_1\mathrm{SNR}^2\xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{T}}\right)\mathbf{R}_{vv}\mathbf{h}_{\mathrm{T}}=\mathbf{0}_{L\times 1}. \tag{2.170}$$

Define the quantities:

$$\tilde{\mathbf{R}}_{xx}=\frac{\mathbf{R}_{xx}}{\sigma_x^2}, \tag{2.171}$$
$$\tilde{\mathbf{R}}_{vv}=\frac{\mathbf{R}_{vv}}{\sigma_v^2}, \tag{2.172}$$
$$\mu'=\mu\beta_1\mathrm{SNR}^2\xi_{\mathrm{nr}}\left(\mathbf{h}_{\mathrm{T}}\right). \tag{2.173}$$

We find the optimal trade-off filter

$$\mathbf{h}_{\mathrm{T}}=\left[\frac{\mu'}{\mathrm{SNR}}\mathbf{I}_{L\times L}+\left(\mathbf{I}_{L\times L}-\frac{\mu'}{\beta_1\mathrm{SNR}}\right)\tilde{\mathbf{R}}_{vv}^{-1}\tilde{\mathbf{R}}_{xx}\right]^{-1}\tilde{\mathbf{R}}_{vv}^{-1}\tilde{\mathbf{R}}_{xx}\mathbf{h}_1, \tag{2.174}$$

which can be compared to the Wiener filter form shown in (2.11).

The purpose of the filter $\mathbf{h}_{\mathrm{T}}$ is the same as the filters derived in [59], [69]. We can play with the parameters μ' and β_1 to get different forms of the trade-off filter. For examples, for $\mu'=0$ we have the speech distortionless filter, $\mathbf{h}_{\mathrm{T}}=\mathbf{h}_1$, and for $\mu'=1$ and $\beta_1\to\infty$, we get the Wiener filter, $\mathbf{h}_{\mathrm{T}}=\mathbf{h}_{\mathrm{W}}$.

Another example of a trade-off filter can be derived by maximizing the output SNR while setting the speech distortion to a certain level. Mathematically, this optimization problem can be formulated as follows:

$$\max_{\mathbf{h}} \text{SNR}(\mathbf{h}) \quad \text{subject to} \quad \rho^2\left(x, \mathbf{h}^T\mathbf{x}\right) = \beta_2, \tag{2.175}$$

where $\beta_2 < 1$. Following the same steps developed for the optimization problem of (2.164), it can be shown that the optimal trade-off filter derived from (2.175) is

$$\mathbf{h}_{\text{T},2} = \left[\frac{\mu''}{\text{SNR}}\mathbf{I}_{L\times L} + \left(\mathbf{I}_{L\times L} - \frac{\mu''}{\beta_2'\text{SNR}}\right)\tilde{\mathbf{R}}_{vv}^{-1}\tilde{\mathbf{R}}_{xx}\right]^{-1}\tilde{\mathbf{R}}_{vv}^{-1}\tilde{\mathbf{R}}_{xx}\mathbf{h}_1, \tag{2.176}$$

where

$$\beta_2' = \beta_2\xi_{\text{nr}}\left(\mathbf{h}_{\text{T},2}\right), \tag{2.177}$$

$$\mu'' = \beta_2\frac{\left[\text{SNR}\xi_{\text{nr}}\left(\mathbf{h}_{\text{T},2}\right)\right]^2}{\mu}. \tag{2.178}$$

The two optimal trade-off filters $\mathbf{h}_{\text{T}}$ and $\mathbf{h}_{\text{T},2}$ are in the same form even though the latter is rarely used in practice because the level of speech distortion is very difficult to control.

2.6 Conclusions

Optimal filters play a key role in noise reduction with a single microphone or with a microphone array. Depending on the context, it is often possible to derive an optimal filter that can lead to an acceptable performance for a given problem.

In this chapter, we have studied three important filters: Wiener, Frost, and Kalman. The Wiener filter is simple and quite useful but has its limitations. We have seen, in detail, how this optimal filter distorts the desired signal. The Frost filter is a form of the Wiener filter in which we attached some constraints. We will see later in this book that the Frost algorithm, when the signal model is well exploited, can give remarkable performances. The Kalman filter which can be seen as a generalization of the Wiener filter for nonstationary signals is powerful but requires the knowledge of some a priori information that is not often available in real-time applications. We have also introduced a viable alternative to the MSE. We have shown how the SPCC can be exploited as a criterion instead of the classical MSE and why it is natural to use in the derivation of different types of optimal filters.

3

Conventional Beamforming Techniques

3.1 Introduction

Beamforming has a long history; it has been studied in many areas such as radar, sonar, seismology, communications, to name a few. It can be used for plenty of different purposes, such as detecting the presence of a signal, estimating the direction of arrival (DOA), and enhancing a desired signal from its measurements corrupted by noise, competing sources, and reverberation. Traditionally, a beamformer is formulated as a spatial filter that operates on the outputs of a sensor array in order to form a desired beam (directivity) pattern. Such a spatial filtering operation can be further decoupled into two sub-processes: synchronization and weight-and-sum. The synchronization process is to delay (or advance) each sensor output by a proper amount of time so that the signal components coming from a desired direction are synchronized. The information required in this step is the time difference of arrival (TDOA), which, if not known *a priori*, can be estimated from the array measurements using time-delay estimation techniques. The weight-and-sum step, as its name indicates, is to weight the aligned signals and then add the results together to form one output. Although both processes play an important role in controlling the array beam pattern (the synchronization part controls the steering direction and the weight-and-sum process controls the beamwidth of the mainlobe and the characteristics of the sidelobes), attention to beamforming is often paid to the second step on determining the weighting coefficients. In many applications, the weighting coefficients can be determined based on a pre-specified array beam pattern, but usually it is more advantageous to estimate the coefficients in an adaptive manner based on the signal and noise characteristics.

The spatial-filter based beamformers were developed for narrowband signals that can be sufficiently characterized by a single frequency. For broadband speech that has rich frequency content, such beamformers would not yield the same beam pattern for different frequencies and the beamwidth decreases as the frequency increases. If we use such a beamformer, when the steering di-

rection is different from the source incident angle, the source signal will be lowpass filtered. In addition, noise coming from a direction different from the beamformer's look direction will not be attenuated uniformly over its entire spectrum, resulting in some disturbing artifacts in the array output. Therefore, response-invariant broadband beamforming techniques have to be developed. A common way to design such a broadband beamformer is to perform a subband decomposition and design narrowband beamformers independently at each frequency. This is equivalent to applying a spatio-temporal filter to the array outputs, which is widely known as the filter-and-sum structure. The core problem of broadband beamforming then becomes one of determining the coefficients of the spatio-temporal filter.

This chapter discusses the basic ideas underlying conventional beamforming in the context of signal enhancement. (Note that the fundamental principles of beamforming vary in functionality. Besides signal enhancement, another major application of beamforming is the measurement of DOA, which will be covered in Chapter 9.) We will begin with a brief discussion on the advantages of using an array as compared to the use of a single sensor. We then explore what approaches can be used for solving the narrowband problem. Although they were not developed for processing speech, the narrowband techniques lay basis for more advanced broadband beamforming in acoustic environments and can be used sometimes with good results with broadband signals. Many fundamental ideas developed in the narrowband case can be extended to the broadband situation. To illustrate this, we will address the philosophy behind the (response-invariant) broadband beamforming, which is of more interest in the context of microphone arrays.

3.2 Problem Description

In sensor arrays, a widely used signal model assumes that each propagation channel introduces some delay and attenuation only. With this assumption and in the scenario where we have an array consisting of N sensors, the array outputs, at time k, are expressed as

$$\begin{aligned} y_n(k) &= \alpha_n s\left[k - t - \mathcal{F}_n(\tau)\right] + v_n(k) \\ &= x_n(k) + v_n(k),\ n = 1, 2, \ldots, N, \end{aligned} \tag{3.1}$$

where α_n $(n = 1, 2, \ldots, N)$, which range between 0 and 1, are the attenuation factors due to propagation effects, $s(k)$ is the unknown source signal (which can be narrowband or broadband), t is the propagation time from the unknown source to sensor 1, $v_n(k)$ is an additive noise signal at the nth sensor, τ is the relative delay [or more often it is called the time difference of arrival (TDOA)] between sensors 1 and 2, and $\mathcal{F}_n(\tau)$ is the relative delay between sensors 1 and n with $\mathcal{F}_1(\tau) = 0$ and $\mathcal{F}_2(\tau) = \tau$. In this chapter, we make a key assumption that τ and $\mathcal{F}_n(\tau)$ are known or can be estimated and the source and noise

signals are uncorrelated. We also assume that all the signals in (3.1) are zero-mean and stationary.

By processing the array observations $y_n(k)$, we can acquire much useful information about the source, such as its position, frequency, etc. The problem considered in this chapter is, however, focused on reducing the effect that the additive noise terms, $v_n(k)$, may have on the desired signal, thereby improving the signal-to-noise ratio (SNR). Without loss of generality, we consider the first sensor as the reference signal. The goal of this chapter can, then, be described as to recover $x_1(k) = \alpha_1 s(k-t)$ up to an eventual delay.

3.3 Delay-and-Sum Technique

The advantages of using an array to enhance the desired signal reception while simultaneously suppressing the undesired noise can be illustrated by a delay-and-sum (DS) beamformer. Such a beamformer consists of two basic processing steps [27], [63], [72], [73], [135], [197], [243]. The first step is to time-shift each sensor signal by a value corresponding to the TDOA between that sensor and the reference one. With the signal model given in (3.1) and after time shifting, we obtain

$$\begin{aligned} y_{\mathrm{a},n}(k) &= y_n\left[k + \mathcal{F}_n(\tau)\right] \\ &= \alpha_n s(k-t) + v_{\mathrm{a},n}(k) \\ &= x_{\mathrm{a},n}(k) + v_{\mathrm{a},n}(k),\ n = 1, 2, \ldots, N, \end{aligned} \tag{3.2}$$

where

$$v_{\mathrm{a},n}(k) = v_n\left[k + \mathcal{F}_n(\tau)\right],$$

and the subscript 'a' implies an aligned copy of the sensor signal. The second step consists of adding up the time-shifted signals, giving the output of a DS beamformer:

$$\begin{aligned} z_{\mathrm{DS}}(k) &= \frac{1}{N}\sum_{n=1}^{N} y_{\mathrm{a},n}(k) \\ &= \alpha_{\mathrm{s}} s(k-t) + \frac{1}{N} v_{\mathrm{s}}(k), \end{aligned} \tag{3.3}$$

where

$$\begin{aligned} \alpha_{\mathrm{s}} &= \frac{1}{N}\sum_{n=1}^{N} \alpha_n, \\ v_{\mathrm{s}}(k) &= \sum_{n=1}^{N} v_{\mathrm{a},n}(k) \\ &= \sum_{n=1}^{N} v_n\left[k + \mathcal{F}_n(\tau)\right]. \end{aligned}$$

Now we can examine the input and output SNRs of the DS beamformer. For the signal model given in (3.1), the input SNR relatively to the reference signal is

$$\text{SNR} = \frac{\sigma_{x_1}^2}{\sigma_{v_1}^2} = \alpha_1^2 \frac{\sigma_s^2}{\sigma_{v_1}^2}, \tag{3.4}$$

where $\sigma_{x_1}^2 = E\left[x_1^2(k)\right]$, $\sigma_{v_1}^2 = E\left[v_1^2(k)\right]$, and $\sigma_s^2 = E\left[s^2(k)\right]$ are the variances of the signals $x_1(k)$, $v_1(k)$, and $s(k)$, respectively. After DS processing, the output SNR can be expressed as the ratio of the variances of the first and second terms in the right-hand side of (3.3):

$$\begin{aligned} \text{oSNR} &= N^2 \alpha_{\text{s}}^2 \frac{E\left[s^2(k-t)\right]}{E\left[v_{\text{s}}^2(k)\right]} \\ &= N^2 \alpha_{\text{s}}^2 \frac{\sigma_s^2}{\sigma_{v_{\text{s}}}^2} \\ &= \left(\sum_{n=1}^{N} \alpha_n\right)^2 \frac{\sigma_s^2}{\sigma_{v_{\text{s}}}^2}, \end{aligned} \tag{3.5}$$

where

$$\begin{aligned} \sigma_{v_{\text{s}}}^2 &= E\left\{\left[\sum_{n=1}^{N} v_n\left[k + \mathcal{F}_n(\tau)\right]\right]^2\right\} \\ &= \sum_{n=1}^{N} \sigma_{v_n}^2 + 2\sum_{i=1}^{N-1}\sum_{j=i+1}^{N} \varrho_{v_i v_j}, \end{aligned} \tag{3.6}$$

with $\sigma_{v_n}^2 = E\left[v_n^2(k)\right]$ being the variance of the noise signal, $v_n(k)$, and $\varrho_{v_i v_j} = E\left\{v_i\left[k + \mathcal{F}_i(\tau)\right] v_j\left[k + \mathcal{F}_j(\tau)\right]\right\}$ being the cross-correlation function between $v_i(k)$ and $v_j(k)$.

The DS beamformer is of interest only if

$$\text{oSNR} > \text{SNR}. \tag{3.7}$$

This will mean that the signal $z_{\text{DS}}(k)$ will be less noisy than any microphone output signal, $y_n(k)$, and will possibly be a good approximation of $x_1(k)$.

Particular Case 1:

In this particular case, we assume that the noise signals at the microphones are uncorrelated , i.e., $\varrho_{v_i v_j} = 0,\ \forall i, j = 1, 2, \ldots, N,\ i \neq j$, and they all have the same variance, i.e., $\sigma_{v_1}^2 = \sigma_{v_2}^2 = \cdots = \sigma_{v_N}^2$. We also suppose that all the attenuation factors are equal to 1 (i.e., $\alpha_n = 1,\ \forall n$). Then it can be easily checked that

$$\text{oSNR} = N \cdot \text{SNR}. \tag{3.8}$$

It is interesting to see that under the previous conditions, a simple time-shifting and adding operation among the sensor outputs results in an improvement in the SNR by a factor equal to the number of sensors.

Particular Case 2:

Here, we only assume that the noise signals have the same energy and that all the attenuation factors are equal to 1. In this case we have

$$\text{oSNR} = \frac{N}{1+\rho_{\text{s}}} \cdot \text{SNR}, \tag{3.9}$$

where

$$\rho_{\text{s}} = \frac{2}{N} \sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \rho_{v_i v_j},$$

$$\rho_{v_i v_j} = \frac{\varrho_{v_i v_j}}{\sigma_{v_i} \sigma_{v_j}}.$$

$\rho_{v_i v_j}$ is the correlation coefficient with $|\rho_{v_i v_j}| \leq 1$. Normally, this coefficient ranges between -1 and 1. If the noise signals at the microphones are completely correlated, i.e., $\rho_{v_i v_j} = 1$, we have $\sum_{i=1}^{N-1} \sum_{j=i+1}^{N} \rho_{v_i v_j} = N(N-1)/2$. In this case, oSNR = SNR. So, no gain is possible with the DS technique. As the value of the correlation coefficient $\rho_{v_i v_j}$ decreases from 1 to 0, the gain in SNR increases. In some situations, the correlation coefficient $\rho_{v_i v_j}$ can be negative. This can happen when the noise signals are from a common point source. In this case, we may get an SNR higher than N or even infinite.

Another way of illustrating the performance of a DS beamformer is through examining the corresponding beam pattern (beam pattern is sometimes written in a compound form as beampattern; it is also called directivity pattern or spatial pattern) [217], which provides a complete characterization of the array system's input-output behavior. From the previous analysis, we easily see that a DS beamformer is indeed an N-point spatial filter and its beam pattern is defined as the magnitude of the spatial filter's directional response. From (3.2) and (3.3), we can check that the nth coefficient of the spatial filter is $\frac{1}{N} e^{j2\pi f \mathcal{F}_n(\tau)}$, where f denotes frequency. The directional response of this filter can be found by performing the Fourier transform. Since $\mathcal{F}_n(\tau)$ depends on both the array geometry and the source position, so the beam pattern of a DS beamformer should be a function of the array geometry and source position. In addition, the beam pattern is also a function of the number of sensors and the signal frequency. Now suppose that we have an equispaced linear array, which consists of N omnidirectional sensors, as illustrated in Fig. 3.1. If we denote the spacing between two neighboring sensors as d, and assume

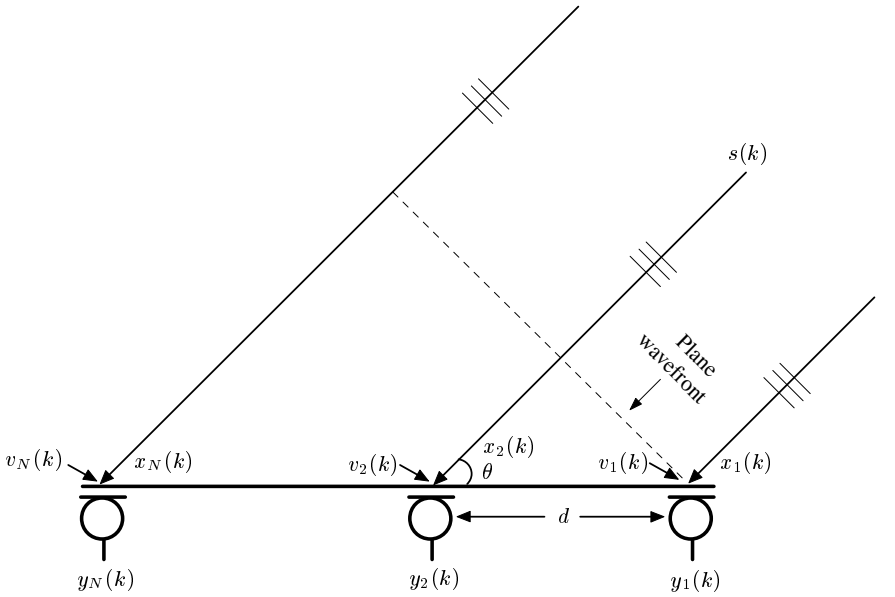


Fig. 3.1. Illustration of an equispaced linear array, where the source $s(k)$ is located in the far field, the incident angle is θ, and the spacing between two neighboring sensors is d.

that the source is in the far field and the wave rays reach the array with an incident angle of θ, the TDOA between the nth and the reference sensors can be written as

$$\mathcal{F}_n(\tau) = (n-1)\tau = (n-1)d\cos(\theta)/c, \tag{3.10}$$

where c denotes the sound velocity in air. In this case, the directional response of the DS filter, which is the spatial Fourier transform of the filter [8], [90], can be expressed as

$$\begin{aligned}\mathcal{S}_{\mathrm{DS}}(\psi,\theta) &= \frac{1}{N}\sum_{n=1}^{N}\left[e^{j2\pi(n-1)fd\cos(\theta)/c}\right]e^{-j2\pi(n-1)fd\cos(\psi)/c} \\ &= \frac{1}{N}\sum_{n=1}^{N}e^{-j2\pi(n-1)fd[\cos(\psi)-\cos(\theta)]/c},\end{aligned} \tag{3.11}$$

where ψ $(0 \le \psi \le \pi)$ is a directional angle. The beam pattern is then written as

$$\begin{aligned}\mathcal{A}_{\mathrm{DS}}(\psi,\theta) &= |\mathcal{S}_{\mathrm{DS}}(\psi,\theta)| \\ &= \left|\frac{\sin\left[N\pi fd(\cos\psi-\cos\theta)/c\right]}{N\sin\left[\pi fd(\cos\psi-\cos\theta)/c\right]}\right|.\end{aligned} \tag{3.12}$$

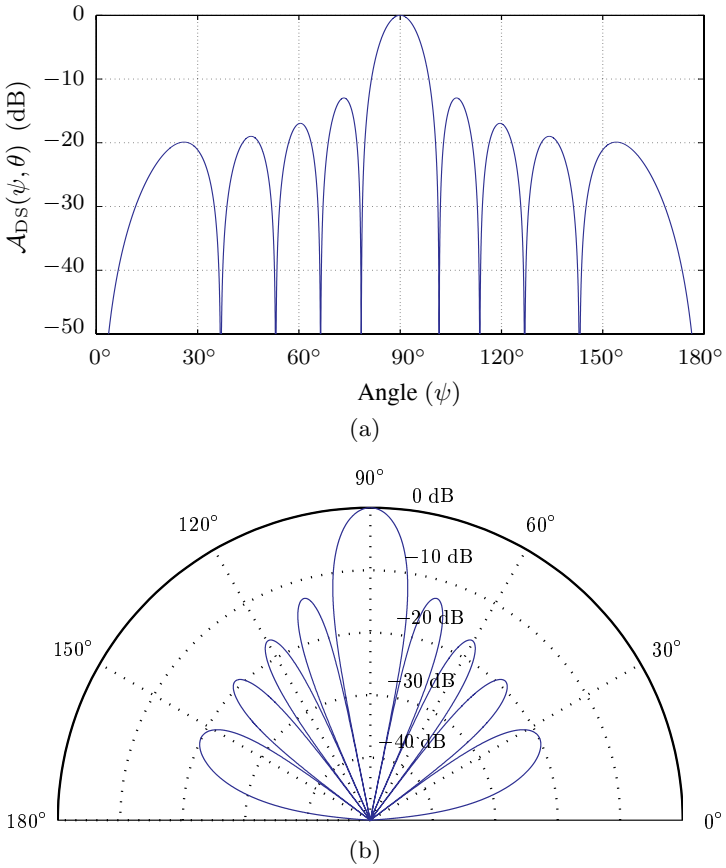


Fig. 3.2. Beam pattern of a ten-sensor array when $\theta = 90°$, $d = 8$ cm, and $f = 2$ kHz: (a) in Cartesian coordinates and (b) in polar coordinates.

Figure 3.2 plots the beam pattern for an equispaced linear array with ten sensors, $d = 8$ cm, $\theta = 90°$, and $f = 2$ kHz. It consists of a total of 9 beams (in general, the number of beams in the range between 0° and 180° is equal to $N - 1$). The one with the highest amplitude is called mainlobe and all the others are called sidelobes. One important parameter regarding the mainlobe is the beamwidth (mainlobe width), which is defined as the region between the first zero-crosses on either side of the mainlobe. With the above linear array, the beamwidth can be easily calculated as $2\cos^{-1}[c/(Ndf)]$. This number decreases with the increase of the number of sensors, the spacing between neighboring sensors, and the signal frequency. The height of the sidelobes represents the gain pattern for noise and competing sources present along the directions other than the desired look direction. In array and beamforming design, we hope to make the sidelobes as low as possible so that signals coming from directions other than the look direction would be attenuated as much

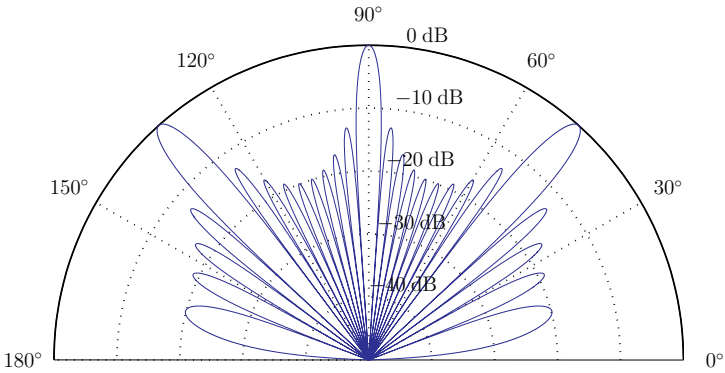


Fig. 3.3. Beam pattern (in polar coordinates) of a ten-sensor array when $\theta = 90°$, $d = 24$ cm, and $f = 2$ kHz.

as possible. In addition, with a spatial filter of length N, there always exists $N-1$ nulls. We can design the weighting coefficients so that these nulls would be placed along the directions of competing sources. This is related to the adaptive beamforming technique and will be covered in greater detail in the next sections.

Before we finish this section, we would like to point out one potential problem with the sensor spacing. From the previous analysis, we see that the array beamwidth decreases as the spacing d increases. So, if we want a sharper beam, we can simply increase the spacing d, which leads to a larger array aperture. This would, in general, lead to more noise reduction. Therefore, in array design, we would expect to set the spacing as large as possible. However, when d is larger than $\lambda/2 = c/(2f)$, where λ is the wavelength of the signal, spatial aliasing would arise. To visualize this problem, we plot the beam pattern for an equispaced linear array same as used in Fig. 3.2(b). The signal frequency f is still 2 kHz. But this time, the array spacing is 24 cm. The corresponding beam pattern is shown in Fig. 3.3. This time, we see three large beams that have a maximum amplitude of 1. The other two are called grating lobes. Signals propagating from directions at which grating lobes occur would be indistinguishable from signals propagating from the mainlobe direction. This ambiguity is often referred to as spatial aliasing. In order to avoid spatial aliasing, the array spacing has to satisfy $d \leq \frac{\lambda}{2} = \frac{c}{2f}$. By analogy to the Nyquist sampling theorem, this result may be interpreted as a spatial sampling theorem.

3.4 Design of a Fixed Beamformer

As seen from the previous discussion, once the array geometry is fixed and the desired steering direction is determined, the characteristics of the beam pattern of a DS beamformer, including the beamwidth, the amplitude of the

sidelobes, and the positions of the nulls, would be fixed. This means that if we want to adjust the beam pattern, we have to make physical changes to the array geometry, which is virtually impossible once an array system is delivered. A legitimate question then arises: can we improve the array performance with some signal processing techniques to adjust its beam pattern without changing its geometry? We attempt to answer this question in this section and discuss a class of techniques called fixed beamforming, which takes into account the array geometry but assumes no information from neither the source nor the noise signals.

Reexamining the DS beamformer, we easily see that the underlying idea is to apply a spatial filter of length N to the sensor outputs. This is similar to the idea of temporal filtering using a finite-duration impulse response (FIR) filter. Therefore, all the techniques developed for designing FIR filters, including both the windowing and optimum-approximation approaches [177], can be applied here. To illustrate how to design a beamformer to achieve a desired beam pattern, we consider here the widely used least-squares (LS) technique, which is an optimum-approximation approach.

Suppose that $\mathbf{h} = \begin{bmatrix} h_1 & h_2 & \cdots & h_N \end{bmatrix}^T$ is a beamforming filter of length N, the corresponding directional response is

$$\mathcal{S}(\psi) = \sum_{n=1}^{N} h_n e^{-j2\pi f \mathcal{F}_n[\tau(\psi)]} = \mathbf{h}^T \boldsymbol{\varsigma}(\psi), \tag{3.13}$$

where

$$\boldsymbol{\varsigma}(\psi) = \begin{bmatrix} e^{-j2\pi f \mathcal{F}_1[\tau(\psi)]} & e^{-j2\pi f \mathcal{F}_2[\tau(\psi)]} & \cdots & e^{-j2\pi f \mathcal{F}_N[\tau(\psi)]} \end{bmatrix}^T.$$

In the LS method, the objective is to optimize the filter coefficients h_n ($n = 1, 2, \ldots, N$) such that the resulting directional response can best approximate a given directional response. To achieve this goal, let us first define the LS approximation criterion:

$$\epsilon^2 = \int_0^{\pi} \vartheta(\psi) \left| \mathcal{S}(\psi) - \mathcal{S}_{\mathrm{d}}(\psi) \right|^2 d\psi, \tag{3.14}$$

where $\mathcal{S}_{\mathrm{d}}(\psi)$ denotes the desired directional response, and $\vartheta(\psi)$ is a positive real weighting function to either emphasize or deemphasize the importance of certain angles.

Substituting (3.13) into (3.14), we can rewrite the LS approximation criterion as

$$\epsilon^2 = \mathbf{h}^T \mathbf{Q} \mathbf{h} - 2\mathbf{h}^T \mathbf{p} + \int_0^{\pi} \vartheta(\psi) |\mathcal{S}_{\mathrm{d}}(\psi)|^2 d\psi, \tag{3.15}$$

where

$$\mathbf{Q} = \int_0^{\pi} \vartheta(\psi)\boldsymbol{\varsigma}(\psi)\boldsymbol{\varsigma}^H(\psi)d\psi,$$
$$\mathbf{p} = \int_0^{\pi} \vartheta(\psi)\mathrm{Re}[\boldsymbol{\varsigma}(\psi)\mathcal{S}_{\mathrm{d}}(\psi)]d\psi, \tag{3.16}$$

$\mathrm{Re}(\cdot)$ denotes real part, and superscript ^{H} denotes transpose conjugate of a vector or a matrix.

Differentiating ϵ^2 with respect to $\mathbf{h}$ and equating the result to zero gives

$$\mathbf{h}_{\mathrm{LS}} = \mathbf{Q}^{-1}\mathbf{p}. \tag{3.17}$$

One can notice that the matrix $\mathbf{Q}$ is a function of $\mathcal{F}_N[\tau(\psi)]$ and vector $\mathbf{p}$ is a function of both $\mathcal{F}_N[\tau(\psi)]$ and $\mathcal{S}_{\mathrm{d}}(\psi)$. Therefore, the LS beamforming filter depends on both the array geometry and the desired directional response.

Now let us consider the case where we have an equispaced linear array, same as used in the previous section. Suppose that we know the source is located in a certain region (between angles ψ_1 and ψ_2), but we do not have the accurate information regarding the source incident direction. So, we want to design a beamformer that can pass the signal incident from the range between ψ_1 and ψ_2, but attenuate signals from all other directions. Mathematically, in this case, we want to obtain a desired directional response

$$\mathcal{S}_{\mathrm{d}}(\psi) = \begin{cases} 1 & \text{if } \psi_1 \le \psi \le \psi_2 \\ 0 & \text{otherwise} \end{cases}. \tag{3.18}$$

If we assume that all the angles are equally important, i.e. $\vartheta(\psi) = 1$, then

$$\begin{aligned}
\mathbf{Q} &= \int_0^{\pi} \boldsymbol{\varsigma}(\psi)\boldsymbol{\varsigma}^H(\psi)d\psi \\
&= \begin{bmatrix} \int_0^{\pi} 1d\psi & \int_0^{\pi} e^{j\tilde{d}_1\cos\psi}d\psi & \cdots & \int_0^{\pi} e^{j\tilde{d}_{N-1}\cos\psi}d\psi \\ \int_0^{\pi} e^{-j\tilde{d}_1\cos\psi}d\psi & \int_0^{\pi} 1d\psi & \cdots & \int_0^{\pi} e^{j\tilde{d}_{N-2}\cos\psi}d\psi \\ \vdots & \vdots & \ddots & \vdots \\ \int_0^{\pi} e^{-j\tilde{d}_{N-1}\cos\psi}d\psi & \cdots & \cdots & \int_0^{\pi} 1d\psi \end{bmatrix} \\
&= \begin{bmatrix} \pi & \int_0^{\pi}\cos(\tilde{d}_1\cos\psi)d\psi & \cdots & \int_0^{\pi}\cos(\tilde{d}_{N-1}\cos\psi)d\psi \\ \int_0^{\pi}\cos(\tilde{d}_1\cos\psi)d\psi & \pi & \cdots & \int_0^{\pi}\cos(\tilde{d}_{N-2}\cos\psi)d\psi \\ \vdots & \vdots & \ddots & \vdots \\ \int_0^{\pi}\cos(\tilde{d}_{N-1}\cos\psi)d\psi & \cdots & \cdots & \pi \end{bmatrix},
\end{aligned} \tag{3.19}$$

$$\mathbf{p} = \begin{bmatrix} \int_{\psi_1}^{\psi_2} 1d\psi \\ \int_{\psi_1}^{\psi_2}\cos(\tilde{d}_1\cos\psi)d\psi \\ \vdots \\ \int_{\psi_1}^{\psi_1}\cos(\tilde{d}_{N-1}\cos\psi)d\psi \end{bmatrix}, \tag{3.20}$$

where $\tilde{d}_n = 2\pi n f d/c$, $n = 1, 2, \ldots, N-1$.

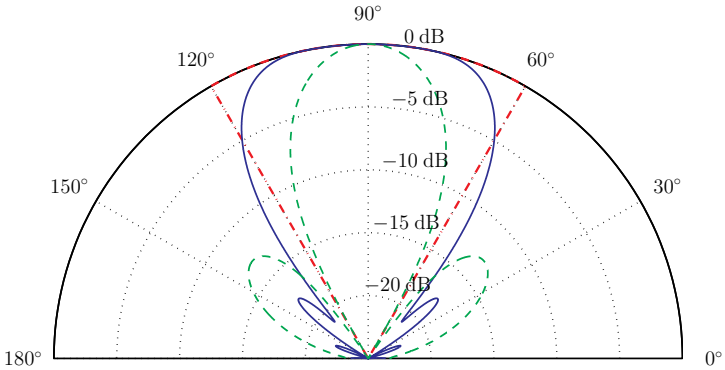


Fig. 3.4. Beam pattern designed using the LS technique (solid line): the array is an equispaced linear one with 10 sensors, $d = 4$ cm, $f = 1.5$ kHz, $\psi_1 = 60°$, $\psi_2 = 120°$. For comparison, the DS (dashed line) and desired beam pattern (dash-dot line) are also shown.

The integrals in (3.19) and (3.20) may seem difficult to evaluate, but they can be computed using numerical methods without any problems. Now let us consider two design examples. In the first one, we consider a scenario where the source may be moving from time to time in the range between 60° and 120°. In order not to distort the source signal, we want a beamfomer with a large beamwidth, covering from 60° to 120°. Figure 3.4 plots such a beamformer design using the LS technique. As seen, its mainlobe is much broader than that of a DS beamformer.

In the second example, we assume that we know the source is located in the broadside direction (90°), with an error less than ±5°. This time, we want to have a narrower beam for more interference reduction. The corresponding beam pattern using the LS method is plotted in Fig 3.5. It is seen that this time the beamwidth is much smaller than that of a DS beamformer.

Note that the LS beamforming filter can be formulated using different LS criteria [4], [61], [173]. The one in (3.17) is achieved by approximating the desired directional response, which takes into account both the magnitude and phase. We can also formulate the LS filter by approximating the desired beam pattern, in which case the phase response will be neglected.

3.5 Maximum Signal-to-Noise Ratio Filter

The fixed beamforming techniques can fully take advantage of the array geometry and source location information to optimize their beam pattern. However, the ability of a fixed-beamforming array system in suppressing noise and competing sources is limited by many factors, e.g., the array aperture. One way to achieve a higher SNR gain when the array geometry is fixed is through using the characteristics of both the source and noise signals, resulting in a

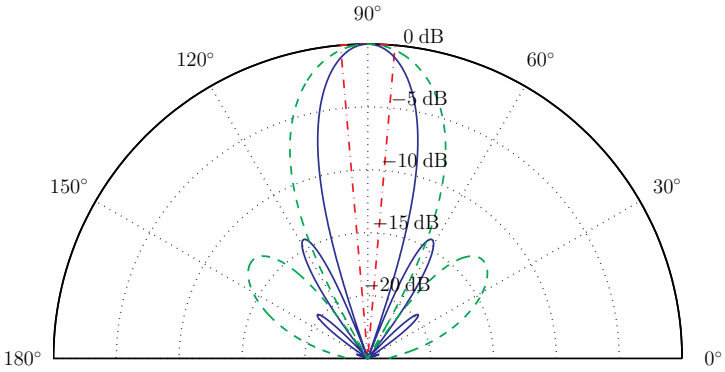


Fig. 3.5. Beam pattern designed using the LS technique (solid line): the array is an equispaced linear one with 10 sensors, $d = 4$ cm, $f = 1.5$ kHz, $\psi_1 = 85°$, $\psi_2 = 95°$. For comparison, the DS (dashed line) and desired beam pattern (dash-dot line) are also shown.

wide variety of array processing algorithms called adaptive beamforming techniques. In this section, we illustrate the idea underlying adaptive beamforming by deriving the optimal filter that maximizes the SNR at the output of the beamformer [5].

In order to show the principle underlying the maximum-SNR technique, let us rewrite (3.2) in a vector/matrix form:

$$\mathbf{y}_{\mathrm{a}}(k) = s(k-t)\boldsymbol{\alpha} + \mathbf{v}_{\mathrm{a}}(k), \tag{3.21}$$

where

$$\begin{aligned}
\mathbf{y}_{\mathrm{a}}(k) &= \left[y_{\mathrm{a},1}(k)\ y_{\mathrm{a},2}(k) \cdots y_{\mathrm{a},N}(k) \right]^T, \\
\mathbf{v}_{\mathrm{a}}(k) &= \left[v_{\mathrm{a},1}(k)\ v_{\mathrm{a},2}(k) \cdots v_{\mathrm{a},N}(k) \right]^T, \\
\boldsymbol{\alpha} &= \left[\alpha_1\ \alpha_2 \cdots \alpha_N \right]^T.
\end{aligned}$$

Since the signal and noise are assumed to be uncorrelated, the correlation matrix of the vector signal $\mathbf{y}_{\mathrm{a}}(k)$ can be expressed as

$$\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}} = \sigma_s^2 \boldsymbol{\alpha}\boldsymbol{\alpha}^T + \mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}}, \tag{3.22}$$

where $\mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}} = E\left[\mathbf{v}_{\mathrm{a}}(k)\mathbf{v}_{\mathrm{a}}^T(k)\right]$ is the noise correlation matrix.

A more general form of a beamformer output is written as

$$\begin{aligned}
z(k) &= \mathbf{h}^T \mathbf{y}_{\mathrm{a}}(k) \\
&= \sum_{n=1}^{N} h_n y_{\mathrm{a},n}(k) \\
&= s(k-t)\mathbf{h}^T\boldsymbol{\alpha} + \mathbf{h}^T\mathbf{v}_{\mathrm{a}}(k),
\end{aligned} \tag{3.23}$$

where

$$\mathbf{h} = \begin{bmatrix} h_1 \; h_2 \cdots h_N \end{bmatrix}^T$$

is some filter of length N. In particular, taking $h_n = 1/N$, $\forall n$, we get the DS beamformer. With this general filter, the output SNR is written as

$$\mathrm{SNR}(\mathbf{h}) = \frac{\sigma_s^2 \left(\mathbf{h}^T \boldsymbol{\alpha}\right)^2}{\mathbf{h}^T \mathbf{R}_{v_\mathrm{a} v_\mathrm{a}} \mathbf{h}}. \tag{3.24}$$

In array processing, we hope to suppress the noise as much as we can. One straightforward way of doing this is to find a filter $\mathbf{h}$ that would maximize the positive quantity $\mathrm{SNR}(\mathbf{h})$. This is equivalent to solving the generalized eigenvalue problem:

$$\sigma_s^2 \boldsymbol{\alpha}\boldsymbol{\alpha}^T \mathbf{h} = \lambda \mathbf{R}_{v_\mathrm{a} v_\mathrm{a}} \mathbf{h}. \tag{3.25}$$

Assuming that $\mathbf{R}_{v_\mathrm{a} v_\mathrm{a}}^{-1}$ exists, the optimal solution to our problem is the eigenvector, $\mathbf{h}_{\max}$, corresponding to the maximum eigenvalue, $\lambda_{\max}$, of $\sigma_s^2 \mathbf{R}_{v_\mathrm{a} v_\mathrm{a}}^{-1} \boldsymbol{\alpha}\boldsymbol{\alpha}^T$. Hence

$$z_{\max}(k) = \mathbf{h}_{\max}^T \mathbf{y}_\mathrm{a}(k), \tag{3.26}$$

$$\mathrm{SNR}(\mathbf{h}_{\max}) = \lambda_{\max}. \tag{3.27}$$

Using the same conditions as in the Particular Case 1 of Section 3.3, (3.25) becomes

$$\mathrm{SNR} \cdot \boldsymbol{\alpha}\boldsymbol{\alpha}^T \mathbf{h}_{\max} = \lambda_{\max} \mathbf{h}_{\max}. \tag{3.28}$$

Left multiplying (3.28) by $\boldsymbol{\alpha}^T$, we get

$$\lambda_{\max} = N \cdot \mathrm{SNR}, \tag{3.29}$$

so that

$$\begin{aligned} \mathrm{SNR}(\mathbf{h}_{\max}) &= N \cdot \mathrm{SNR} \\ &= \mathrm{oSNR}. \end{aligned} \tag{3.30}$$

This implies that

$$\mathbf{h}_{\max} = \frac{1}{N} \begin{bmatrix} 1 \; 1 \cdots 1 \end{bmatrix}^T. \tag{3.31}$$

Therefore, in this particular case, the maximum SNR filter is identical to the DS beamformer. This observation is indeed very interesting because it shows that even though the DS filter was derived with no optimality properties associated with it, it can be optimal under certain conditions.

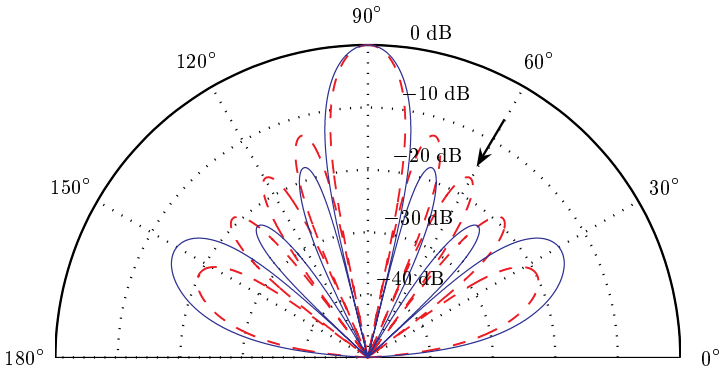


Fig. 3.6. Beam pattern for the maximum SNR filter (solid line): the array is an equispaced linear one with ten sensors; $d = 8$ cm; the noise signals are from a point narrowband source with unit amplitude and a frequency of 2 kHz; the noise source is located in the far field and propagates to the array with an incident angle of 60°. For comparison, the beam pattern for the DS algorithm is also shown (dashed line).

More insights into the maximum SNR filter can be obtained by considering scenarios where the noise signals are from a common point source. Let us consider an example where the noise source is a narrowband signal with unit amplitude and propagates to the array with an incident angle of 60°. Figure 3.6 plots the corresponding array beam pattern when the mainlobe is steered to $\theta = 90°$. Although the mainlobe is similar to that of a DS beamformer, the sidelobe structure is significantly different. Particularly, the maximum SNR filter produces a beam pattern having a null in the direction along which the noise source propagates to the array. In comparison, the nulls of a DS beamformer are located in fixed directions and are independent of the noise source. So, the maximum SNR filter indeed adapts its filter coefficients to the noise environment for maximum noise reduction.

From an SNR perspective, the maximum SNR technique is obviously the best we can do. However, in real acoustic environments this approach also has the possibility to maximize the speech distortion.

3.6 Minimum Variance Distortionless Response Filter

The minimum variance distortionless response (MVDR) technique, which is due to Capon [35], [134], [178], is perhaps the most widely used adaptive beamformer. The basic underlying idea is to choose the coefficients of the filter, $\mathbf{h}$, that minimize the output power, $E\left[z^2(k)\right] = \mathbf{h}^T\mathbf{R}_{y_\mathrm{a}y_\mathrm{a}}\mathbf{h}$, with the constraint that the desired signal [i.e., $x_1(k)$] is not affected. The MVDR problem for choosing the weights is thus written as [148], [216]

$$\min_{\mathbf{h}} \mathbf{h}^T\mathbf{R}_{y_\mathrm{a}y_\mathrm{a}}\mathbf{h} \quad \text{subject to} \quad \mathbf{h}^T\boldsymbol{\alpha} = \alpha_1. \tag{3.32}$$

The method of Lagrange multipliers can be used to solve (3.32), resulting in

$$\mathbf{h}_{\mathrm{C}} = \alpha_1 \frac{\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}}^{-1}\boldsymbol{\alpha}}{\boldsymbol{\alpha}^T\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}}^{-1}\boldsymbol{\alpha}}, \tag{3.33}$$

where the subscript 'C' denotes Capon. Therefore the beamformer output with the MVDR filter is

$$\begin{aligned} z_{\mathrm{C}}(k) &= \mathbf{h}_{\mathrm{C}}^T\mathbf{y}_{\mathrm{a}}(k) \\ &= \alpha_1 \frac{\boldsymbol{\alpha}^T\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}}^{-1}\mathbf{y}_{\mathrm{a}}(k)}{\boldsymbol{\alpha}^T\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}}^{-1}\boldsymbol{\alpha}} \\ &= x_1(k) + r_{\mathrm{n}}(k), \end{aligned} \tag{3.34}$$

where

$$r_{\mathrm{n}}(k) = \alpha_1 \frac{\boldsymbol{\alpha}^T\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}}^{-1}\mathbf{v}_{\mathrm{a}}(k)}{\boldsymbol{\alpha}^T\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}}^{-1}\boldsymbol{\alpha}}$$

is the residual noise.

The output SNR with the Capon filter can be evaluated as follows:

$$\begin{aligned} \mathrm{SNR}(\mathbf{h}_{\mathrm{C}}) &= \alpha_1^2 \frac{\sigma_s^2}{\sigma_{r_{\mathrm{n}}}^2} \\ &= \frac{\sigma_{v_1}^2}{\sigma_{r_{\mathrm{n}}}^2} \cdot \mathrm{SNR}, \end{aligned} \tag{3.35}$$

where $\sigma_{r_{\mathrm{n}}}^2 = E\left[r_{\mathrm{n}}^2(k)\right]$.

Determining the inverse of $\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}}$ from (3.22) with the Woodbury's identity

$$\left[\mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}} + \sigma_s^2\boldsymbol{\alpha}\boldsymbol{\alpha}^T\right]^{-1} = \mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}}^{-1} - \frac{\mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}}^{-1}\boldsymbol{\alpha}\boldsymbol{\alpha}^T\mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}}^{-1}}{\sigma_s^{-2} + \boldsymbol{\alpha}^T\mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}}^{-1}\boldsymbol{\alpha}^T} \tag{3.36}$$

and substituting the result into (3.33), we obtain:

$$\mathbf{h}_{\mathrm{C}} = \alpha_1 \frac{\mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}}^{-1}\boldsymbol{\alpha}}{\boldsymbol{\alpha}^T\mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}}^{-1}\boldsymbol{\alpha}}. \tag{3.37}$$

Using this form of the Capon filter, it is easy to check that $\mathbf{h}_{\mathrm{C}}$ is an eigenvector of (3.25) and

$$\mathbf{h}_{\mathrm{C}} = \mathbf{h}_{\max}. \tag{3.38}$$

Therefore, for the particular problem considered in this chapter, minimizing the total output power while keeping the signal from a specified direction constant is the same as maximizing the output SNR [88].

From (3.35), we can find that the residual noise power is

$$\sigma_{r_{\mathrm{n}}}^2 = \left(\boldsymbol{\alpha}^T \mathbf{R}_{v_{\mathrm{a}} v_{\mathrm{a}}}^{-1} \boldsymbol{\alpha}\right)^{-1}. \tag{3.39}$$

Identical to the maximum SNR filter, the output SNR with the Capon filter can also be written as

$$\begin{aligned} \mathrm{SNR}(\mathbf{h}_{\mathrm{C}}) &= \lambda_{\max} \\ &= \sigma_s^2 \left(\boldsymbol{\alpha}^T \mathbf{R}_{v_{\mathrm{a}} v_{\mathrm{a}}}^{-1} \boldsymbol{\alpha}\right). \end{aligned} \tag{3.40}$$

Applying the same conditions as in the Particular Case 1 of Section 3.3, we obtain:

$$\mathrm{SNR}(\mathbf{h}_{\mathrm{C}}) = N \cdot \mathrm{SNR}, \tag{3.41}$$

implying that the Capon filter degenerates to a DS beamformer when noise signals observed at the array are mutually uncorrelated and have the same power. But same as what we analyzed in section 3.6, the advantage of the Capon filter over a DS beamformer is that this adaptive beamformer can adapt itself to the noise environment for maximum noise reduction.

In more complicated propagation environments where reverberation is present, the Capon filter can be extended to a more general algorithm called the linearly constrained minimum variance filter. This will be studied in great details in Chapter 4.

3.7 Approach with a Reference Signal

Assume now that the reference or desired signal, $x_1(k)$, is available. We define the error signal as

$$\begin{aligned} e(k) &= x_1(k) - z(k) \\ &= \alpha_1 s(k-t) - \mathbf{h}^T \mathbf{y}_{\mathrm{a}}(k), \end{aligned} \tag{3.42}$$

which is the difference between the reference signal and its estimate. This error is then used in the MSE criterion

$$J(\mathbf{h}) = E\left[e^2(k)\right] \tag{3.43}$$

to find the optimal coefficients. The minimization of $J(\mathbf{h})$ with respect to the vector $\mathbf{h}$ yields to the well-known Wiener filter:

$$\mathbf{h}_{\mathrm{W}} = \mathbf{R}_{y_{\mathrm{a}} y_{\mathrm{a}}}^{-1} \mathbf{r}_{y_{\mathrm{a}} x_1}, \tag{3.44}$$

where

$$\mathbf{r}_{y_{\mathrm{a}} x_1} = E\left[\mathbf{y}_{\mathrm{a}}(k) x_1(k)\right] \tag{3.45}$$

is the cross-correlation vector between $\mathbf{y}_{\mathrm{a}}(k)$ and $x_1(k)$. Obviously, the desired signal, $x_1(k)$, is not available in most applications. As a result, $\mathbf{r}_{y_{\mathrm{a}} x_1}$ can not

be estimated as given in (3.45) and the optimal filter, $\mathbf{h}_{\mathrm{W}}$, can not be found. However, in many noise reduction applications there are interesting ways to estimate $\mathbf{r}_{y_{\mathrm{a}}x_1}$ [16], [218].

We are now ready to show how the Wiener filter is related to the other classical filters. Replacing (3.21) and $x_1(k) = \alpha_1 s(k-t)$ in (3.45) it is easy to see that this cross-correlation vector is

$$\mathbf{r}_{y_{\mathrm{a}}x_1} = \sigma_s^2 \alpha_1 \boldsymbol{\alpha}. \tag{3.46}$$

Using the decomposition of $\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}}^{-1}$ given by (3.36), the Wiener filter can be rewritten as [66]

$$\begin{aligned} \mathbf{h}_{\mathrm{W}} &= \frac{\alpha_1 \sigma_s^2}{1 + \sigma_s^2 \boldsymbol{\alpha}^T \mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}}^{-1} \boldsymbol{\alpha}} \cdot \mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}}^{-1} \boldsymbol{\alpha} \\ &= \beta_s \mathbf{h}_{\mathrm{C}}, \end{aligned} \tag{3.47}$$

where

$$\beta_s = \frac{\sigma_s^2 \boldsymbol{\alpha}^T \mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}}^{-1} \boldsymbol{\alpha}}{1 + \sigma_s^2 \boldsymbol{\alpha}^T \mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}}}^{-1} \boldsymbol{\alpha}}. \tag{3.48}$$

The first point we can observe is that the Wiener filter is proportional to the Capon filter. The second point is that since the Capon filter is equal to the maximum SNR filter and this latter is specified up to a constant, the Wiener filter also maximizes the output SNR. In other words, with the model given in (3.21) the maximum SNR, MVDR, and Wiener filters are equivalent as far as the output SNR is concerned.

It is very important to understand that, contrary to the MVDR filter for example, the Wiener filter will distort the desired signal with a more general model (real room acoustic environment). It seems that it is the price to pay for noise reduction. Different aspects and properties of the Wiener filter were discussed in Chapter 2 for the single-channel case and the multichannel version will be studied in Chapter 5.

3.8 Response-Invariant Broadband Beamformers

In the previous sections, we have introduced many basic terminologies and widely-used concepts in beamforming. A number of techniques, including non-adaptive and adaptive ones, were discussed to form a desired beam pattern so as to recover a desired source signal from its observations corrupted by noise and competing sources. However, the aforementioned techniques are narrow-band in nature in the sense that the resulting beam characteristics, particularly the beamwidth, are a function of the signal frequency. To visualize the frequency dependency of these techniques, we plot in Fig. 3.7 a 3-dimensional beam pattern of a DS beamformer where the signal has a bandwidth of 3.7 kHz

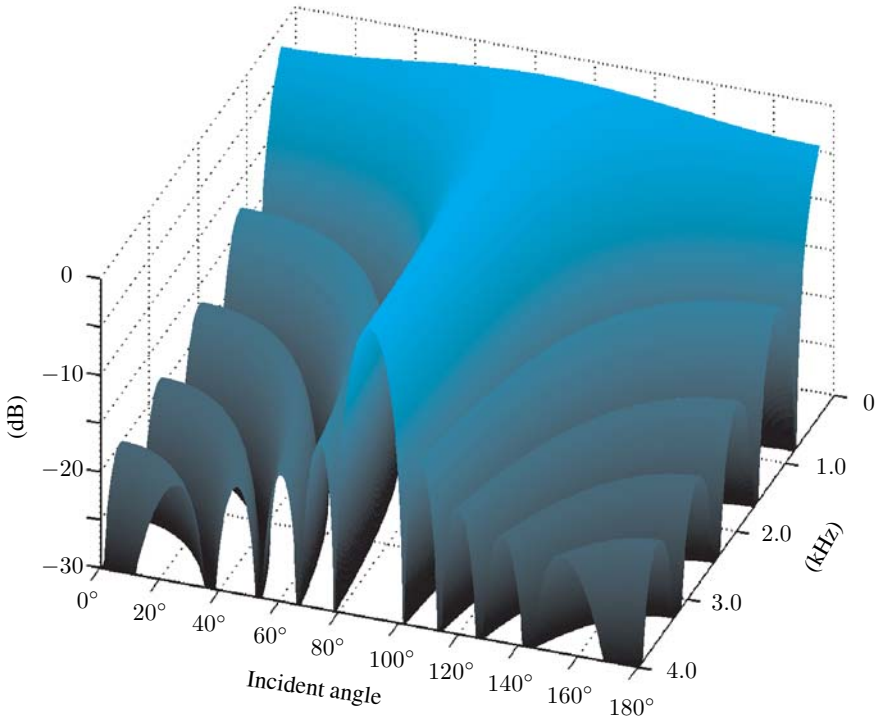


Fig. 3.7. 3-dimensional view of a DS beamformer: the array is an equispaced linear one with ten sensors; $d = 4$ cm; signal frequency is from 300 Hz to 4 kHz.

(from 300 Hz to 4 kHz). It can be clearly seen that the beampattern is not the same across the whole frequency band. Therefore, if we use such a beamformer for broadband signals like speech, and if the steering direction is different from the signal incident angle, the signal will be low-pass filtered. In addition, noise and interference signals will not be uniformly attenuated over its entire spectrum. This "spectral tilt" results in a disturbing artifact in the array output [224]. As a result, it is desirable to develop beamformers with constant beamwidth over frequency in order to deal with broadband information. The resulting techniques are called (response-invariant) broadband beamforming.

One way to obtain a broadband beamformer is to use harmonically nested subarrays [72], [73], [142]. Every subarray is linear and equally-spaced, and is designed for operating at a single frequency. But such a solution requires a large array with a great number of microphones even though subarrays may share sensors in the array. Another way to design a broadband beamformer based on classical narrowband techniques is to perform narrowband decomposition as illustrated in Fig. 3.8, and design narrowband beamformers independently at each frequency. The broadband output is synthesized from

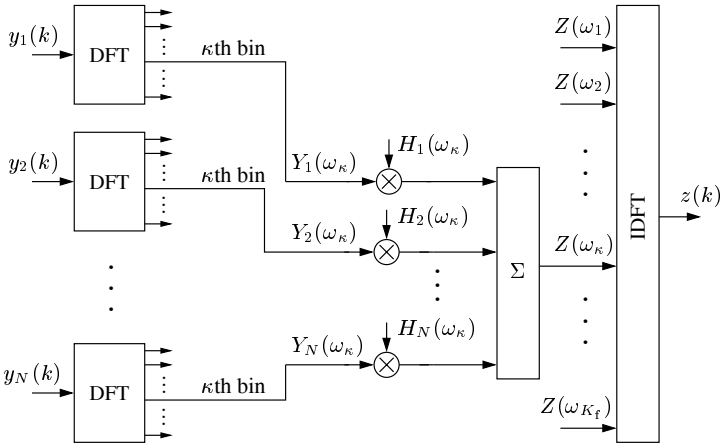


Fig. 3.8. The structure of a frequency-domain broadband beamformer.

the outputs of narrowband beamformers. Figure 3.9 presents an example, where each subband beamformer is designed using the LS method discussed in Section 3.4.

The structure of a frequency-domain broadband beamformer as shown in Fig. 3.8 can be equivalently transformed into its time-domain counterpart shown in Fig. 3.10, where an FIR filter is applied to each sensor output, and the filtered sensor signals are summed together to form a single output. This is widely known as the filter-and-sum beamformer first developed by Frost in [76] although the original idea was not dealing with the broadband issue. Mathematically, a filter-and-sum beamformer can be written as

$$z(k) = \sum_{n=1}^{N} \mathbf{h}_n^T \mathbf{y}_n(k), \tag{3.49}$$

where

$$\begin{aligned} \mathbf{h}_n &= \left[h_{n,0}\ h_{n,1} \cdots h_{n,L_h-1} \right], \\ \mathbf{y}_n(k) &= \left[y_n(k)\ y_n(k-1) \cdots y_n(k-L_h+1) \right], \end{aligned}$$

$n = 1, 2, \ldots, N$, and L_h is the length of the beamforming filter. Now the beamforming problem becomes one of finding the desired filters $\mathbf{h}_n$.

The invention of the filter-and-sum beamformer has opened a new page in array signal processing. Not only that we can use this idea to design broadband beamformers [211], we also can use it to deal with reverberation, another distraction that is so difficult to cope with. About how to design the filters will be discussed in the following chapters. In the next section, we show a simple broadband design example for null steering.

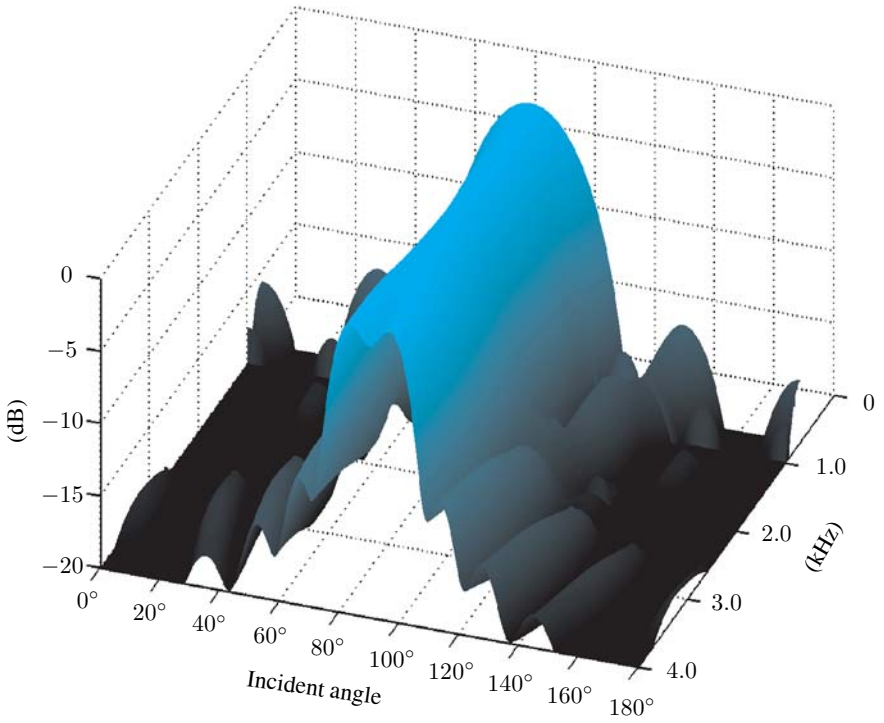


Fig. 3.9. 3-dimensional view of a LS broadband beamformer: the array is an equispaced linear one with ten sensors; $d = 4$ cm; signal frequency is from 300 Hz to 4 kHz.

3.9 Null-Steering Technique

We have shown that, if the noise signals are from a point source, both the maximum SNR and Capon filters place a null along the direction corresponding to the noise source. In this section, we discuss a more generic technique called null-steering, which originates from the ideas of sidelobe cancellers [51], [110], and generalized sidelobe canceller [31], [32], [94]. The motivation behind null-steering is to cancel one or multiple competing source (interference) signals propagating from known directions [47], [75], [87], [88]. As in the previous techniques, we consider an array system consisting of N elements. Unlike the signal model given in (3.1), here we assume that there are multiple sources in the wavefield, and the array outputs are expressed as

$$y_n(k) = \sum_{m=1}^{M} \alpha_{nm} s_m \left[k - t_m - \mathcal{F}_n(\tau_m)\right], \; n = 1, 2, \ldots, N, \tag{3.50}$$

where s_m, $m = 1, 2, \ldots, M$ $(M \leq N)$ are the source signals, α_{nm} are the attenuation factors due to propagation effects, t_m is the propagation time from

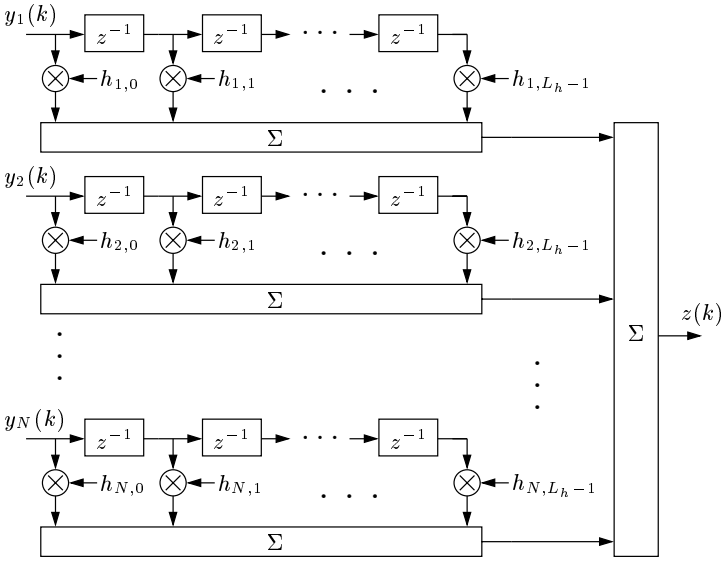


Fig. 3.10. The structure of a filter-and-sum beamformer.

the source s_m to sensor 1, τ_m is the relative delay between microphones 1 and 2 for the mth source, and $\mathcal{F}_n(\tau_m)$ is the relative delay between microphones 1 and n for the mth source with $\mathcal{F}_1(\tau_m) = 0$ and $\mathcal{F}_2(\tau_m) = \tau_m$. Again, we assume that τ_m and $\mathcal{F}_n(\cdot)$ are known. Without loss of generality, we consider the first source, s_1, as the desired signal and the $M-1$ remaining sources, $s_2, \ldots, s_M$, as the interferers.

Expression (3.50) can be rewritten in a more convenient way:

$$y_n(k) = \sum_{m=1}^{M} \mathbf{g}_{nm}^T \mathbf{s}_m(k - t_m), \ n = 1, 2, \ldots, N, \tag{3.51}$$

where

$$\mathbf{g}_{nm} = \left[0 \cdots 0 \ \alpha_{nm} \ 0 \cdots 0 \right]^T$$

is a filter of length L_g whose $[\mathcal{F}_n(\tau_m) + 1]$th component is equal to α_{nm}, and

$$\mathbf{s}_m(k - t_m) = [s_m(k - t_m) \ s_m(k - t_m - 1) \ \cdots \ s_m\left[k - t_m - \mathcal{F}_n(\tau_m)\right] \cdots \ s_m(k - t_m - L_g + 1)]^T.$$

The objective of a null-steering algorithm is to find N filters

$$\mathbf{h}_n = \left[h_{n,0} \ h_{n,1} \cdots h_{n,L_h-1} \right]^T, \ n = 1, 2, \ldots, N,$$

of length L_h such that the output of the beamformer

$$z(k) = \sum_{n=1}^{N} \mathbf{h}_n^T \mathbf{y}_n(k), \tag{3.52}$$

with

$$\mathbf{y}_n(k) = \left[y_n(k)\ y_n(k-1) \cdots y_n(k-L_h+1) \right]^T,\ n = 1, 2, \ldots, N,$$

is a good approximation of the desired source, s_1, and such that the $M-1$ interferers, $s_2, \ldots, s_M$, are attenuated as much as possible. This is a broadband processing approach.

Let us rewrite the microphone signals of (3.51) in a vector/matrix form:

$$\mathbf{y}_n(k) = \sum_{m=1}^{M} \mathbf{G}_{nm} \mathbf{s}_{L,m}(k - t_m),\ n = 1, 2, \ldots, N, \tag{3.53}$$

where

$$\mathbf{G}_{nm} = \begin{bmatrix} \mathbf{g}_{nm}^T & 0 & 0 & \cdots & 0 \\ 0 & \mathbf{g}_{nm}^T & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & \mathbf{g}_{nm}^T \end{bmatrix},$$
$$n = 1, 2, \ldots, N,\ m = 1, 2, \ldots, M,$$

is a Sylvester matrix of size $L_h \times L$, with $L = L_g + L_h - 1$, and

$$\mathbf{s}_{L,m}(k - t_m) = \left[s_m(k - t_m)\ s_m(k - t_m - 1) \cdots s_m(k - t_m - L + 1) \right]^T,$$
$$m = 1, 2, \ldots, M.$$

Substituting (3.53) into (3.52), we find that

$$z(k) = \sum_{m=1}^{M} \left[\sum_{n=1}^{N} \mathbf{h}_n^T \mathbf{G}_{nm} \right] \mathbf{s}_{L,m}(k - t_m). \tag{3.54}$$

From the above expression, we see that in order to perfectly recover $s_1(k)$ the following M conditions have to be satisfied:

$$\sum_{n=1}^{N} \mathbf{G}_{n1}^T \mathbf{h}_n = \mathbf{u}, \tag{3.55}$$

$$\sum_{n=1}^{N} \mathbf{G}_{nm}^T \mathbf{h}_n = \mathbf{0}_{L \times 1},\ m = 2, \ldots, M, \tag{3.56}$$

where

$$\mathbf{u} = \left[1\ 0 \cdots 0\ 0 \right]^T \tag{3.57}$$

is a vector of length L. In matrix/vector form, the M previous conditions are

$$\mathbf{G}^T\mathbf{h} = \mathbf{u}', \tag{3.58}$$

where

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}_{11} & \mathbf{G}_{12} & \cdots & \mathbf{G}_{1M} \\ \mathbf{G}_{21} & \mathbf{G}_{22} & \cdots & \mathbf{G}_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{G}_{N1} & \mathbf{G}_{N2} & \cdots & \mathbf{G}_{NM} \end{bmatrix}_{NL_h \times ML},$$

$$\mathbf{h} = \begin{bmatrix} \mathbf{h}_1^T & \mathbf{h}_2^T & \cdots & \mathbf{h}_N^T \end{bmatrix}^T,$$

$$\mathbf{u}' = \begin{bmatrix} \mathbf{u}^T & \mathbf{0}_{L\times 1}^T & \cdots & \mathbf{0}_{L\times 1}^T \end{bmatrix}^T.$$

Depending on the values of N and M, we have two cases, i.e., $N = M$ and $N > M$.

Case 1: $N = M$.

In this case, $ML = NL = NL_h + NL_g - N$. Since $L_g > 1$, we have $ML > NL_h$. This means that the number of rows of $\mathbf{G}^T$ is always larger than its number of columns. Assuming that the matrix $\mathbf{G}^T$ has full column rank, we can take the least-squares (LS) solution for the linear system (3.58), which is

$$\mathbf{h}_{\text{LS}} = \left[\mathbf{G}\mathbf{G}^T\right]^{-1}\mathbf{G}\mathbf{u}'. \tag{3.59}$$

Case 2: $N > M$.

When we have more microphones than sources, all 3 cases $ML > NL_h$, $ML = NL_h$, and $ML < NL_h$ can occur depending on the values of L_g and L_h. If $ML > NL_h$, then we can still take the LS solution as given in (3.59). If $ML = NL_h$, we have an exact solution:

$$\mathbf{h}_{\text{E}} = \left[\mathbf{G}^T\right]^{-1}\mathbf{u}'. \tag{3.60}$$

Finally, for the last case $ML < NL_h$, we can take the minimum-norm solution:

$$\mathbf{h}_{\text{MN}} = \mathbf{G}\left[\mathbf{G}^T\mathbf{G}\right]^{-1}\mathbf{u}'. \tag{3.61}$$

More sophisticated solutions for interference suppression are described in Chapter 7 in the general multiple-input/multiple-output framework. But before leaving this chapter, we will discuss in the next section the conditions that are required to recover the desired signal.

3.10 Microphone Array Pattern Function

Having presented the basic techniques for narrowband and broadband beamforming, we are now in a position to discuss the array pattern, which can be

used to examine the beamformer's response to an arbitrary propagation field just as the frequency response of a temporal filter can be used to analyze its response to an arbitrary signal [34]. In the narrowband situation, two forms of array pattern have been studied: beam pattern and steered response. The term beam pattern, as has been used throughout the text, characterizes the array's input-output behavior when the beamformer is steered to a specific direction. It can be used to analyze how the array output is affected by signals different from the focused one. In comparison, the steered response measures the beamformer's output when it is scanned by systematically varying the steering angle from 0° to 180°. (It is also of interest, occasionally, to measure the steered response from 0° to 360°.)

Both beam pattern and steered response are very useful in analyzing narrowband beamformers. However, they tend to be inadequate to characterize the performance of broadband beamformers in reverberant environments. In this situation things are less obvious to understand than the narrowband case where only a monochromatic plane wave is considered. In this section, we try to derive another form of array pattern for two different signal models with a broadband source, which is useful in analyzing microphone arrays.

3.10.1 First Signal Model

Consider a white noise source (since it covers the whole spectrum), s, with variance $\sigma_s^2 = 1$. In this first signal model, we consider that the nth sensor signal can be written as

$$y_n(k) = s\left[k - t - \mathcal{F}_n(\tau_s)\right],\ n = 1, 2, \ldots, N, \tag{3.62}$$

where τ_s is the relative delay between microphones 1 and 2 for the source signal s. (For convenience, we slightly changed the notation for the relative delay by adding a subscript s to it.) We assume that the signal arrives first at microphone 1. We examine the far-field case and a linear equispaced array where $\mathcal{F}_n(\tau_s) = (n-1)\tau_s$. As explained in the previous section, (3.62) can be rewritten as

$$y_n(k) = \mathbf{g}^T\left[\mathcal{F}_n(\tau_s)\right]\mathbf{s}(k-t),\ n = 1, 2, \ldots, N, \tag{3.63}$$

where

$$\mathbf{g}\left[\mathcal{F}_n(\tau_s)\right] = \left[0 \cdots 0\ 1\ 0 \cdots 0\right]^T$$

is a filter of length $L_g \geq \mathcal{F}_N(\tau_s) + 1$ whose $[\mathcal{F}_n(\tau_s) + 1]$th component is equal to 1, and

$$\begin{aligned}\mathbf{s}(k-t) = [&s(k-t)\ s(k-t-1)\ \cdots\ s\left[k - t - \mathcal{F}_n(\tau_s)\right] \\ &\cdots\ s(k-t-L_g+1)]^T.\end{aligned}$$

Consider the N filters

$$\mathbf{h}\left[\mathcal{F}_n(\tau)\right] = \frac{1}{N}\left[0 \cdots 0\ 1\ 0 \cdots 0\right]^T,\ n = 1, 2, \ldots, N,$$

of length L_h whose $[\mathcal{F}_n(\tau) + 1]$th component is equal to $1/N$. The output of the beamformer is

$$\begin{aligned} z(k) &= \sum_{n=1}^{N} \mathbf{h}^T\left[\mathcal{F}_{N+1-n}(\tau)\right]\mathbf{y}_n(k) \\ &= \frac{1}{N}\sum_{n=1}^{N} y_n\left[k - \mathcal{F}_{N+1-n}(\tau)\right] \\ &= \frac{1}{N}\sum_{n=1}^{N} s\left[k - t - \mathcal{F}_{N+1-n}(\tau) - \mathcal{F}_n(\tau_s)\right]. \end{aligned} \tag{3.64}$$

We see that for $\tau = \tau_s$, $z(k) = s\left[k - t - \mathcal{F}_N(\tau_s)\right]$.

Expression (3.64) can also be put in the following form:

$$z(k) = \mathbf{h}^T(\tau)\mathbf{y}(k), \tag{3.65}$$

where

$$\begin{aligned} \mathbf{h}(\tau) &= \left[\mathbf{h}^T\left[\mathcal{F}_N(\tau)\right]\ \mathbf{h}^T\left[\mathcal{F}_{N-1}(\tau)\right] \cdots \mathbf{h}^T\left[\mathcal{F}_1(\tau)\right]\right]^T, \\ \mathbf{y}(k) &= \left[\mathbf{y}_1^T(k)\ \mathbf{y}_2^T(k) \cdots \mathbf{y}_N^T(k)\right]^T. \end{aligned}$$

Also

$$\mathbf{y}_n(k) = \mathbf{G}\left[\mathcal{F}_n(\tau_s)\right]\mathbf{s}_L(k - t),\ n = 1, 2, \ldots, N, \tag{3.66}$$

where

$$\mathbf{G}\left[\mathcal{F}_n(\tau_s)\right] = \begin{bmatrix} \mathbf{g}^T\left[\mathcal{F}_n(\tau_s)\right] & 0 & 0 & \cdots & 0 \\ 0 & \mathbf{g}^T\left[\mathcal{F}_n(\tau_s)\right] & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & \mathbf{g}^T\left[\mathcal{F}_n(\tau_s)\right] \end{bmatrix},$$
$$n = 1, 2, \ldots, N,$$

is a Sylvester matrix of size $L_h \times L$, with $L = L_g + L_h - 1$, and

$$\mathbf{s}_L(k - t) = \left[s(k - t)\ s(k - t - 1) \cdots s(k - t - L + 1)\right]^T.$$

We deduce from the previous expressions that

$$z(k) = \mathbf{h}^T(\tau)\mathbf{G}(\tau_s)\mathbf{s}_L(k - t), \tag{3.67}$$

where

$$\mathbf{G}(\tau_s) = \begin{bmatrix} \mathbf{G}\left[\mathcal{F}_1(\tau_s)\right] \\ \mathbf{G}\left[\mathcal{F}_2(\tau_s)\right] \\ \vdots \\ \mathbf{G}\left[\mathcal{F}_N(\tau_s)\right] \end{bmatrix}_{NL_h \times L}.$$

The matrix $\mathbf{G}(\tau_s)$ can be seen as the steering matrix, which incorporates all the information of the desired signal position. Therefore, the variance of the output beamformer is

$$E\left[z^2(k)\right] = \left\|\mathbf{G}^T(\tau_s)\mathbf{h}(\tau)\right\|_2^2. \tag{3.68}$$

We define the microphone array pattern function as

$$\mathcal{A}(\tau) = 1 - \left\|\mathbf{G}^T(\tau_s)\mathbf{h}(\tau) - \mathbf{u}\left[\mathcal{F}_N(\tau)\right]\right\|_2, \tag{3.69}$$

with $\mathcal{A}(\tau_s) = 1$, and

$$\mathbf{u}\left[\mathcal{F}_N(\tau)\right] = \left[0 \cdots 0\ 1\ 0 \cdots 0\right]^T$$

is a vector of length L whose $[\mathcal{F}_N(\tau)+1]$th component is equal to 1. In the search of the beamforming filter, we expect that there is one unique filter $\mathbf{h}(\tau)$ such that $\mathcal{A}(\tau) = 1$. If there are several such filters, that would indicate spatial aliasing problems.

3.10.2 Second Signal Model

Again consider a white noise source, s, with variance $\sigma_s^2 = 1$. In this subsection, we choose the signal model:

$$\begin{aligned} y_n(k) &= g_n * s(k) \\ &= \mathbf{g}_n^T \mathbf{s}(k), \end{aligned} \tag{3.70}$$

where $*$ stands for convolution and g_n is the acoustic impulse response of length L_g from the source $s(k)$ to the nth microphone. Using the previous notation, it is easy to see that

$$z(k) = \mathbf{h}^T \mathbf{G}\mathbf{s}_L(k), \tag{3.71}$$

so that

$$E\left[z^2(k)\right] = \left\|\mathbf{G}^T\mathbf{h}\right\|_2^2 \tag{3.72}$$

and the microphone array pattern function is

$$\mathcal{A}(\mathbf{h}) = 1 - \left\|\mathbf{G}^T\mathbf{h} - \mathbf{u}\right\|_2 \tag{3.73}$$

where $\mathbf{G}$ is the steering matrix (containing all the impulse responses from the desired source to the N microphones) and $\mathbf{u}$ is defined in (3.57).

Now let's take $L_h = (L_g - 1)/(N - 1)$ and assume that L_h is an integer, then the matrix $\mathbf{G}^T$ becomes a square one. To find a vector $\mathbf{h}$ such that $z(k) = s(k)$, we need to solve the linear system $\mathbf{G}^T\mathbf{h} = \mathbf{u}$. This solution is unique if $\mathbf{G}^T$ is full rank. In this case, there exists only one vector $\mathbf{h}$ such that $\mathcal{A}(\mathbf{h}) = 1$. If $\mathbf{G}^T$ is not full rank, which is equivalent to saying that the N polynomials formed from $g_1, g_2, \ldots, g_N$ share common zeroes, there will be two cases:

Case 1:

if $\mathbf{G}^T$ and the augmented matrix $[\mathbf{G}^T|\mathbf{u}]$ have the same rank, there will be more than one vector $\mathbf{h}$ such that $\mathcal{A}(\mathbf{h}) = 1$. As a result, we should expect spatial aliasing as what we can experience in narrowband situations.

Case 2:

if the rank of $\mathbf{G}^T$ is less than that of the augmented matrix $[\mathbf{G}^T|\mathbf{u}]$, the linear system $\mathbf{G}^T\mathbf{h} = \mathbf{u}$ has no solution. As a result, we are not able to recover the source signal. This situation may happen when the array does not have enough aperture and there is not adequate diversity among the microphone channels. An extreme example is when all the sensors are co-positioned. Then the array system degenerates to the single-channel one and apparently, it is impossible to recover the source signal with beamforming.

The above two cases suggest two requirements in array design: the spacing among sensors cannot be too large (as compared to the wavelength). Otherwise we will experience the spatial aliasing problem, which causes ambiguity in recovering the desired signal. On the other hand, the sensors cannot be too close. If they are too close, the array does not provide enough aperture for recovering the source signal.

3.11 Conclusions

This chapter reviewed the fundamental principles underlying conventional narrowband beamforming techniques, most of which were originally developed in the fields of radar and sonar. While the basic ideas in narrowband beamforming can be generalized to the design of broadband beamformers, directly applying a narrowband beamformer to broadband signals can create many issues such as colorizing the desired signal and spectrally tilting the ambient noise. In order to avoid signal distortion, it is indispensable to develop broadband beamformers that have constant beam characteristics over frequency. To this end, we discussed two approaches: subband decomposition and filter-and-sum. Theoretically, these two approaches are equivalent (one can be treated as the counterpart of the other in a different domain), though they may have different design and implementation advantages. Also discussed in this chapter is the array pattern, which is useful in examining the performance of beamforming and studying the conditions under which the desired signal can be recovered.

4

On the Use of the LCMV Filter in Room Acoustic Environments

4.1 Introduction

The linearly constrained minimum variance (LCMV) filter [76], also known as the Frost algorithm (named after O. L. Frost, even though he might not be the inventor), has been extremely popular in antenna arrays. It can be useful not only in microphone arrays for speech enhancement but also in communications, radar, and sonar. There are different ways to define the constraints that are inherently built in the structure of this algorithm. However, the basic idea behind this filter is to try to extract the desired signal coming from a specific direction while minimizing contributions to the output due to interfering signals and noise arriving from directions other than the direction of interest [216].

This chapter attempts to show in which conditions the LCMV filter can be used in room acoustic environments. In order to help the reader better understand how the LCMV filter works, we will present in Section 4.2 three mathematical models for which the LCMV filter is derived. Section 4.3 explains the LCMV filter with the simple anechoic model. Section 4.4 presents the Frost algorithm in the context of the more sophisticated (and also more realistic) reverberant model. Section 4.5 derives the LCMV filter for the more practical spatio-temporal model. Very often, an algorithm in the frequency domain gives better insights than its time-domain version. For this reason, we derive the Frost algorithm in the frequency domain in Section 4.6. Finally, we draw our conclusions in Section 4.7.

4.2 Signal Models

Before discussing how to use the LCMV filter, we need first to explain the mathematical models that can be employed to describe a room acoustic environment. These models will help us better understand how the LCMV filter

works, what are its potentials, and where are its limits. In the following, we will describe the anechoic, reverberant, and spatio-temporal models.

4.2.1 Anechoic Model

Suppose that we have an array consisting of N sensors, the anechoic model assumes that the signal picked up by each microphone is a delayed and attenuated version of the original source signal plus some additive noise. Mathematically, the received signals, at time k, are expressed as

$$\begin{aligned} y_n(k) &= \alpha_n s\left[k - t - \mathcal{F}_n(\tau)\right] + v_n(k) \\ &= x_n(k) + v_n(k), \end{aligned} \tag{4.1}$$

where $\alpha_n,\ n = 1, 2, \ldots, N$, are the attenuation factors due to propagation effects, $s(k)$ is the unknown source signal, t is the propagation time from the unknown source to sensor 1, $v_n(k)$ is an additive noise signal at the nth microphone, τ is the relative delay between microphones 1 and 2, and $\mathcal{F}_n(\tau)$ is the relative delay between microphones 1 and n with $\mathcal{F}_1(\tau) = 0$ and $\mathcal{F}_2(\tau) = \tau$. For example, in the far-field case (plane wave propagation) and for a linear equispaced array, we have

$$\mathcal{F}_n(\tau) = (n-1)\tau. \tag{4.2}$$

It is further assumed that $v_n(k)$ is a zero-mean Gaussian random process that is uncorrelated with $s(k)$.[1] It is also assumed that $s(k)$ is zero-mean and reasonably broadband.

4.2.2 Reverberant Model

Most of the rooms are reverberant which means that each sensor often receives a large number of echoes due to reflections of the wavefront from objects and room boundaries such as walls, ceiling, and floor [125]. In this model, the received signals are expressed as

$$\begin{aligned} y_n(k) &= g_n * s(k) + v_n(k) \\ &= x_n(k) + v_n(k), \end{aligned} \tag{4.3}$$

where g_n is the impulse response from the unknown source $s(k)$ to the nth microphone. Again, we assume that $s(k)$ is zero-mean, reasonably broadband, and uncorrelated with the additive noise $v_n(k)$. In a vector/matrix form, the signal model (4.3) can be rewritten as

$$y_n(k) = \mathbf{g}_n^T \mathbf{s}(k) + v_n(k),\ n = 1, 2, \ldots, N, \tag{4.4}$$

[1] The case where $v_n(k)$ is correlated with $s(k)$ is equivalent to the reverberant model.

where

$$\mathbf{g}_n = \left[\, g_{n,0}\ g_{n,1}\ \cdots\ g_{n,L_g-1} \,\right]^T,$$
$$\mathbf{s}(k) = \left[\, s(k)\ s(k-1)\ \cdots\ s(k-L_g+1) \,\right]^T,$$

and L_g is the length of the longest acoustic impulse responses among the N channels $g_n,\ n = 1, 2, \ldots, N$.

4.2.3 Spatio-Temporal Model

In this model, we exploit the spatial information of the unknown source as well as its temporal signature. Indeed, using the z-transform, the signal $x_n(k)$ in (4.3) can be rewritten as

$$X_n(z) = S(z)G_n(z),\ n = 1, 2, \ldots, N, \tag{4.5}$$

where $X_n(z)$, $S(z)$, and $G_n(z)$ are the z-transforms of $x_n(k)$, $s(k)$, and g_n, respectively, with $G_n(z) = \sum_{l=0}^{L_g-1} g_{n,l} z^{-l}$. From (4.5) it is easy to verify that the signals $x_n(k),\ n = 2, 3, \ldots, N$, are related to $x_1(k)$ as follows:

$$\begin{aligned} X_n(z) &= \frac{G_n(z)}{G_1(z)} X_1(z) \\ &= W_n(z)X_1(z),\ n = 2, 3, \ldots, N, \end{aligned} \tag{4.6}$$

where $W_n(z)$ is an infinite impulse response (IIR) filter. We will assume that this IIR filter can be well approximated by a large FIR filter. With this assumption, we can rewrite (4.6) in the time domain:

$$\mathbf{x}_n(k) = \mathbf{W}_n \mathbf{x}_1(k),\ n = 2, 3, \ldots, N, \tag{4.7}$$

where

$$\mathbf{x}_n(k) = \left[\, x_n(k)\ x_n(k-1)\ \cdots\ x_n(k-L_h+1) \,\right]^T,\ n = 1, 2, \ldots, N,$$

and $\mathbf{W}_n$ is an $L_h \times L_h$ matrix.

With these three models in mind, we will derive and study the LCMV filter for dereverberation and noise reduction for each one of them.

4.3 The LCMV Filter with the Anechoic Model

In the anechoic model, the relative delay [$\mathcal{F}_n(\tau)$ or τ)] needs to be known or accurately estimated. Fortunately, many robust methods exist to estimate τ from a set of microphones; see for examples [40], [57] and references therein. The knowledge of this relative delay allows us to time-align the received signals

in the array aperture, such that the desired signal becomes coherent after this processing

$$\begin{aligned} y_{\mathrm{a},n}(k) &= y_n\left[k + \mathcal{F}_n(\tau)\right] \\ &= \alpha_n s(k-t) + v_{\mathrm{a},n}(k),\ n = 1, 2, \ldots, N, \end{aligned} \tag{4.8}$$

where

$$v_{\mathrm{a},n}(k) = v_n\left[k + \mathcal{F}_n(\tau)\right].$$

This alignment has also the potential to somewhat misalign the noise at the sensors thereby reducing its spatial coherence. So even in the presence of a unique point-noise source, this may not appear that way anymore at the sensors as long as the source and the noise signals come from different positions.

It is now more convenient to work with the samples $y_{\mathrm{a},n}(k)$ or the $N \times 1$ vector

$$\mathbf{y}_{\mathrm{a}}(k) = s(k-t)\boldsymbol{\alpha} + \mathbf{v}_{\mathrm{a}}(k), \tag{4.9}$$

where

$$\begin{aligned} \mathbf{y}_{\mathrm{a}}(k) &= \left[\, y_{\mathrm{a},1}(k)\ y_{\mathrm{a},2}(k) \cdots y_{\mathrm{a},N}(k) \,\right]^T, \\ \mathbf{v}_{\mathrm{a}}(k) &= \left[\, v_{\mathrm{a},1}(k)\ v_{\mathrm{a},2}(k) \cdots v_{\mathrm{a},N}(k) \,\right]^T, \\ \boldsymbol{\alpha} &= \left[\, \alpha_1\ \alpha_2 \cdots \alpha_N \,\right]^T. \end{aligned}$$

If we consider the most recent L_h samples of each microphone, we can form the $NL_h \times 1$ vector:

$$\mathbf{y}_{\mathrm{a},NL_h}(k) = \mathbf{x}_{\mathrm{a},NL_h}(k) + \mathbf{v}_{\mathrm{a},NL_h}(k), \tag{4.10}$$

where

$$\begin{aligned} \mathbf{y}_{\mathrm{a},NL_h}(k) &= \left[\, \mathbf{y}_{\mathrm{a}}^T(k)\ \mathbf{y}_{\mathrm{a}}^T(k-1) \cdots \mathbf{y}_{\mathrm{a}}^T(k-L_h+1) \,\right]^T, \\ \mathbf{x}_{\mathrm{a},NL_h}(k) &= \left[\, s(k-t)\boldsymbol{\alpha}^T\ s(k-t-1)\boldsymbol{\alpha}^T \cdots s(k-t-L_h+1)\boldsymbol{\alpha}^T \,\right]^T, \\ \mathbf{v}_{\mathrm{a},NL_h}(k) &= \left[\, \mathbf{v}_{\mathrm{a}}^T(k)\ \mathbf{v}_{\mathrm{a}}^T(k-1) \cdots \mathbf{v}_{\mathrm{a}}^T(k-L_h+1) \,\right]^T. \end{aligned}$$

The aim here is to find an array filter, $\mathbf{h}$, of length NL_h in such a way that the signal at its output is equal (or close) to $\sum_{l=0}^{L_h-1} u_l s(k-t-l)$, where the u_l are some chosen numbers. These coefficients help shaping the spectrum of $s(k)$.

First, L_h constraints need to be found in order to have

$$\mathbf{h}^T \mathbf{x}_{\mathrm{a},NL_h}(k) = \sum_{l=0}^{L_h-1} u_l s(k-t-l). \tag{4.11}$$

It is clear from (4.11) that the L_h constraints should be

$$\mathbf{c}_{\boldsymbol{\alpha},l}^T\mathbf{h} = u_l,\ l = 0, 1, \ldots, L_h - 1, \tag{4.12}$$

where

$$\mathbf{c}_{\boldsymbol{\alpha},l} = \Big[\underbrace{\mathbf{0}_{N\times 1}^T \cdots \mathbf{0}_{N\times 1}^T}_{l\text{th group}}\ \underbrace{\boldsymbol{\alpha}^T}_{l\text{th group}}\ \underbrace{\mathbf{0}_{N\times 1}^T \cdots \mathbf{0}_{N\times 1}^T}_{l\text{th group}}\Big]^T$$

is the lth constraint vector of length NL_h. The constraints in (4.12) can be put in a matrix form:

$$\mathbf{C}_{\boldsymbol{\alpha}}^T\mathbf{h} = \mathbf{u}, \tag{4.13}$$

with

$$\begin{aligned}\mathbf{C}_{\boldsymbol{\alpha}} &= \left[\mathbf{c}_{\boldsymbol{\alpha},0}\ \mathbf{c}_{\boldsymbol{\alpha},1} \cdots \mathbf{c}_{\boldsymbol{\alpha},L_h-1}\right],\\ \mathbf{u} &= \left[u_0\ u_1 \cdots u_{L_h-1}\right]^T.\end{aligned}$$

The vector $\mathbf{u}$ contains the coefficients of an FIR filter that maintains a chosen frequency response of the desired signal $s(k)$ and the constraint matrix, $\mathbf{C}_{\boldsymbol{\alpha}}$, is of size $NL_h \times L_h$.

Then the second step consists of minimizing the total array output power

$$\mathbf{h}^T\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}},NL_h}\mathbf{h},$$

where

$$\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}},NL_h} = E\left[\mathbf{y}_{\mathrm{a},NL_h}(k)\mathbf{y}_{\mathrm{a},NL_h}^T(k)\right]$$

is the $NL_h \times NL_h$ correlation matrix of the microphone signals. Therefore, to find the optimal filter we need to solve the optimization problem [76]:

$$\min_{\mathbf{h}} \mathbf{h}^T\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}},NL_h}\mathbf{h} \quad \text{subject to} \quad \mathbf{C}_{\boldsymbol{\alpha}}^T\mathbf{h} = \mathbf{u}. \tag{4.14}$$

Expression (4.14) is, indeed, easy to solve and its optimal solution is

$$\mathbf{h}_{\mathrm{A}} = \mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}},NL_h}^{-1}\mathbf{C}_{\boldsymbol{\alpha}}\left(\mathbf{C}_{\boldsymbol{\alpha}}^T\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}},NL_h}^{-1}\mathbf{C}_{\boldsymbol{\alpha}}\right)^{-1}\mathbf{u}, \tag{4.15}$$

where the subscript "A" indicates an anechoic signal model. In (4.15), we assume that $\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}},NL_h}$ has full rank and a necessary condition for that to be true is that the correlation matrix of the noise

$$\mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}},NL_h} = E\left[\mathbf{v}_{\mathrm{a},NL_h}(k)\mathbf{v}_{\mathrm{a},NL_h}^T(k)\right]$$

is positive definite. Let us show that. We can write the correlation matrix of the microphone signals as

$$\begin{aligned}\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}},NL_h} &= \mathbf{R}_{x_{\mathrm{a}}x_{\mathrm{a}},NL_h} + \mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}},NL_h} \\ &= E\left\{[\mathbf{s}(k-t)\otimes\boldsymbol{\alpha}]\,[\mathbf{s}(k-t)\otimes\boldsymbol{\alpha}]^T\right\} + \mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}},NL_h} \\ &= \mathbf{R}_{ss,L_h}\otimes\left(\boldsymbol{\alpha}\boldsymbol{\alpha}^T\right) + \mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}},NL_h}, \end{aligned} \tag{4.16}$$

where $\mathbf{R}_{ss,L_h} = E\left[\mathbf{s}(k-t)\mathbf{s}^T(k-t)\right]$ is the $L_h \times L_h$ correlation matrix, assumed to have full rank, of the signal

$$\mathbf{s}(k-t) = \left[\, s(k-t)\; s(k-t-1)\;\cdots\; s(k-t-L_h+1)\,\right]^T,$$

and $\otimes$ is the Kronecker product [91]. From this well-known property

$$\begin{aligned}\mathrm{rank}\left[\mathbf{R}_{ss,L_h}\otimes\left(\boldsymbol{\alpha}\boldsymbol{\alpha}^T\right)\right] &= \left[\mathrm{rank}\left(\mathbf{R}_{ss,L_h}\right)\right]\left[\mathrm{rank}\left(\boldsymbol{\alpha}\boldsymbol{\alpha}^T\right)\right] \\ &= L_h, \end{aligned} \tag{4.17}$$

it is clear that the $NL_h \times NL_h$ correlation matrix $\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}},NL_h}$ can be full rank only if $\mathbf{R}_{v_{\mathrm{a}}v_{\mathrm{a}},NL_h}$ is also full rank since the rank of $\mathbf{R}_{x_{\mathrm{a}}x_{\mathrm{a}},NL_h}$ is equal to L_h.

If the noise is correlated with the source signal,[2] we can see from (4.14) that risks are very high to cancel portions of $s(k)$ and there is no easy fix for this crucial problem [108].

Two particular interesting cases can be deduced from the LCMV filter:

- If we take $L_h = 1$ and $\mathbf{h}_{\mathrm{A}} = \mathbf{1}/N$, where $\mathbf{1}$ is a vector of N ones, we get the classical delay-and-sum beamformer [216].
- If we take $L_h = 1$ and $u_0 = 1$, we obtain the minimum variance distortionless response (MVDR) filter due to Capon [35]:

$$\mathbf{h}_{\mathrm{A}} = \frac{\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}}^{-1}\boldsymbol{\alpha}}{\boldsymbol{\alpha}^T\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}}^{-1}\boldsymbol{\alpha}}, \tag{4.18}$$

  where $\mathbf{h}_{\mathrm{A}}$ is a filter of length N and $\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}} = E\left[\mathbf{y}_{\mathrm{a}}(k)\mathbf{y}_{\mathrm{a}}^T(k)\right]$. Therefore, $\mathbf{h}_{\mathrm{A}}^T\mathbf{y}_{\mathrm{a}}(k)$ will be a good estimate of the sample $s(k-t)$.

For both the LCMV and MVDR filters a good estimator of the vector $\boldsymbol{\alpha}$ is required. Although several techniques exist like the one based on blind identification, the accuracy may not be enough in practice, so this problem is still a very open one. Another possible simple estimator is based on the maximum eigenvector of the matrix $\mathbf{R}_{y_{\mathrm{a}}y_{\mathrm{a}}}$ as explained in [57]. Moreover, to make the anechoic model more realistic, we need to assume that the desired source and the noise are correlated in (4.8). As a result, cancellation of the desired signal is unavoidable with this model.

[2] This scenario models the reverberation.

4.4 The LCMV Filter with the Reverberant Model

In this section we suppose that the N impulse responses from the desired source to the microphones are known (or can be estimated) and are stationary.

We consider N array filters $\mathbf{h}_n,\ n = 1, 2, \ldots, N$, of length L_h. The microphone signals [eq. (4.4)] can be rewritten in the following form:

$$\mathbf{y}_n(k) = \mathbf{G}_n \mathbf{s}_L(k) + \mathbf{v}_n(k),\ n = 1, 2, \ldots, N, \tag{4.19}$$

where

$$\begin{aligned}
\mathbf{y}_n(k) &= \left[y_n(k)\ y_n(k-1) \cdots y_n(k-L_h+1) \right]^T, \\
\mathbf{v}_n(k) &= \left[v_n(k)\ v_n(k-1) \cdots v_n(k-L_h+1) \right]^T, \\
\mathbf{s}_L(k) &= \left[s(k)\ s(k-1) \cdots s(k-L+1) \right]^T,
\end{aligned}$$

and

$$\mathbf{G}_n = \begin{bmatrix}
g_{n,0} & \cdots & g_{n,L_g-1} & 0 & 0 & \cdots & 0 \\
0 & g_{n,0} & \cdots & g_{n,L_g-1} & 0 & \cdots & 0 \\
\vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\
0 & 0 & \cdots & 0 & g_{n,0} & \cdots & g_{n,L_g-1}
\end{bmatrix}$$

is a Sylvester matrix of size $L_h \times L$, with $L = L_h + L_g - 1$.

If we concatenate the N observation vectors together, we get:

$$\begin{aligned}
\mathbf{y}(k) &= \left[\mathbf{y}_1^T(k)\ \mathbf{y}_2^T(k) \cdots \mathbf{y}_N^T(k) \right]^T \\
&= \mathbf{G}\mathbf{s}_L(k) + \mathbf{v}(k),
\end{aligned} \tag{4.20}$$

where

$$\mathbf{G} = \begin{bmatrix} \mathbf{G}_1 \\ \mathbf{G}_2 \\ \vdots \\ \mathbf{G}_N \end{bmatrix}_{NL_h \times L},$$

$$\mathbf{v}(k) = \left[\mathbf{v}_1^T(k)\ \mathbf{v}_2^T(k) \cdots \mathbf{v}_N^T(k) \right]^T.$$

The $NL_h \times NL_h$ covariance matrix corresponding to $\mathbf{y}(k)$ is

$$\mathbf{R}_{yy} = E\left[\mathbf{y}(k)\mathbf{y}^T(k) \right] = \mathbf{G}\mathbf{R}_{ss}\mathbf{G}^T + \mathbf{R}_{vv}, \tag{4.21}$$

with $\mathbf{R}_{ss} = E\left[\mathbf{s}_L(k)\mathbf{s}_L^T(k) \right]$ and $\mathbf{R}_{vv} = E\left[\mathbf{v}(k)\mathbf{v}^T(k) \right]$. We assume that $\mathbf{R}_{yy}$ is invertible, which implies that $\mathbf{R}_{vv}$ is positive definite or the matrix $\mathbf{G}\mathbf{R}_{ss}\mathbf{G}^T$ is full rank.

With all this information, the LCMV filter is obtained by solving the following optimization problem [18]:

$$\min_{\mathbf{h}} \mathbf{h}^T \mathbf{R}_{yy} \mathbf{h} \quad \text{subject to} \quad \mathbf{G}^T \mathbf{h} = \mathbf{u} \tag{4.22}$$

where

$$\mathbf{h} = \left[\mathbf{h}_1^T \ \mathbf{h}_2^T \ \cdots \ \mathbf{h}_N^T \right]^T$$

and

$$\mathbf{u} = \left[1 \ 0 \cdots 0 \right]^T$$

is a vector of length L whose first component is equal to 1 while all others are zeroes. In (4.22), the L constraints are necessary for the dereverberation of the signal of interest while the minimization is required to reduce the noise.

The optimal solution to (4.22) is

$$\mathbf{h}_{\mathrm{R}} = \mathbf{R}_{yy}^{-1} \mathbf{G} \left(\mathbf{G}^T \mathbf{R}_{yy}^{-1} \mathbf{G} \right)^{-1} \mathbf{u}, \tag{4.23}$$

where the subscript "R" indicates a reverberant signal model. Assume that $\mathbf{R}_{vv}$ and $\mathbf{R}_{yy}$ are positive definite, a necessary condition for $\left(\mathbf{G}^T \mathbf{R}_{yy}^{-1} \mathbf{G} \right)$ to be nonsingular (in order that $\mathbf{h}_{\mathrm{R}}$ exists) is to have $NL_h \geq L$, which implies that

$$L_h \geq \frac{L_g - 1}{N - 1}. \tag{4.24}$$

The other condition for the matrix $\left(\mathbf{G}^T \mathbf{R}_{yy}^{-1} \mathbf{G} \right)$ to be nonsingular is that $\mathbf{G}$ has full column rank, which is equivalent to saying that the N polynomials formed from $g_1, g_2, \ldots, g_N$ share no common zeroes. If indeed they have common zeroes, the constraints in (4.22) should be changed such that the vector $\mathbf{u}$ will contain the coefficients of the polynomial of the greatest common divisor of $g_1, g_2, \ldots, g_N$. As a result, dereverberation is possible up to a filtering operation.

An important thing to observe from (4.24) is that the minimum value required for the length of the filters $\mathbf{h}_{\mathrm{R},n}$, $n = 1, 2, \ldots, N$, decreases as the number of microphones increases. As a consequence, the LCMV filter has the potential to significantly reduce the effect of the background noise with a large number of microphones.

If we take the minimum value for L_h, i.e., $L_h = (L_g - 1)/(N - 1)$ and assume that L_h is an integer, $\mathbf{G}$ turns to a square matrix and (4.23) becomes:

$$\mathbf{h}_{\mathrm{R}} = \left[\mathbf{G}^T \right]^{-1} \mathbf{u}, \tag{4.25}$$

which is the MINT method [166]. Taking the minimum length will only dereverberate the signal of interest without any noise reduction. As we increase L_h from its minimum value, the degrees of freedom increase as well for better noise reduction.

Let us show now that minimizing the background noise without distorting the desired signal is equivalent to minimizing the total array output power with the same constraint. Indeed, using the constraint and (4.21), we see that $\mathbf{h}^T\mathbf{R}_{yy}\mathbf{h} = \sigma_s^2 + \mathbf{h}^T\mathbf{R}_{vv}\mathbf{h}$, where σ_s^2 is the variance of $s(k)$, which is equivalent to minimizing the background noise without distorting the desired signal. A more rigorous way of showing this is by applying the matrix inversion lemma to (4.21),

$$\left(\mathbf{G}\mathbf{R}_{ss}\mathbf{G}^T + \mathbf{R}_{vv}\right)^{-1} = \mathbf{R}_{vv}^{-1} - \mathbf{R}_{vv}^{-1}\mathbf{G}\left(\mathbf{G}^T\mathbf{R}_{vv}^{-1}\mathbf{G} + \mathbf{R}_{ss}^{-1}\right)^{-1}\mathbf{G}^T\mathbf{R}_{vv}^{-1} \tag{4.26}$$

and the identity

$$\begin{aligned}\left(\mathbf{G}^T\mathbf{R}_{vv}^{-1}\mathbf{G}\right)^{-1} - \left(\mathbf{G}^T\mathbf{R}_{vv}^{-1}\mathbf{G} + \mathbf{R}_{ss}^{-1}\right)^{-1} = \\ \left(\mathbf{G}^T\mathbf{R}_{vv}^{-1}\mathbf{G}\right)^{-1}\left[\mathbf{R}_{ss} + \left(\mathbf{G}^T\mathbf{R}_{vv}^{-1}\mathbf{G}\right)^{-1}\right]^{-1}\left(\mathbf{G}^T\mathbf{R}_{vv}^{-1}\mathbf{G}\right)^{-1},\end{aligned} \tag{4.27}$$

it is easy to see that

$$\left(\mathbf{G}^T\mathbf{R}_{yy}^{-1}\mathbf{G}\right)^{-1} = \mathbf{R}_{ss} + \left(\mathbf{G}^T\mathbf{R}_{vv}^{-1}\mathbf{G}\right)^{-1}. \tag{4.28}$$

As a result, we can check that

$$\mathbf{R}_{yy}^{-1}\mathbf{G}\left(\mathbf{G}^T\mathbf{R}_{yy}^{-1}\mathbf{G}\right)^{-1} = \mathbf{R}_{vv}^{-1}\mathbf{G}\left(\mathbf{G}^T\mathbf{R}_{vv}^{-1}\mathbf{G}\right)^{-1}. \tag{4.29}$$

Therefore, the LCMV filter can be also put in this form

$$\mathbf{h}_{\mathrm{R}} = \mathbf{R}_{vv}^{-1}\mathbf{G}\left(\mathbf{G}^T\mathbf{R}_{vv}^{-1}\mathbf{G}\right)^{-1}\mathbf{u}. \tag{4.30}$$

The LCMV filter with the reverberant model is very attractive from a theoretical point of view since it allows, in general, perfect dereverberation (desired signal stays intact) with a great amount of noise reduction as the value of L_h of the model filters is increased from its required minimum. However, in this context the LCMV filter may not be very practical since the acoustic impulse responses from the unknown source to the N microphones are difficult to estimate in real-world applications.

4.5 The LCMV Filter with the Spatio-Temporal Model

It seems that in order to avoid signal cancellation, we need to make sure that we dereverberate the signal of interest perfectly or up to a known filter. This requires the knowledge of a huge amount of information, i.e. the N acoustic

impulse responses from the signal of interest to the microphones, which is not very practical to acquire in most applications. It is then fair to ask if it's possible to perform noise reduction at one of the microphone signals, $x_n(k)$, without trying to recover the desired source $s(k)$ but with no further distortion on $x_n(k)$? The LCMV filter developed with the spatio-temporal model attempts to do that. In the rest, we will see how to recover the signal $x_1(k)$ the best possible way.

Now consider the array filter $\mathbf{h}$ of length NL_h and the total array output power $\mathbf{h}^T\mathbf{R}_{yy}\mathbf{h}$. We have:

$$\mathbf{h}^T\mathbf{R}_{yy}\mathbf{h} = \mathbf{h}^T\mathbf{R}_{xx}\mathbf{h} + \mathbf{h}^T\mathbf{R}_{vv}\mathbf{h}, \tag{4.31}$$

where $\mathbf{R}_{xx} = E\left[\mathbf{x}(k)\mathbf{x}^T(k)\right]$ is the correlation matrix of the signal

$$\mathbf{x}(k) = \left[\mathbf{x}_1^T(k)\ \mathbf{x}_2^T(k)\ \cdots\ \mathbf{x}_N^T(k)\right]^T. \tag{4.32}$$

Using (4.7) in (4.31), we find that

$$\mathbf{h}^T\mathbf{R}_{yy}\mathbf{h} = \mathbf{h}^T\mathbf{W}\mathbf{R}_{x_1x_1}\mathbf{W}^T\mathbf{h} + \mathbf{h}^T\mathbf{R}_{vv}\mathbf{h}, \tag{4.33}$$

where $\mathbf{R}_{x_1x_1} = E\left[\mathbf{x}_1(k)\mathbf{x}_1^T(k)\right]$ and

$$\mathbf{W} = \begin{bmatrix} \mathbf{I}_{L_h\times L_h} \\ \mathbf{W}_2 \\ \vdots \\ \mathbf{W}_N \end{bmatrix}$$

is a matrix of size $NL_h \times L_h$. Taking $\mathbf{W}^T\mathbf{h} = \mathbf{u}'$, (4.33) becomes

$$\mathbf{h}^T\mathbf{R}_{yy}\mathbf{h} = \sigma_{x_1}^2 + \mathbf{h}^T\mathbf{R}_{vv}\mathbf{h}, \tag{4.34}$$

where

$$\mathbf{u}' = \left[1\ 0\ \cdots\ 0\right]^T$$

is a vector of length L_h whose first component is equal to 1 while all others are zeroes and $\sigma_{x_1}^2$ is the variance of $x_1(k)$. Expression (4.34) shows clearly that it is possible to recover $x_1(k)$ undistorted while reducing the noise.

Therefore, from (4.34) we deduce the two optimization problems:

$$\min_{\mathbf{h}} \mathbf{h}^T\mathbf{R}_{yy}\mathbf{h} \quad \text{subject to} \quad \mathbf{W}^T\mathbf{h} = \mathbf{u}', \tag{4.35}$$

$$\min_{\mathbf{h}} \mathbf{h}^T\mathbf{R}_{vv}\mathbf{h} \quad \text{subject to} \quad \mathbf{W}^T\mathbf{h} = \mathbf{u}', \tag{4.36}$$

for which the optimal solutions are

$$\mathbf{h}_{\mathrm{ST},y} = \mathbf{R}_{yy}^{-1}\mathbf{W}\left(\mathbf{W}^T\mathbf{R}_{yy}^{-1}\mathbf{W}\right)^{-1}\mathbf{u}', \tag{4.37}$$

$$\mathbf{h}_{\mathrm{ST},v} = \mathbf{R}_{vv}^{-1}\mathbf{W}\left(\mathbf{W}^T\mathbf{R}_{vv}^{-1}\mathbf{W}\right)^{-1}\mathbf{u}', \tag{4.38}$$

where the subscript "ST" indicates a spatio-temporal signal model. These solutions are not only more realistic than the one given in Section 4.4 but they also require much less constraints, in principle, since $L_h \ll L$.

Now, we need to determine the filter matrix $\mathbf{W}$. An optimal estimator, in the Wiener sense, can be obtained by minimizing the following cost function:

$$J\left(\mathbf{W}_n\right) = E\left\{\left[\mathbf{x}_n(k) - \mathbf{W}_n\mathbf{x}_1(k)\right]^T\left[\mathbf{x}_n(k) - \mathbf{W}_n\mathbf{x}_1(k)\right]\right\}. \tag{4.39}$$

We easily find the optimal filter:

$$\mathbf{W}_{n,\mathrm{o}} = \mathbf{R}_{x_n x_1}\mathbf{R}_{x_1 x_1}^{-1}, \tag{4.40}$$

where $\mathbf{R}_{x_n x_1} = E\left[\mathbf{x}_n(k)\mathbf{x}_1^T(k)\right]$ is the cross-correlation matrix of the speech signals. However, the signals $x_n(k),\ n = 1, 2, \ldots, N$, are not observable so the Wiener filter matrix, as given in (4.40), can not be estimated in practice. But using $\mathbf{x}_n(k) = \mathbf{y}_n(k) - \mathbf{v}_n(k)$, we can verify that

$$\mathbf{R}_{x_n x_1} = \mathbf{R}_{y_n y_1} - \mathbf{R}_{v_n v_1},\ n = 1, 2, \ldots, N, \tag{4.41}$$

where $\mathbf{R}_{y_n y_1} = E\left[\mathbf{y}_n(k)\mathbf{y}_1^T(k)\right]$ and $\mathbf{R}_{v_n v_1} = E\left[\mathbf{v}_n(k)\mathbf{v}_1^T(k)\right]$. As a result

$$\mathbf{W}_{n,\mathrm{o}} = \left(\mathbf{R}_{y_n y_1} - \mathbf{R}_{v_n v_1}\right)\left(\mathbf{R}_{y_1 y_1} - \mathbf{R}_{v_1 v_1}\right)^{-1}. \tag{4.42}$$

The optimal filter matrix depends now only on the second-order statistics of the observation and noise signals. The statistics of the noise signals can be estimated during silences [when $s(k) = 0$] if we assume that the noise is stationary so that its statistics can be used for a next period when the speech is active. Note that if the source does not move, the optimal matrix needs to be estimated only once. Finally, the optimal LCMV filters based on the spatio-temporal model are given by

$$\mathbf{h}_{\mathrm{ST},y} = \mathbf{R}_{yy}^{-1}\mathbf{W}_{\mathrm{o}}\left(\mathbf{W}_{\mathrm{o}}^T\mathbf{R}_{yy}^{-1}\mathbf{W}_{\mathrm{o}}\right)^{-1}\mathbf{u}', \tag{4.43}$$

$$\mathbf{h}_{\mathrm{ST},v} = \mathbf{R}_{vv}^{-1}\mathbf{W}_{\mathrm{o}}\left(\mathbf{W}_{\mathrm{o}}^T\mathbf{R}_{vv}^{-1}\mathbf{W}_{\mathrm{o}}\right)^{-1}\mathbf{u}', \tag{4.44}$$

where

$$\mathbf{W}_{\mathrm{o}} = \begin{bmatrix} \mathbf{I}_{L_h\times L_h} \\ \mathbf{W}_{2,\mathrm{o}} \\ \vdots \\ \mathbf{W}_{N,\mathrm{o}} \end{bmatrix}.$$

In general, $\mathbf{h}_{\mathrm{ST},y} \neq \mathbf{h}_{\mathrm{ST},v}$ because (4.7) does not hold exactly and can only be approximated. It is reasonable to believe that these LCMV filters are the most useful ones in practice since they do not require that much *a priori* information to make them work in real-world applications. Moreover, even the geometry of the antenna does not need to be known and the calibration is not necessary. This is due to the fact that all this information is implicitly estimated in the matrix $\mathbf{W}_{\mathrm{o}}$.

Before finishing this part, let us show the link between the concept derived in this section and the so-called transfer function generalized sidelobe canceller (TF-GSC) [79], [80]. Using the signal model given in Section 4.4, we can easily see that

$$\mathbf{R}_{x_n x_1} = \mathbf{G}_n \mathbf{R}_{ss} \mathbf{G}_1^T, \tag{4.45}$$

$$\mathbf{R}_{x_1 x_1} = \mathbf{G}_1 \mathbf{R}_{ss} \mathbf{G}_1^T. \tag{4.46}$$

Substituting (4.45) and (4.46) into (4.40), we obtain

$$\mathbf{W}_{n,\mathrm{o}} = \mathbf{G}_n \mathbf{R}_{ss} \mathbf{G}_1^T \left[\mathbf{G}_1 \mathbf{R}_{ss} \mathbf{G}_1^T \right]^{-1}. \tag{4.47}$$

If the source signal $s(k)$ is white, then

$$\mathbf{R}_{ss} = \sigma_s^2 \cdot \mathbf{I}, \tag{4.48}$$

where σ_s^2 is the variance of the source signal. The optimal prediction matrix becomes

$$\mathbf{W}_{n,\mathrm{o}} = \mathbf{G}_n \mathbf{G}_1^T \left[\mathbf{G}_1 \mathbf{G}_1^T \right]^{-1}, \tag{4.49}$$

which depends solely on the channel information. In this particular case, the $\mathbf{W}_{n,\mathrm{o}}$ matrix can be viewed as the time-domain counterpart of the relative transfer function of the TF-GSC, so the LCMV filters given in (4.43)–(4.44) are equivalent to the TF-GSC approach [79]. However, in practical applications, speech signal is not white. Then, $\mathbf{W}_{n,\mathrm{o}}$ depends not only on the channel impulse responses, but also on the source correlation matrix. This indicates that the developed LCMV estimators exploit both the spatial and temporal prediction information for noise reduction. For more details on one of the LCMV filters ($\mathbf{h}_{\mathrm{ST},v}$) developed in this section, we invite the readers to consult [21], [44].

4.5.1 Experimental Results

In this subsection we evaluate the performance of the LCMV filter $\mathbf{h}_{\mathrm{ST},v}$ in real acoustic environments. We set up a multiple-microphone system in the varechoic chamber at Bell Labs [which is a room that measures 6.7 m long by 6.1 m wide by 2.9 m high ($x \times y \times z$)]. A total of ten microphones

are used and their locations are, respectively, at (2.437, 5.600, 1.400), (2.537, 5.600, 1.400), (2.637, 5.600, 1.400), (2.737, 5.600, 1.400), (2.837, 5.600, 1.400), (2.937, 5.600, 1.400), (3.037, 5.600, 1.400), (3.137, 5.600, 1.400), (3.237, 5.600, 1.400), and (3.337, 5.600, 1.400). To simulate a sound source, we place a loudspeaker at (1.337, 3.162, 1.600), playing back a speech signal prerecorded from a female speaker. To make the experiments repeatable, we first measured the acoustic channel impulse responses from the source to the ten microphones (each impulse response is first measured at 48 kHz and then downsampled to 8 kHz). These measured impulse responses are then treated as the true ones. During the experiments, the microphone outputs are generated by convolving the source signal with the corresponding measured impulse responses. Noise is then added to the convolved results to control the (input) SNR level.

The optimal speech estimate is

$$\hat{x}_1(k) = \sum_{n=1}^{N} \mathbf{h}_{n,\mathrm{ST},v}^T \mathbf{y}_n(k) = x_{1,\mathrm{nr}}(k) + v_{1,\mathrm{nr}}(k),$$

where $x_{1,\mathrm{nr}}(k) = \sum_{n=1}^{N} \mathbf{h}_{n,\mathrm{ST},v}^T \mathbf{x}_n(k)$ and $v_{1,\mathrm{nr}}(k) = \sum_{n=1}^{N} \mathbf{h}_{n,\mathrm{ST},v}^T \mathbf{v}_n(k)$ are, respectively, the speech filtered by the optimal filter and the residual noise. To assess the performance, we evaluate two measures, namely the output SNR and the Itakura-Saito (IS) distance [131]. The output SNR is defined as

$$\mathrm{SNR}_{\mathrm{o}} = \frac{E\left[x_{1,\mathrm{nr}}^2(k)\right]}{E\left[v_{1,\mathrm{nr}}^2(k)\right]}.$$

This measurement, when compared with the input SNR, tells us how much noise is reduced. The IS distance is a speech-distortion measure. For a detailed description of the IS distance, we refer to [131]. Many studies have shown that the IS measure is highly correlated with subjective quality judgements and two speech signals would be perceptually nearly identical if the IS distance between them is less than 0.1. In this experiment, we compute the IS distance between $x_1(k)$ and $x_{1,\mathrm{nr}}(k)$, which measures the degree of speech distortion due to the optimal filter.

In order to estimate and use the optimal filter given in (4.44), we need to specify the filter length L_h. If there is no reverberation, it is relatively easy to determine L_h, i.e., it needs only to be long enough to cover the maximal TDOA between the reference and the other microphones. In presence of reverberation, however, the determination of L_h would become more difficult and its value should, in theory, depend on the reverberation condition. Generally speaking, a longer filter has to be used if the environment is more reverberant. This experiment investigates the impact of the filter length on the algorithm performance. To eliminate the effect due to noise estimation, here we assume that the statistics of the noise signals are known *a priori*. The input SNR is 10 dB and the reverberation condition is controlled such that the reverberation time T_{60} is approximately 240 ms. The results are plotted in Fig. 4.1.

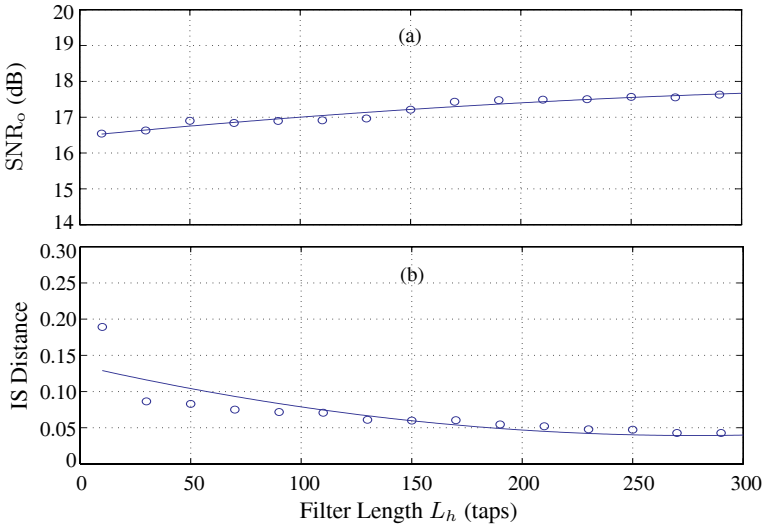


Fig. 4.1. The output SNR and the IS distance, both as a function of the filter length L_h: (a) SNR_o and (b) IS distance. The source is a speech signal from a female speaker; the background noise at each microphone is a computer-generated white Gaussian process; input SNR = 10 dB; and $T_{60} = 240$ ms. The fitting curve is a second-order polynomial.

One can see from Fig. 4.1(a) that the output SNR increases with L. So the longer is the filter, the more the noise is reduced. Compared with SNR_o, the IS distance decreases with L_h. This is understandable. As L_h increases, we will get a better prediction of $\mathbf{x}_n(k)$ from $\mathbf{x}_1(k)$. Consequently, the algorithm achieves more noise reduction and meanwhile causes less speech distortion. We also see from Fig. 4.1 that the output SNR increases almost linearly with L_h. Unlike the SNR curve, the relationship between the IS distance and the filter length L_h is not linear. Instead, the curve first decreases quickly as the filter length increases, and then continues to decrease but with a slower rate. After $L_h = 250$, continuing to increase L_h does not seem to further decrease the IS distance. So, from a speech-distortion point of view, $L_h = 250$ is long enough for a reasonable good performance.

The second experiment is to test the robustness of the multichannel algorithm to reverberation. The parameters used are: $L_h = 250$, $N = 10$, and input SNR = 10 dB. Compared with the previous experiments, this one does not assume to know the noise statistics. Instead, we developed a short-term energy based VAD (voice activity detector) to distinguish speech-plus-noise from noise-only segments. The noise covariance matrix is then computed from the noise-only segments using a batch method and the optimal filter is subsequently estimated according to (4.44). We tested the algorithm in two noise conditions: computer generated white Gaussian noise and a noise signal recorded in a New York Stock Exchange (NYSE) room. The results are de-

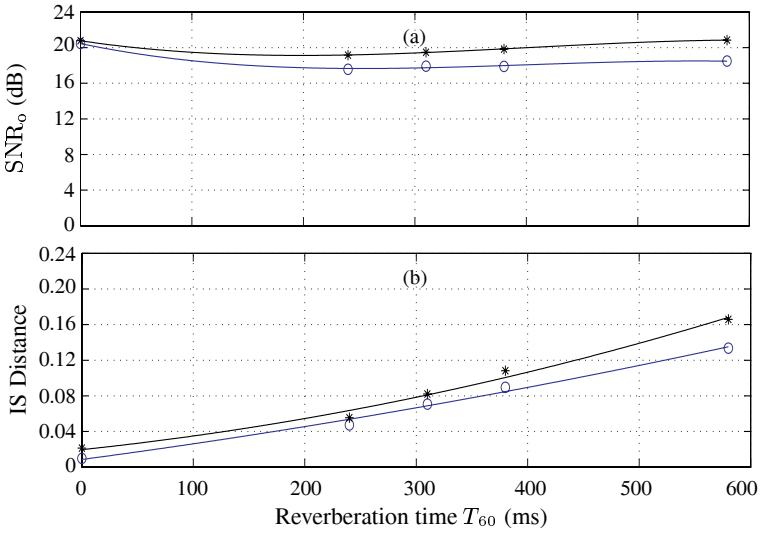


Fig. 4.2. Noise-reduction performance versus T_{60}. $*$: in white Gaussian noise; $\circ$: in NYSE noise; $L = 250$; input SNR = 10 dB. The fitting curve is a second-order polynomial.

picted in Fig. 4.2. We see that the output SNR in both situations does not vary much when the reverberation time changes. This indeed demonstrates that the developed LCMV filter is very immune to reverberation. In comparison with the output SNR, we see that the IS distance grows with the reverberation time. This result should not come as a surprise. As the reverberation time T_{60} increases, it becomes more difficult to predict the speech observed at one microphone from that received at another microphone. As a result, more speech distortion is unavoidable but it is still perceptually almost negligible.

4.6 The LCMV Filter in the Frequency Domain

For completeness, we derive in this section the LCMV filter in the frequency domain with the reverberant model.

Using the z-transform with $z = e^{j\omega}$, (4.3) can be rewritten as

$$Y_n(z) = S(z)G_n(z) + V_n(z),\ n = 1, 2, \ldots, N. \tag{4.50}$$

Consider the $N \times 1$ vector:

$$\begin{aligned} \mathbf{y}(z) &= \left[Y_1(z)\ Y_2(z) \cdots Y_N(z) \right]^T \\ &= S(z)\mathbf{g}(z) + \mathbf{v}(z), \end{aligned} \tag{4.51}$$

where

$$\mathbf{g}(z) = \left[G_1(z)\ G_2(z) \cdots G_N(z) \right]^T,$$
$$\mathbf{v}(z) = \left[V_1(z)\ V_2(z) \cdots V_N(z) \right]^T.$$

The power spectral density (PSD) matrix of the microphone signals is

$$\begin{aligned} \mathbf{\Phi}_{yy}(z) &= E\left[\mathbf{y}(z)\mathbf{y}^H(z)\right] \\ &= \phi_{ss}(z)\mathbf{g}(z)\mathbf{g}^H(z) + \mathbf{\Phi}_{vv}(z), \end{aligned} \tag{4.52}$$

where $\phi_{ss}(z) = E\left[|S(z)|^2\right]$ is the PSD of the source signal $s(k)$ and $\mathbf{\Phi}_{vv}(z) = E\left[\mathbf{v}(z)\mathbf{v}^H(z)\right]$ is the PSD matrix of the noise.

The constraint of the frequency-domain LCMV filter is based on the extended Euclid's algorithm: given the polynomials $G_1(z), G_2(z), \ldots, G_N(z)$, we can always find N other polynomials $H_1(z), H_2(z), \ldots, H_N(z)$ such that

$$\mathbf{h}^H(z)\mathbf{g}(z) = P(z), \tag{4.53}$$

where

$$\mathbf{h}(z) = \left[H_1(z)\ H_2(z) \cdots H_N(z) \right]^T,$$

$$P(z) = \gcd\left[G_1(z), G_2(z), \ldots, G_N(z)\right] = \gcd\left[\mathbf{g}(z)\right], \tag{4.54}$$

$\gcd[\cdot]$ denotes the greatest common divisor of the polynomials involved, and $\deg\left[H_n(z)\right] = L_h - 1 < L_g - L_p$, with $\deg\left[G_n(z)\right] = L_g - 1$ and $\deg\left[P(z)\right] = L_p - 1$.

Now that we have the constraint, we can formulate our optimization problem:

$$\min_{\mathbf{h}(z)} \mathbf{h}^H(z)\mathbf{\Phi}_{yy}(z)\mathbf{h}(z) \quad \text{subject to} \quad \mathbf{h}^H(z)\mathbf{g}(z) = P(z), \tag{4.55}$$

and the optimal solution is

$$\mathbf{h}_{\mathrm{F}}(z) = \frac{\mathbf{\Phi}_{yy}^{-1}(z)\mathbf{g}(z)P^*(z)}{\mathbf{g}^H(z)\mathbf{\Phi}_{yy}^{-1}(z)\mathbf{g}(z)}, \tag{4.56}$$

where the subscript "F" indicates that it's a frequency-domain filter and superscript * denotes complex conjugation. In the frequency domain, the LCMV filter simplifies to an MVDR filter [1], [2], [79].

Determining the inverse of $\mathbf{\Phi}_{yy}(z)$ from (4.52) with the Woodbury's identity

$$\begin{aligned} &\left[\mathbf{\Phi}_{vv}(z) + \phi_{ss}(z)\mathbf{g}(z)\mathbf{g}^H(z)\right]^{-1} = \\ &\quad \mathbf{\Phi}_{vv}^{-1}(z) - \frac{\mathbf{\Phi}_{vv}^{-1}(z)\mathbf{g}(z)\mathbf{g}^H(z)\mathbf{\Phi}_{vv}^{-1}(z)}{\phi_{ss}^{-1}(z) + \mathbf{g}^H(z)\mathbf{\Phi}_{vv}^{-1}(z)\mathbf{g}(z)} \end{aligned} \tag{4.57}$$

and substituting the result into (4.56), we obtain:

$$\mathbf{h}_{\mathrm{F}}(z) = \frac{\mathbf{\Phi}_{vv}^{-1}(z)\mathbf{g}(z)P^*(z)}{\mathbf{g}^H(z)\mathbf{\Phi}_{vv}^{-1}(z)\mathbf{g}(z)}. \tag{4.58}$$

With this form, we can deduce the residual noise:

$$\begin{aligned} |R(z)|^2 &= \mathbf{h}_{\mathrm{F}}^H(z)\mathbf{\Phi}_{vv}^{-1}(z)\mathbf{h}_{\mathrm{F}}(z) \\ &= \frac{|P(z)|^2}{\mathbf{g}^H(z)\mathbf{\Phi}_{vv}^{-1}(z)\mathbf{g}(z)}. \end{aligned} \tag{4.59}$$

We can observe that the residual noise depends on two elements: the magnitude square of the polynomial $P(z)$ and the coherence of the noise. The larger the number of common zeroes among the acoustic impulse responses, the higher the residual noise. Also, the higher the coherence of the noise at the microphones, the higher the residual error.

This simple analysis, in the frequency domain, shows the limits of the LCMV filter with the reverberant model on dereverberation and noise reduction. The performance of this optimal filter depends quite a lot on the reverberation of the room (i.e., the acoustic impulse responses) and the characteristics of the noise. Because of this high dependency, it is reasonable to assert that this filter may not be that reliable in practice.

4.7 Conclusions

In this chapter, the classical LCMV filter was studied in room acoustic environments. For a deep insight into this filter, we have proposed three mathematical models: anechoic, reverberant, and spatio-temporal. The anechoic model is not very realistic so the LCMV filter derived in this context may not perform very well if used in a real room. The more realistic reverberant model requires an unrealistic huge amount of information (i.e., acoustic impulse responses). For this reason, the LCMV filter with this model is not really implementable even in subbands. Finally, the two LCMV filters derived with the spatio-temporal model seem promising since they allow reduction of the background noise with little distortion of the reference signal but dereverberation is not possible.

Contrary to what it's claimed here and there, dereverberation does not seem feasible with the LCMV filter in general. As for noise reduction, the LCMV filter is of interest only if it does not distort the reference speech signal. If we do not want to distort the source signal, we need to dereverberate it exactly otherwise some signal cancellation will happen. However, we can do some noise reduction at anyone of the microphone signals without distorting the speech component at that microphone (in this case, there is no dereverberation), which will be studied more thoroughly in the next chapter.

5

Noise Reduction with Multiple Microphones: a Unified Treatment

5.1 Introduction

Wherever we are, noise (originating from various ambient sound sources) is permanently present. As a result, speech signals can not be acquired and processed, in general, in pure form. It is known for a long time that noise can profoundly affect human-to-human and human-to-machine communications, including changing a talker's speaking pattern, modifying the characteristics of the speech signal, degrading speech quality and intelligibility, and affecting the listener's perception and machine's processing of the recorded speech. In order to make voice communication feasible, natural, and comfortable in the presence of noise regardless of the noise level, it is desirable to develop digital signal processing techniques to "clean" the microphone signal before it is stored, transmitted, or played out. This problem has been a major challenge for many researchers and engineers for more than four decades [16].

In the single-channel scenario, the signal picked up by the microphone can be modeled as a superposition of the clean speech and noise. The objective of noise reduction, then, becomes to restore the original clean speech from the mixed signal. The first single-channel noise reduction algorithm was developed more than 40 years ago by Schroeder [199], [200]. He proposed an analog implementation of the spectral magnitude subtraction. This work, however, has not received much public attention, probably because it was never published in journals or conferences. About 15 years later, Boll, in his informative paper [24], reinvented the spectral subtraction method but in the digital domain. Almost at the same time, Lim and Oppenheim, in their landmark work [153], systematically formulated the noise-reduction problem and studied and compared the different algorithms known at that time. Since then many algorithms have been derived in the time and frequency domains [16], [43], [156], [218]. The main drawback of single-channel speech enhancement algorithms is that they distort the desired speech signal. So researchers have proposed to use multiple microphones or microphone arrays in order to better deal with this fundamental problem.

The objective of this chapter is to study the most important noise reduction algorithms in the multichannel case. The main desire is to see if, indeed, the use of multiple microphones can help in minimizing speech distortion while having a good amount of noise reduction at the same time. This chapter is organized as follows. Section 5.2 describes the problem and the signal model while Section 5.3 gives some very useful definitions that will help the reader understand how noise reduction algorithms work. Section 5.4 explains the multichannel Wiener filter. Section 5.5 develops the subspace method with multiple microphones. In Section 5.6, the spatio-temporal prediction approach is derived. Section 5.7 deals with the difficult problem of coherent noise. In Section 5.8, it is shown how the adaptive noise cancellation idea can be used in this context. Section 5.9 generalizes the Kalman filter to the multichannel case. In Section 5.10, we present some simulations. Finally, we give our conclusions in Section 5.11.

5.2 Signal Model and Problem Description

In this section, we explain the problem that we wish to tackle. We consider the general situation where we have N microphone signals whose outputs, at the discrete time k, are

$$\begin{aligned} y_n(k) &= g_n * s(k) + v_n(k) \\ &= x_n(k) + v_n(k),\ n = 1, 2, \ldots, N, \end{aligned} \tag{5.1}$$

where g_n is the impulse response from the unknown source to the nth microphone and $v_n(k)$ is the noise at microphone n. We assume that the signals $v_n(k)$ and $x_n(k)$ are uncorrelated and zero-mean. Without loss of generality, we consider the first microphone signal $y_1(k)$ as the reference. Our main objective in this chapter is noise reduction [16], [218]; hence we will try to recover $x_1(k)$ the best way we can in some sense by observing not only one microphone signal but N of them. We do not attempt here to recover $s(k)$ (i.e., speech dereverberation) except in Section 5.9 with the Kalman filter. This problem, although very important, is difficult and requires other techniques to solve it [18], [123], [125]. (See also Chapters 4, 7, and 8.) Contrary to most beamforming techniques, the geometry of the microphone array has little or no impact on the algorithms presented here, so the calibration step is not necessary.

The signal model given in (5.1) can be written in a vector/matrix form if we process the data by blocks of L samples:

$$\mathbf{y}_n(k) = \mathbf{x}_n(k) + \mathbf{v}_n(k),\ n = 1, 2, \ldots, N, \tag{5.2}$$

where

$$\mathbf{y}_n(k) = \left[y_n(k)\ y_n(k-1) \cdots y_n(k-L+1) \right]^T$$

is a vector containing the L most recent samples of the noisy speech signal $y_n(k)$, and $\mathbf{x}_n(k)$ and $\mathbf{v}_n(k)$ are defined in a similar way to $\mathbf{y}_n(k)$. Again, our objective is to estimate $\mathbf{x}_1(k)$ from the observations $\mathbf{y}_n(k)$, $n = 1, 2, \ldots, N$.

Usually, we estimate the noise-free speech, $\mathbf{x}_1(k)$, by applying a linear transformation to the microphone signals, i.e.,

$$\begin{aligned}\mathbf{z}(k) &= \sum_{n=1}^{N} \mathbf{H}_n \mathbf{y}_n(k) \\ &= \mathbf{H}\mathbf{y}(k) \\ &= \mathbf{H}\left[\mathbf{x}(k) + \mathbf{v}(k)\right], \end{aligned} \tag{5.3}$$

where

$$\begin{aligned}\mathbf{y}(k) &= \left[\mathbf{y}_1^T(k)\ \mathbf{y}_2^T(k) \cdots \mathbf{y}_N^T(k)\right]^T, \\ \mathbf{x}(k) &= \left[\mathbf{x}_1^T(k)\ \mathbf{x}_2^T(k) \cdots \mathbf{x}_N^T(k)\right]^T, \\ \mathbf{v}(k) &= \left[\mathbf{v}_1^T(k)\ \mathbf{v}_2^T(k) \cdots \mathbf{v}_N^T(k)\right]^T, \\ \mathbf{H} &= \left[\mathbf{H}_1\ \mathbf{H}_2 \cdots \mathbf{H}_N\right], \end{aligned}$$

and $\mathbf{H}_n$, $n = 1, 2, \ldots, N$, are the filtering matrices of size $L \times L$, so $\mathbf{H}$ is the global filtering matrix of size $L \times NL$. From this estimate, we define the error signal vector as

$$\begin{aligned}\mathbf{e}(k) &= \mathbf{z}(k) - \mathbf{x}_1(k) \\ &= (\mathbf{H} - \mathbf{U})\,\mathbf{x}(k) + \mathbf{H}\mathbf{v}(k) \\ &= \mathbf{e}_x(k) + \mathbf{e}_v(k), \end{aligned} \tag{5.4}$$

where

$$\mathbf{U} = \left[\mathbf{I}_{L\times L}\ \mathbf{0}_{L\times L} \cdots \mathbf{0}_{L\times L}\right]$$

is an $L \times NL$ matrix with $\mathbf{I}_{L\times L}$ being the identity matrix of size $L \times L$,

$$\mathbf{e}_x(k) = (\mathbf{H} - \mathbf{U})\,\mathbf{x}(k) \tag{5.5}$$

is the speech distortion due to the linear transformation, and

$$\mathbf{e}_v(k) = \mathbf{H}\mathbf{v}(k) \tag{5.6}$$

represents the residual noise.

5.3 Some Useful Definitions

In Chapter 2 we have defined many objective measures for evaluating the performance of single-channel noise-reduction algorithms. In this section, we

extend those measures to the multichannel situation, which will be useful in the rest of this chapter.

The best way to quantify the amount of noise from an observed signal is the SNR. Since our reference microphone is the first one, we define the input SNR as

$$\begin{aligned}\mathrm{SNR} &= \frac{\sigma_{x_1}^2}{\sigma_{v_1}^2} = \frac{E\left[\mathbf{x}_1^T(k)\mathbf{x}_1(k)\right]}{E\left[\mathbf{v}_1^T(k)\mathbf{v}_1(k)\right]} \\ &= \frac{\mathrm{tr}\left\{E\left[\mathbf{U}\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{U}^T\right]\right\}}{\mathrm{tr}\left\{E\left[\mathbf{U}\mathbf{v}(k)\mathbf{v}^T(k)\mathbf{U}^T\right]\right\}},\end{aligned} \tag{5.7}$$

where tr[·] denotes the trace of a matrix.

The primary issue that we must determine with noise reduction is how much noise is actually attenuated. The noise-reduction factor is a measure of this and its mathematical definition, in the multichannel case, is

$$\begin{aligned}\xi_{\mathrm{nr}}(\mathbf{H}) &= \frac{E\left[\mathbf{v}_1^T(k)\mathbf{v}_1(k)\right]}{E\left[\mathbf{e}_v^T(k)\mathbf{e}_v(k)\right]} \\ &= \frac{\mathrm{tr}\left\{E\left[\mathbf{U}\mathbf{v}(k)\mathbf{v}^T(k)\mathbf{U}^T\right]\right\}}{\mathrm{tr}\left\{E\left[\mathbf{H}\mathbf{v}(k)\mathbf{v}^T(k)\mathbf{H}^T\right]\right\}}.\end{aligned} \tag{5.8}$$

This factor should be lower bounded by 1. The larger the value of $\xi_{\mathrm{nr}}(\mathbf{H})$, the more the noise is reduced.

Most, if not all, of the known methods achieve noise reduction at the price of distorting the speech signal. Therefore, it is extremely useful to quantify this distortion. The multichannel speech-distortion index is defined as follows:

$$\upsilon_{\mathrm{sd}}(\mathbf{H}) = \frac{E\left[\mathbf{e}_x^T(k)\mathbf{e}_x(k)\right]}{E\left[\mathbf{x}_1^T(k)\mathbf{x}_1(k)\right]}. \tag{5.9}$$

This parameter is lower bounded by 0 and expected to be upper bounded by 1. The higher the value of $\upsilon_{\mathrm{sd}}(\mathbf{H})$, the more the speech signal $x_1(k)$ is distorted.

Noise reduction is done at the expense of speech reduction. Similar to the noise-reduction factor, we give the definition of the speech-reduction factor:

$$\xi_{\mathrm{sr}}(\mathbf{H}) = \frac{\mathrm{tr}\left\{E\left[\mathbf{U}\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{U}^T\right]\right\}}{\mathrm{tr}\left\{E\left[\mathbf{H}\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{H}^T\right]\right\}}. \tag{5.10}$$

This factor is also lower bounded by 1.

In order to know if the filtering matrix ($\mathbf{H}$) improves the SNR, we evaluate the output SNR after noise reduction as

$$\mathrm{SNR}(\mathbf{H}) = \frac{\mathrm{tr}\left\{E\left[\mathbf{H}\mathbf{x}(k)\mathbf{x}^T(k)\mathbf{H}^T\right]\right\}}{\mathrm{tr}\left\{E\left[\mathbf{H}\mathbf{v}(k)\mathbf{v}^T(k)\mathbf{H}^T\right]\right\}}. \tag{5.11}$$

It is nice to find a filter $\mathbf{H}$ in such a way that $\mathrm{SNR}(\mathbf{H}) > \mathrm{SNR}$ since the SNR is the most reliable objective measure we have in our hands for the evaluation of speech enhancement algorithms and it's also reasonable to assume, to some extent, some correlation between SNR and subjective listening. However, maximizing $\mathrm{SNR}(\mathbf{H})$ is certainly not the best thing to do since the distortion of the speech signal will likely be maximized as well.

Using expressions (5.7), (5.8), (5.10), and (5.11), it is easy to see that we always have:

$$\frac{\mathrm{SNR}(\mathbf{H})}{\mathrm{SNR}} = \frac{\xi_{\mathrm{nr}}(\mathbf{H})}{\xi_{\mathrm{sr}}(\mathbf{H})}. \tag{5.12}$$

Hence, $\mathrm{SNR}(\mathbf{H}) > \mathrm{SNR}$ if and only if $\xi_{\mathrm{nr}}(\mathbf{H}) > \xi_{\mathrm{sr}}(\mathbf{H})$. So is it possible that with a judicious choice of the filtering matrix $\mathbf{H}$ we can have $\xi_{\mathrm{nr}}(\mathbf{H}) > \xi_{\mathrm{sr}}(\mathbf{H})$? The answer is yes. A generally rough and intuitive justification to this answer is quite simple: improvement of the output SNR is due to the fact that speech signals are partly predictable. In this situation, $\mathbf{H}$ is a kind of a complex predictor or interpolator matrix and as a result, $\xi_{\mathrm{sr}}(\mathbf{H})$ can be close to 1 while $\xi_{\mathrm{nr}}(\mathbf{H})$ can be much larger than 1. This fact is very important for the single-microphone case and has the potential to be also important in the multichannel case where we can exploit not only the temporal prediction of the speech signal but also the spatial prediction of the observed signals from different microphones in order to improve the output SNR and minimize the speech distortion.

5.4 Wiener Filter

In this section, we derive the classical optimal Wiener filter for noise reduction. Let us first write the mean-square error (MSE) criterion

$$\begin{aligned} J(\mathbf{H}) &= \mathrm{tr}\left\{E\left[\mathbf{e}(k)\mathbf{e}^T(k)\right]\right\} \\ &= E\left[\mathbf{x}_1^T(k)\mathbf{x}_1(k)\right] + \mathrm{tr}\left[\mathbf{H}\mathbf{R}_{yy}\mathbf{H}^T\right] - 2\mathrm{tr}\left[\mathbf{H}\mathbf{R}_{yx_1}\right], \end{aligned} \tag{5.13}$$

where $\mathbf{R}_{yy} = E\left[\mathbf{y}(k)\mathbf{y}^T(k)\right]$ is the $NL \times NL$ correlation matrix of the observation signals and $\mathbf{R}_{yx_1} = E\left[\mathbf{y}(k)\mathbf{x}_1^T(k)\right]$ is the $NL \times L$ cross-correlation matrix between the observation and speech signals. Differentiating the MSE criterion with respect to $\mathbf{H}$ and setting the result to zero, we find the Wiener filter matrix [59], [60]

$$\mathbf{H}_{\mathrm{W}}^T = \mathbf{R}_{yy}^{-1}\mathbf{R}_{yx_1}. \tag{5.14}$$

The previous equation is of little help in practice since the vector $\mathbf{x}_1(k)$ is unobservable. However, it is easy to check that

$$\mathbf{R}_{yx_1} = (\mathbf{R}_{yy} - \mathbf{R}_{vv})\,\mathbf{U}^T, \tag{5.15}$$

with $\mathbf{R}_{vv} = E\left[\mathbf{v}(k)\mathbf{v}^T(k)\right]$ being the $NL \times NL$ correlation matrix of the noise signals. Now $\mathbf{R}_{yx_1}$ depends on the correlation matrices $\mathbf{R}_{yy}$ and $\mathbf{R}_{vv}$: the first one can be easily estimated during speech-and-noise periods while the second one can be estimated during noise-only intervals assuming that the statistics of the noise do not change much with time. Substituting (5.15) into (5.14), we get

$$\mathbf{H}_{\mathrm{W}}^T = \left(\mathbf{I}_{NL\times NL} - \mathbf{R}_{yy}^{-1}\mathbf{R}_{vv}\right)\mathbf{U}^T. \tag{5.16}$$

The minimum MSE (MMSE) is obtained by replacing $\mathbf{H}_{\mathrm{W}}$ in (5.13), i.e. $J(\mathbf{H}_{\mathrm{W}})$. There are different ways to express this MMSE. One useful expression is

$$J(\mathbf{H}_{\mathrm{W}}) = \mathrm{tr}\left(\mathbf{U}\mathbf{R}_{vv}\mathbf{U}^T\right) - \mathrm{tr}\left(\mathbf{U}\mathbf{R}_{vv}\mathbf{R}_{yy}^{-1}\mathbf{R}_{vv}\mathbf{U}^T\right). \tag{5.17}$$

Now we can define the normalized MMSE (NMMSE)

$$\tilde{J}(\mathbf{H}_{\mathrm{W}}) = \frac{J(\mathbf{H}_{\mathrm{W}})}{J(\mathbf{U})} = \frac{J(\mathbf{H}_{\mathrm{W}})}{E\left[\mathbf{v}_1^T(k)\mathbf{v}_1(k)\right]}, \tag{5.18}$$

where $0 \leq \tilde{J}(\mathbf{H}_{\mathrm{W}}) \leq 1$. This definition is related to the speech-distortion index and the noise-reduction factor by the formula

$$\tilde{J}(\mathbf{H}_{\mathrm{W}}) = \mathrm{SNR} \cdot \upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{W}}) + \frac{1}{\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{W}})}. \tag{5.19}$$

As a matter of fact, (5.19) is valid for any filter $\mathbf{H}$, i.e.,

$$\tilde{J}(\mathbf{H}) = \mathrm{SNR} \cdot \upsilon_{\mathrm{sd}}(\mathbf{H}) + \frac{1}{\xi_{\mathrm{nr}}(\mathbf{H})}. \tag{5.20}$$

We deduce the two inequalities

$$\upsilon_{\mathrm{sd}}(\mathbf{H}) \leq \frac{1}{\mathrm{SNR}}\left[1 - \frac{1}{\xi_{\mathrm{nr}}(\mathbf{H})}\right], \tag{5.21}$$

$$\xi_{\mathrm{nr}}(\mathbf{H}) \geq \frac{1}{1 - \mathrm{SNR} \cdot \upsilon_{\mathrm{sd}}(\mathbf{H})}. \tag{5.22}$$

It can be shown that $\mathrm{SNR}(\mathbf{H}_{\mathrm{W}}) \geq \mathrm{SNR}$ for any filter matrix dimension and for all possible speech and noise correlation matrices [16], [41], [62]. This may come at a heavy price: large speech distortion. Using this property and expression (5.12), we deduce that

$$\mathrm{SNR} \leq \mathrm{SNR}(\mathbf{H}_{\mathrm{W}}) \leq \mathrm{SNR} \cdot \xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{W}}). \tag{5.23}$$

From (5.19) and (5.23) we can get this upper bound for $\mathrm{SNR}(\mathbf{H}_{\mathrm{W}})$:

$$\mathrm{SNR}(\mathbf{H}_{\mathrm{W}}) \leq \frac{1}{\frac{\tilde{J}(\mathbf{H}_{\mathrm{W}})}{\mathrm{SNR}} - \upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{W}})}, \tag{5.24}$$

which shows that the output SNR is improved at the expense of speech distortion. It is seen that the Wiener formulation does not explicitly exploit the spatial information.

Particular case: single microphone and white noise.

We assume here that only one microphone signal is available (i.e., $N = 1$) and the noise picked up by this microphone is white (i.e., $\mathbf{R}_{v_1 v_1} = \sigma_{v_1}^2 \mathbf{I}_{L\times L}$). In this situation, the Wiener filter matrix becomes

$$\mathbf{H}_{\mathrm{W}} = \mathbf{I}_{L\times L} - \sigma_{v_1}^2 \mathbf{R}_{y_1 y_1}^{-1}, \tag{5.25}$$

where

$$\mathbf{R}_{y_1 y_1} = \mathbf{R}_{x_1 x_1} + \sigma_{v_1}^2 \mathbf{I}_{L\times L}.$$

It is well known that the inverse of the Toeplitz matrix $\mathbf{R}_{y_1 y_1}$ can be factorized as follows [12], [140] (see also Chapter 2):

$$\mathbf{R}_{y_1 y_1}^{-1} = \begin{bmatrix} 1 & -c_{1,0} & \cdots & -c_{L-1,0} \\ -c_{0,1} & 1 & \cdots & -c_{L-1,1} \\ \vdots & \vdots & \ddots & \vdots \\ -c_{0,L-1} & -c_{1,L-1} & \cdots & 1 \end{bmatrix} \times \begin{bmatrix} 1/E_0 & 0 & \cdots & 0 \\ 0 & 1/E_1 & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & 0 & \cdots & 1/E_{L-1} \end{bmatrix}, \tag{5.26}$$

where the columns of the first matrix in the right-hand side of (5.26) are the linear interpolators of the signal $y_1(k)$ and the elements E_l in the diagonal matrix are the respective interpolation-error powers.

Using the factorization of $\mathbf{R}_{y_1 y_1}^{-1}$ in (5.17), the MMSE and NMMSE can be rewritten, respectively, as

$$J(\mathbf{H}_{\mathrm{W}}) = L\sigma_{v_1}^2 - \left(\sigma_{v_1}^2\right)^2 \sum_{l=0}^{L-1} \frac{1}{E_l}, \tag{5.27}$$

$$\tilde{J}(\mathbf{H}_{\mathrm{W}}) = 1 - \frac{\sigma_{v_1}^2}{L} \sum_{l=0}^{L-1} \frac{1}{E_l}. \tag{5.28}$$

Assume that the noise-free speech signal, $x_1(k)$, is very well predictable. In this scenario, $E_l \approx \sigma_{v_1}^2, \ \forall \ l$, and replacing this value in (5.28) we find that $\tilde{J}(\mathbf{H}_{\mathrm{W}}) \approx 0$. From (5.19), we then deduce that $\upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{W}}) \approx 0$ (almost no speech distortion) and $\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{W}}) \approx \infty$ (almost infinite noise reduction). Notice that this result seems independent of the SNR. Also, since $\mathbf{H}_{\mathrm{W}}\mathbf{x}(k) \approx \mathbf{x}_1(k)$, this means that $\xi_{\mathrm{sr}}(\mathbf{H}_{\mathrm{W}}) \approx 1$; as a result $\mathrm{SNR}(\mathbf{H}_{\mathrm{W}}) \approx \infty$ and we can almost perfectly recover the signal $x_1(k)$.

At the other extreme case, let us see now what happens when the source signal $x_1(k)$ is not predictable at all. In this situation, $E_l \approx \sigma_{y_1}^2, \ \forall \ l$ and $c_{ij} \approx 0, \ \forall \ i, j$. Using these values, we get

$$\mathbf{H}_{\mathrm{W}} \approx \frac{\mathrm{SNR}}{1+\mathrm{SNR}}\mathbf{I}_{L\times L}, \tag{5.29}$$

$$\tilde{J}(\mathbf{H}_{\mathrm{W}}) \approx \frac{\mathrm{SNR}}{1+\mathrm{SNR}}. \tag{5.30}$$

With the help of the two previous equations, it's straightforward to obtain

$$\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{W}}) \approx \left(1+\frac{1}{\mathrm{SNR}}\right)^2, \tag{5.31}$$

$$\upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{W}}) \approx \frac{1}{(1+\mathrm{SNR})^2}, \tag{5.32}$$

$$\mathrm{SNR}(\mathbf{H}_{\mathrm{W}}) \approx \mathrm{SNR}. \tag{5.33}$$

While some noise reduction is achieved (at the price of speech distortion), there is no improvement in the output SNR, meaning that the Wiener filter has no positive effect on the microphone signal $y_1(k)$.

This analysis, even though simple, is quite insightful. It shows that the Wiener filter may not be that bad after all, as long as the source signal is somewhat predictable. However, in practice some discontinuities could be heard from a voiced signal to an unvoiced one, since for the former the noise will be mostly removed while it will not for the latter.

A possible consequence of this analysis is the effect of reverberation. Indeed, even if the source signal $s(k)$ is white, thanks to the effect of the impulse response g_1, the signal $x_1(k)$ is not white and may become more "predictable." Hence, by making the source signal, $s(k)$, more predictable, reverberation may help the Wiener filter for better noise reduction. We can draw the same kind of conclusion for any number of microphones.

5.5 Subspace Method

In the Wiener filter, we can not control the compromise between noise reduction and speech distortion. So this filter derived from the classical MSE criterion may be limited in practice because of its lack of flexibility. Ephraim

and Van Trees proposed, in the single-channel case, a more meaningful criterion which consists of minimizing the speech distortion while keeping the residual noise power below some given threshold [69]. The deduced optimal estimator is shown to be a Wiener filter with adjustable input noise level. This filter was developed in the white noise case. Since then, many algorithms have been proposed to deal with the general colored noise [107], [111], [151], [165], [189]. However, the most elegant algorithm is the one using the generalized eigenvalue decomposition [111], [112], [132].

Using the same signal model described in Section 5.2, the optimal filter with the subspace technique can be mathematically derived from the optimization problem

$$\mathbf{H}_{\mathrm{S}} = \arg\min_{\mathbf{H}} J_x(\mathbf{H}) \quad \text{subject to} \quad J_v(\mathbf{H}) \leq L\sigma^2, \tag{5.34}$$

where

$$J_x(\mathbf{H}) = \mathrm{tr}\left\{E\left[\mathbf{e}_x(k)\mathbf{e}_x^T(k)\right]\right\}, \tag{5.35}$$

$$J_v(\mathbf{H}) = \mathrm{tr}\left\{E\left[\mathbf{e}_v(k)\mathbf{e}_v^T(k)\right]\right\}, \tag{5.36}$$

and $\sigma^2 < \sigma_{v_1}^2$ in order to have some noise reduction. If we use a Lagrange multiplier, μ, to adjoin the constraint to the cost function, (5.34) can be rewritten as

$$\mathbf{H}_{\mathrm{S}} = \arg\min_{\mathbf{H}} \mathcal{L}(\mathbf{H}, \mu), \tag{5.37}$$

with

$$\mathcal{L}(\mathbf{H}, \mu) = J_x(\mathbf{H}) + \mu\left[J_v(\mathbf{H}) - L\sigma^2\right] \tag{5.38}$$

and $\mu \geq 0$. We can easily prove from (5.37) that the optimal filter is

$$\begin{aligned}\mathbf{H}_{\mathrm{S}}^T &= \left(\mathbf{R}_{xx} + \mu\mathbf{R}_{vv}\right)^{-1}\mathbf{R}_{xx}\mathbf{U}^T \\ &= \left[\mathbf{R}_{yy} + (\mu-1)\mathbf{R}_{vv}\right]^{-1}\left[\mathbf{R}_{yy} - \mathbf{R}_{vv}\right]\mathbf{U}^T \\ &= \left[\mathbf{I}_{NL\times NL} + (\mu-1)\mathbf{R}_{yy}^{-1}\mathbf{R}_{vv}\right]^{-1}\mathbf{H}_{\mathrm{W}}^T,\end{aligned} \tag{5.39}$$

where $\mathbf{R}_{xx} = E\left[\mathbf{x}(k)\mathbf{x}^T(k)\right]$ is the $NL \times NL$ correlation matrix of the speech signal at the different microphones and the Lagrange multiplier satisfies $J_v(\mathbf{H}_{\mathrm{S}}) = L\sigma^2$, which implies that

$$\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{S}}) = \frac{\sigma_{v_1}^2}{\sigma^2} > 1. \tag{5.40}$$

From (5.21), we get

$$\upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{S}}) \leq \frac{\sigma_{v_1}^2 - \sigma^2}{\sigma_{x_1}^2}. \tag{5.41}$$

Since $\tilde{J}(\mathbf{H}_{\mathrm{W}}) \leq \tilde{J}(\mathbf{H}_{\mathrm{S}}),\ \forall\mu$, we also have

$$\upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{S}}) \geq \upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{W}}) + \frac{1}{\mathrm{SNR}}\left[\frac{1}{\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{W}})} - \frac{1}{\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{S}})}\right]. \tag{5.42}$$

Therefore, $\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{S}}) \geq \xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{W}})$ implies that $\upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{S}}) \geq \upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{W}})$. However, $\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{S}}) \leq \xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{W}})$ does not imply that $\upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{S}}) \leq \upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{W}})$.

In practice it's not easy to determine an optimal value of μ. Therefore, when this parameter is chosen in an ad-hoc way, we can see that for

- $\mu = 1$, $\mathbf{H}_{\mathrm{S}} = \mathbf{H}_{\mathrm{W}}$;
- $\mu = 0$, $\mathbf{H}_{\mathrm{S}} = \mathbf{U}$;
- $\mu > 1$, results in low residual noise at the expense of high speech distortion;
- $\mu < 1$, we get little speech distortion but not so much noise reduction.

In the single-channel case, it can be shown that $\mathrm{SNR}(\mathbf{H}_{\mathrm{S}}) \geq \mathrm{SNR}$ [42]. The same kind of proof holds for any number of microphones.

As shown in [77], the two symmetric matrices $\mathbf{R}_{xx}$ and $\mathbf{R}_{vv}$ can be jointly diagonalized if $\mathbf{R}_{vv}$ is positive definite. This joint diagonalization was first used by Jensen et al. [132] and then by Hu and Loizou [111], [112], [113] in the single-channel case. In our multichannel context we have

$$\mathbf{R}_{xx} = \mathbf{B}^T\mathbf{\Lambda}\mathbf{B}, \tag{5.43}$$
$$\mathbf{R}_{vv} = \mathbf{B}^T\mathbf{B}, \tag{5.44}$$
$$\mathbf{R}_{yy} = \mathbf{B}^T\left[\mathbf{I}_{NL\times NL} + \mathbf{\Lambda}\right]\mathbf{B}, \tag{5.45}$$

where $\mathbf{B}$ is a full rank square matrix but not necessarily orthogonal, and the diagonal matrix

$$\mathbf{\Lambda} = \mathrm{diag}\left[\lambda_1\ \lambda_2\ \cdots\ \lambda_{NL}\right] \tag{5.46}$$

are the eigenvalues of the matrix $\mathbf{R}_{vv}^{-1}\mathbf{R}_{xx}$ with $\lambda_1 \geq \lambda_2 \geq \cdots \geq \lambda_{NL} \geq 0$.

Applying the decompositions (5.43)–(5.45) in (5.39), the optimal estimator becomes

$$\mathbf{H}_{\mathrm{S}} = \mathbf{U}\mathbf{B}^T\mathbf{\Lambda}\left(\mathbf{\Lambda} + \mu\mathbf{I}_{NL\times NL}\right)^{-1}\mathbf{B}^{-T}. \tag{5.47}$$

Therefore, the estimation of the speech signal, $\mathbf{x}_1(k)$, is done in three steps: first we apply the transform $\mathbf{B}^{-T}$ to the noisy signal; second the transformed signal is modified by the gain function $\mathbf{\Lambda}\left(\mathbf{\Lambda} + \mu\mathbf{I}_{NL\times NL}\right)^{-1}$; and finally we transform back the signal to its original domain by applying the transform $\mathbf{U}\mathbf{B}^T$.

Usually, a speech signal can be modelled as a linear combination of a number of some (linearly independent) basis vectors smaller than the dimension of these vectors. As a result, the vector space of the noisy signal can be decomposed in two subspaces: the signal-plus-noise subspace of length L_{s} and the noise subspace of length L_{n}, with $NL = L_{\mathrm{s}} + L_{\mathrm{n}}$. This implies that the

last L_n eigenvalues of the matrix $\mathbf{R}_{vv}^{-1}\mathbf{R}_{xx}$ are equal to zero. Therefore, we can rewrite (5.47) as

$$\mathbf{H}_\mathrm{S} = \mathbf{U}\mathbf{B}^T \begin{bmatrix} \mathbf{\Sigma} & \mathbf{0}_{L_\mathrm{s}\times L_\mathrm{n}} \\ \mathbf{0}_{L_\mathrm{n}\times L_\mathrm{s}} & \mathbf{0}_{L_\mathrm{n}\times L_\mathrm{n}} \end{bmatrix} \mathbf{B}^{-T}, \tag{5.48}$$

where

$$\mathbf{\Sigma} = \mathrm{diag}\left[\frac{\lambda_1}{\lambda_1+\mu}, \frac{\lambda_2}{\lambda_2+\mu}, \cdots, \frac{\lambda_{L_\mathrm{s}}}{\lambda_{L_\mathrm{s}}+\mu}\right] \tag{5.49}$$

is an $L_\mathrm{s} \times L_\mathrm{s}$ diagonal matrix. We now clearly see that noise reduction with the subspace method is achieved by nulling the noise subspace and cleaning the speech-plus-noise subspace via a reweighted reconstruction.

Like the Wiener filter, the optimal filter based on the subspace approach does not take explicitly and fully advantage of the spatial information in order to minimize the distortion of the speech signal.

5.6 Spatio-Temporal Prediction Approach

As explained in the previous sections, the fact that speech is partially predictable helps all algorithms in reducing the level of noise in the microphone signal $y_1(k)$. Implicitly, temporal prediction of the signal of interest plays a fundamental role in speech enhancement. What about spatial prediction? Is its role as important as temporal prediction? Since the speech signals picked up by the microphones come from a unique source, the same signals at microphones $2, \ldots, N$ can be predicted from the first microphone signal. Can this help?

Now assume that we can find an $L \times L$ filter matrix, $\mathbf{W}_n$, such that

$$\mathbf{x}_n(k) = \mathbf{W}_n^T \mathbf{x}_1(k),\ n = 2, \ldots, N. \tag{5.50}$$

We will see later how to determine the optimal matrix, $\mathbf{W}_{n,\mathrm{o}}$. Expression (5.50) can be seen as a spatio-temporal prediction where we try to predict the microphone signal samples $\mathbf{x}_n(k)$ from $\mathbf{x}_1(k)$.

Substituting (5.50) into (5.5), we find that

$$\mathbf{e}_x(k) = \left(\mathbf{H}\mathbf{W}^T - \mathbf{I}_{L\times L}\right)\mathbf{x}_1(k), \tag{5.51}$$

where

$$\mathbf{W} = \left[\mathbf{I}_{L\times L}\ \mathbf{W}_2 \cdots \mathbf{W}_N\right]$$

is a matrix of size $L \times NL$.

In the single-channel case, there is no way we can reduce the level of the background noise without distorting the speech signal. In the Wiener filter

(with one or more microphones), we minimize the classical MSE without much concern on the residual noise and speech distortion. In the subspace approach, we minimize the speech distortion while keeping the residual noise power below a threshold. However, from the spatio-temporal prediction approach, we see clearly that by using at least two microphones it is possible to have noise reduction with no speech distortion [if (5.50) is met] by simply minimizing $J_v(\mathbf{H})$ with the constraint that $\mathbf{H}\mathbf{W}^T = \mathbf{I}_{L\times L}$. Therefore, our optimization problem is

$$\min_{\mathbf{H}} J_v(\mathbf{H}) \quad \text{subject to} \quad \mathbf{I}_{L\times L} = \mathbf{H}\mathbf{W}^T. \tag{5.52}$$

By using Lagrange multipliers, we easily find the optimal solution

$$\mathbf{H}_{\text{ST}} = \left(\mathbf{W}\mathbf{R}_{vv}^{-1}\mathbf{W}^T\right)^{-1}\mathbf{W}\mathbf{R}_{vv}^{-1}, \tag{5.53}$$

where we assumed that the noise signals $v_n(k),\ n = 1, 2, \ldots, N$, are not completely coherent so that $\mathbf{R}_{vv}$ is not singular. Expression (5.53) has the same form as the linearly constrained minimum variance (LCMV) beamformer (see Chapter 4) [54], [76]; however the spatio-temporal prediction based-approach is more general and certainly deals better, from a practical point of view, with the real acoustic environment where the spatial property is taken into account.

The second step is to determine the filter matrix $\mathbf{W}$ for spatio-temporal prediction. An optimal estimator, in the Wiener sense, can be obtained by minimizing the following cost function

$$J_{\text{f}}(\mathbf{W}_n) = E\left\{\left[\mathbf{x}_n(k) - \mathbf{W}_n^T\mathbf{x}_1(k)\right]^T\left[\mathbf{x}_n(k) - \mathbf{W}_n^T\mathbf{x}_1(k)\right]\right\}. \tag{5.54}$$

We easily find the optimal spatio-temporal prediction filter

$$\mathbf{W}_{n,\text{o}}^T = \mathbf{R}_{x_n x_1}\mathbf{R}_{x_1 x_1}^{-1}, \tag{5.55}$$

where $\mathbf{R}_{x_n x_1} = E\left[\mathbf{x}_n(k)\mathbf{x}_1^T(k)\right]$ and $\mathbf{R}_{x_1 x_1} = E\left[\mathbf{x}_1(k)\mathbf{x}_1^T(k)\right]$ are the cross-correlation and correlation matrices of the speech signals, respectively. However, the signals $x_n(k),\ n = 1, 2, \ldots, N$, are not observable so the Wiener filter matrix, as given in (5.55), can not be estimated in practice. But using $\mathbf{x}_n(k) = \mathbf{y}_n(k) - \mathbf{v}_n(k)$, we can verify that

$$\mathbf{R}_{x_n x_1} = \mathbf{R}_{y_n y_1} - \mathbf{R}_{v_n v_1},\ n = 1, 2, \ldots, N, \tag{5.56}$$

where $\mathbf{R}_{y_n y_1} = E\left[\mathbf{y}_n(k)\mathbf{y}_1^T(k)\right]$ and $\mathbf{R}_{v_n v_1} = E\left[\mathbf{v}_n(k)\mathbf{v}_1^T(k)\right]$. As a result,

$$\mathbf{W}_{n,\text{o}}^T = \left(\mathbf{R}_{y_n y_1} - \mathbf{R}_{v_n v_1}\right)\left(\mathbf{R}_{y_1 y_1} - \mathbf{R}_{v_1 v_1}\right)^{-1}. \tag{5.57}$$

The optimal filter matrix depends now only on the second order statistics of the observation and noise signals. The statistics of the noise signals can

be estimated during silences [when $s(k) = 0$] if we assume that the noise is stationary so that its statistics can be used for a next frame when the speech is active. We also assume that a voice activity detector (VAD) is available so that the Wiener filter matrix is estimated only when the speech source is active. Note that if the source does not move, the optimal matrix needs to be estimated only once. Finally, the optimal filter matrix based on spatio-temporal prediction is given by

$$\mathbf{H}_{\mathrm{ST}} = \left(\mathbf{W}_{\mathrm{o}}\mathbf{R}_{vv}^{-1}\mathbf{W}_{\mathrm{o}}^{T}\right)^{-1}\mathbf{W}_{\mathrm{o}}\mathbf{R}_{vv}^{-1}, \tag{5.58}$$

where

$$\mathbf{W}_{\mathrm{o}} = \left[\mathbf{I}_{L\times L}\ \mathbf{W}_{2,\mathrm{o}}\cdots\mathbf{W}_{N,\mathrm{o}}\right].$$

In general, we do not have exactly $\mathbf{x}_n(k) = \mathbf{W}_{n,\mathrm{o}}^{T}\mathbf{x}_1(k)$ so that some speech distortion is expected. But for large filter matrices, we can approach this equality so that this distortion can be kept low. In this case, it can be verified that

$$\upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{ST}}) \approx 0, \tag{5.59}$$

$$\xi_{\mathrm{sr}}(\mathbf{H}_{\mathrm{ST}}) \approx 1, \tag{5.60}$$

$$\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{ST}}) \approx \frac{L\sigma_{v_1}^2}{\mathrm{tr}\left[\left(\mathbf{W}_{\mathrm{o}}\mathbf{R}_{vv}^{-1}\mathbf{W}_{\mathrm{o}}^{T}\right)^{-1}\right]} \approx \frac{1}{\tilde{J}(\mathbf{H}_{\mathrm{ST}})} \geq 1, \tag{5.61}$$

which implies that

$$\mathrm{SNR}(\mathbf{H}_{\mathrm{ST}}) \approx \mathrm{SNR}\cdot\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{ST}}) \geq \mathrm{SNR}. \tag{5.62}$$

Also, since $\tilde{J}(\mathbf{H}_{\mathrm{W}}) \leq \tilde{J}(\mathbf{H}_{\mathrm{ST}})$, we have $\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{ST}}) \leq \xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{W}})$.

Clearly, we see that this approach has the potential to introduce minimum distortion to the speech signal thanks to the fact that the microphone observations of the source signal are spatially and temporally predictable.

5.7 Case of Perfectly Coherent Noise

In this section, we study the particular case where the noise signals at the microphones are perfectly coherent. This means that these signals are generated from a unique source as follows:

$$\begin{aligned} v_n(k) &= g_{b,n} * b(k) \\ &= \mathbf{g}_{b,n}^{T}\mathbf{b}(k),\ n = 1, 2, \ldots, N, \end{aligned} \tag{5.63}$$

where

$$\mathbf{g}_{b,n} = \left[\, g_{b,n,0}\ g_{b,n,1}\ \cdots\ g_{b,n,L-1} \,\right]^T$$

is the impulse response of length L from the noise source, $b(k)$, to the nth microphone, and $\mathbf{b}(k)$ is a vector containing the L most recent samples of the signal $b(k)$.

It can easily be checked that we have the following relations at time k [96], [125], [155]

$$\mathbf{v}_i^T(k)\mathbf{g}_{b,j} = \mathbf{v}_j^T(k)\mathbf{g}_{b,i},\ i,j = 1,2,\ldots,N. \tag{5.64}$$

Multiplying (5.64) by $\mathbf{v}_i(k)$ and taking expectation yields

$$\mathbf{R}_{v_iv_i}\mathbf{g}_{b,j} = \mathbf{R}_{v_iv_j}\mathbf{g}_{b,i},\ i,j = 1,2,\ldots,N. \tag{5.65}$$

This implies that the noise covariance matrix $\mathbf{R}_{vv}$ is not full rank and some of the methods presented in this chapter for noise reduction may not work well since it is required that the inverse of this matrix exists. In fact, we can show that if the impulse responses $\mathbf{g}_{b,n},\ n = 1,2,\ldots,N$, do not share any common zeroes and the autocorrelation matrix $\mathbf{R}_{bb} = E\left[\mathbf{b}(k)\mathbf{b}^T(k)\right]$ has full rank, the dimension of the null space of $\mathbf{R}_{vv}$ is equal to $(N-2)L+1$ for $N \geq 2$.

In this particular context, we propose to use an $NL \times L$ filter matrix $\mathbf{H}_{\mathrm{E}}^T$ (where the subscript 'E' stands for eigenvector) for noise reduction such that for:

- $N = 2$, the first column of $\mathbf{H}_{\mathrm{E}}^T$ is the (unique) eigenvector of $\mathbf{R}_{vv}$ corresponding to the eigenvalue 0 and the $L-1$ remaining columns are zeroes;
- $N > 2$, the L columns of $\mathbf{H}_{\mathrm{E}}^T$ are the L eigenvectors of $\mathbf{R}_{vv}$ corresponding to the eigenvalue 0 that can minimize speech distortion. (Since the dimension of the null space of $\mathbf{R}_{vv}$ can be much larger than L for $N > 2$, it is preferable to choose from this null space the eigenvectors that minimize speech distortion.)

With the choice of this filter matrix, we always have:

$$\mathbf{R}_{vv}\mathbf{H}_{\mathrm{E}}^T = \mathbf{0}_{NL\times L}. \tag{5.66}$$

As a result, we also deduce that

$$\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{E}}) = \infty, \tag{5.67}$$

$$\upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{E}}) = \frac{\tilde{J}(\mathbf{H}_{\mathrm{E}})}{\mathrm{SNR}} \geq \upsilon_{\mathrm{sd}}(\mathbf{H}_{\mathrm{W}}), \tag{5.68}$$

$$\mathrm{SNR}(\mathbf{H}_{\mathrm{E}}) = \infty. \tag{5.69}$$

We see that even in this context, we can do a pretty good job at noise reduction. On one hand, the fact that the noise signal at one microphone can be (spatially) predicted from any other microphone[1] can terribly affect the performance of some methods, on the other hand this fact can be exploited differently to perform noise reduction efficiently.

[1] This implies that the noise signals are perfectly coherent.

5.8 Adaptive Noise Cancellation

The objective of adaptive noise cancellation (ANC) is to eliminate the background noise by adaptively recreating the noise replica using a reference signal (of the noise field) [11], [231], [232]. In our context, it's difficult to have a true noise reference free of the speech signal. The best way to tackle this problem is to estimate the noise replica during silences. Therefore, we will try to find an estimator of the first microphone noise samples from the $N-1$ other microphone signals (which are considered as the noise reference) during noise only periods and use this estimate to attenuate the noise at microphone 1 during speech activity. However, contrary to the classical ANC method, speech distortion may be unavoidable here since speech may also be present at the noise reference (in other words, no clean noise reference is available).

With this in mind, the residual noise is now

$$\begin{aligned}\mathbf{e}_v(k) &= \mathbf{v}_1(k) + \sum_{n=2}^{N} \mathbf{H}_n \mathbf{v}_n(k) \\ &= \mathbf{H}\mathbf{v}(k)\end{aligned} \tag{5.70}$$

with $\mathbf{H}_1 = \mathbf{I}_{L\times L}$. To find the optimal estimator, we only need to solve the following optimization problem:

$$\min_{\mathbf{H}} J_v(\mathbf{H}) \quad \text{subject to} \quad \mathbf{I}_{L\times L} = \mathbf{H}\mathbf{U}^T, \tag{5.71}$$

for which the solution is

$$\mathbf{H}_{\mathrm{A}} = \left(\mathbf{U}\mathbf{R}_{vv}^{-1}\mathbf{U}^T\right)^{-1}\mathbf{U}\mathbf{R}_{vv}^{-1}, \tag{5.72}$$

where we assumed that the noise signals $v_n(k),\ n = 1, 2, \ldots, N$, are not perfectly coherent so that $\mathbf{R}_{vv}$ is a full rank matrix. The optimal filter matrix $\mathbf{H}_{\mathrm{A}}$ can be seen as a spatio-temporal linear predictor for the noise. Now suppose that this noise is spatially uncorrelated; in this case it's easy to see that $\mathbf{R}_{vv}$ is a block diagonal matrix. As a result, $\mathbf{H}_{\mathrm{A}} = \mathbf{U}$ and noise reduction is not possible. This is analogous to the classical ANC approach where the noise at the primary and auxiliary inputs should be at least partially coherent. Therefore, the noise must be somewhat spatially correlated in order for $\mathbf{H}_{\mathrm{A}}$ to have some effect on the microphone signals. The more coherent the noise is at the microphones, the more noise reduction is expected (and as a consequence more speech distortion).

We always have

$$\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{A}}) = \frac{L\sigma_{v_1}^2}{\mathrm{tr}\left[\left(\mathbf{U}\mathbf{R}_{vv}^{-1}\mathbf{U}^T\right)^{-1}\right]} \geq 1. \tag{5.73}$$

If the multichannel coherence of the noise is close to 0 then $\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{A}})$ is close to 1. On the other hand, if the multichannel coherence of the noise tends to 1 then $\xi_{\mathrm{nr}}(\mathbf{H}_{\mathrm{A}})$ tends to ∞.

5.9 Kalman Filter

The use of the Kalman filter for speech enhancement in the single-channel case, under the assumption that the noise is white, was first proposed by Paliwal and Basu [179]. A couple of years later, this technique was extended to the colored noise situation by Gibson et al. [86]. Until today we can still argue whether or not the Kalman filter is practical since some of the assumptions to make it work in speech applications may not be so realistic. For example, it is always assumed that the linear prediction (LP) model parameters of the clean speech are known, which is, of course, not true. However, some reasonable estimators can now be found in the literature [78], [150].

In this section, we attempt to generalize this concept to the multichannel case. Contrary to the methods presented in the previous sections, we will try to recover the speech source, $s(k)$, directly. So we perform both speech dereverberation and noise reduction.

We can rewrite the signal model given in (5.1) as

$$\mathbf{s}(k) = \mathbf{A}_s\mathbf{s}(k-1) + \mathbf{u}v_s(k), \tag{5.74}$$

$$\mathbf{y}_{\mathrm{a}}(k) = \mathbf{G}\mathbf{s}(k) + \mathbf{v}_{\mathrm{a}}(k), \tag{5.75}$$

where

$$\mathbf{A}_s = \begin{bmatrix} a_{s,1} & a_{s,2} & \cdots & a_{s,L-1} & a_{s,L} \\ 1 & 0 & \cdots & 0 & 0 \\ 0 & 1 & \cdots & 0 & 0 \\ \vdots & \vdots & \ddots & \vdots & \vdots \\ 0 & 0 & \cdots & 1 & 0 \end{bmatrix}_{L\times L} \tag{5.76}$$

with $a_{s,l}$ $(l = 1, 2, \ldots, L)$ being the LP coefficients of the signal $s(k)$,

$$\begin{aligned} \mathbf{s}(k) &= \left[s(k)\ s(k-1) \cdots s(k-L+1) \right]^T, \\ \mathbf{u} &= \left[1\ 0 \cdots 0 \right]^T, \\ \mathbf{y}_{\mathrm{a}}(k) &= \left[y_1(k)\ y_2(k) \cdots y_N(k) \right]^T, \\ \mathbf{v}_{\mathrm{a}}(k) &= \left[v_1(k)\ v_2(k) \cdots v_N(k) \right]^T, \\ \mathbf{G} &= \begin{bmatrix} \mathbf{g}_1^T \\ \mathbf{g}_2^T \\ \vdots \\ \mathbf{g}_N^T \end{bmatrix}_{N\times L}, \end{aligned}$$

and $v_s(k)$ is a white signal with variance $\sigma_{v_s}^2$. (Note that the notation for $\mathbf{y}_{\mathrm{a}}(k)$ and $\mathbf{v}_{\mathrm{a}}(k)$ is slightly different than the one given in Chapter 3.) To simplify the derivation of the algorithm, we suppose that the noise is spatially-temporally white and has the same variance, σ_v^2, at all microphones so that

$E\left[\mathbf{v}_{\mathrm{a}}(k)\mathbf{v}_{\mathrm{a}}^T(k)\right] = \sigma_v^2\mathbf{I}_{N\times N}$. Now assume that all the parameters $a_{s,l}$ $(l = 1, 2, \ldots, L)$, $\mathbf{G}$, $\sigma_{v_s}^2$, and σ_v^2 are known or can be estimated, an optimal estimate of $\mathbf{s}(k)$ can be obtained with the Kalman filter [141] (see also Chapter 2):

$$\mathbf{R}_{ee}(k|k-1) = \mathbf{A}_s\mathbf{R}_{ee}(k-1|k-1)\mathbf{A}_s^T + \sigma_{v_s}^2\mathbf{u}\mathbf{u}^T, \tag{5.77}$$

$$\mathbf{K}(k) = \mathbf{R}_{ee}(k|k-1)\mathbf{G}^T \times \left[\mathbf{G}\mathbf{R}_{ee}(k|k-1)\mathbf{G}^T + \sigma_v^2\mathbf{I}_{N\times N}\right]^{-1}, \tag{5.78}$$

$$\hat{\mathbf{s}}(k) = \mathbf{A}_s\hat{\mathbf{s}}(k-1) + \mathbf{K}(k)\left[\mathbf{y}_{\mathrm{a}}(k) - \mathbf{G}\mathbf{A}_s\hat{\mathbf{s}}(k-1)\right], \tag{5.79}$$

$$\mathbf{R}_{ee}(k|k) = \left[\mathbf{I}_{L\times L} - \mathbf{K}(k)\mathbf{G}\right]\mathbf{R}_{ee}(k|k-1), \tag{5.80}$$

where $\hat{\mathbf{s}}(k)$ is the estimate of $\mathbf{s}(k)$, $\mathbf{K}(k)$ is the Kalman gain matrix,

$$\mathbf{R}_{ee}(k|k-1) = E\left\{\left[\mathbf{s}(k) - \mathbf{A}_s\hat{\mathbf{s}}(k-1)\right]\left[\mathbf{s}(k) - \mathbf{A}_s\hat{\mathbf{s}}(k-1)\right]^T\right\}$$

is the predicted state-error covariance matrix, and

$$\mathbf{R}_{ee}(k|k) = E\left\{\left[\mathbf{s}(k) - \hat{\mathbf{s}}(k)\right]\left[\mathbf{s}(k) - \hat{\mathbf{s}}(k)\right]^T\right\}$$

is the filtered state-error covariance matrix. The algorithm is initialized as follows: $\hat{\mathbf{s}}(0) = E\left[\mathbf{s}(0)\right]$ and $\mathbf{R}_{ee}(0|0) = E\left[\mathbf{s}(0)\mathbf{s}^T(0)\right]$.

It is interesting to see that the generalization to the multiple microphone case is not only feasible but could also be more interesting than with one microphone only, since dereverberation is possible. This comes at a heavy price, though, since more parameters have to be known or estimated; especially the impulse responses from the source to the microphones. Blind estimation of these impulses is a possibility but needless to say that this multichannel Kalman filter may be even less practical than its single-channel counterpart. Many aspects of this approach still need to be investigated.

5.10 Simulations

We have carried out a number of simulations to experimentally study the three main algorithms (Wiener filter, subspace, and spatio-temporal prediction) in real acoustic environments under different operation conditions. In this section, we will present the results, which highlight the merits and limitations inherent in these noise-reduction techniques, and justify what we learned through theoretical analysis in the previous sections. In these experiments, we use the output SNR and speech-distortion index defined in Sect. 5.3 as the performance measures.

5.10.1 Acoustic Environments and Experimental Setup

The simulations were conducted with the impulse responses measured in the varechoic chamber at Bell Labs [101]. A diagram of the floor plan layout is

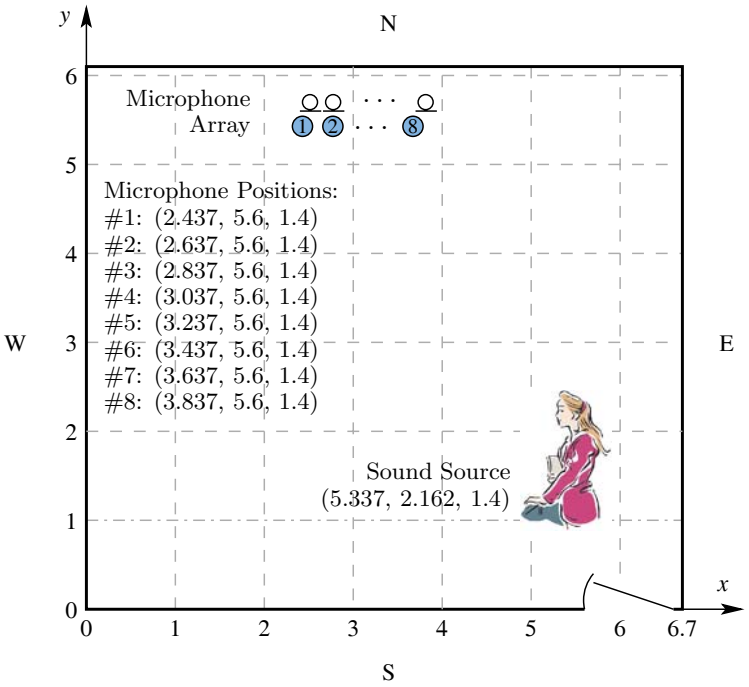


Fig. 5.1. Floor plan of the varechoic chamber at Bell Labs (coordinate values measured in meters).

shown in Fig. 5.1. For convenience, positions in the floor plan are designated by (x, y) coordinates with reference to the southwest corner and corresponding to meters along the (South, West) walls. The chamber measures $x = 6.7$ m wide by $y = 6.1$ m deep by $z = 2.9$ m high. It is a rectangular room with 368 electronically controlled panels that vary the acoustic absorption of the walls, floor, and ceiling [225]. Each panel consists of two perforated sheets whose holes, if aligned, expose sound absorbing material (fiberglass) behind, but if shifted to misalign, form a highly reflective surface. The panels are individually controlled so that the holes on one particular panel are either fully open (absorbing state) or fully closed (reflective state). Therefore, by varying the binary state of each panel in any combination, 2^{238} different room characteristics can be simulated. In the database of channel impulse responses from [101], there are four panel configurations with 89%, 75%, 30%, and 0% of panels open, respectively corresponding to approximately 240, 310, 380, and 580 ms 60-dB reverberation time T_{60} in the 20–4000 Hz band. In our study, all four configurations were used to evaluate the performance of the noise-reduction algorithms. But for conciseness and also due to space limitations, we present here only the results for the least and the most reverberant environments, i.e., $T_{60} = 240$ ms and 580 ms, respectively.

A linear microphone array which consists of 22 omni-directional microphones was employed in the measurement and the spacing between adjacent microphones is about 10 cm. The array was mounted 1.4 m above the floor and parallel to the North wall at a distance of 50 cm. A loudspeaker was placed at 31 different pre-specified positions to measure the impulse response to each microphone. In the simulations, no more than eight microphones will be chosen and the sound source is fixed at one loudspeaker position. The positions of the microphones and the sound source are shown in Fig. 5.1.

Signals were sampled at 8 kHz and the length of the measured impulse responses is of 4096 samples. Depending on the simulation specification, the sound source is either a female speech signal or a white Gaussian random signal. Then we compute the microphone outputs by convolving the source signal and the corresponding channel impulse responses. The additive noise is Gaussian and is white in both time and space. The SNR at the microphones is fixed at 10 dB. The source signal is 12 seconds long. The first 5 seconds of the microphone outputs are used to compute the initial estimates of $\mathbf{R}_{yy}$ and $\mathbf{R}_{vv}$. The last first 5 seconds are then used for performance evaluation of the noise-reduction algorithms. In this procedure, the estimates of $\mathbf{R}_{yy}$ and $\mathbf{R}_{vv}$ are recursively updated according to

$$\mathbf{R}_{yy}(k) = \lambda \mathbf{R}_{yy}(k-1) + (1-\lambda)\mathbf{y}(k)\mathbf{y}^T(k), \tag{5.81}$$

$$\mathbf{R}_{vv}(k) = \lambda \mathbf{R}_{vv}(k-1) + (1-\lambda)\mathbf{v}(k)\mathbf{v}^T(k), \tag{5.82}$$

where $0 < \lambda < 1$ is the forgetting factor. Intuitively, it can be of some benefits to choose different values of λ for $\mathbf{R}_{yy}(k)$ and $\mathbf{R}_{vv}(k)$, since the statistics of speech and noise generally vary in different rates in practice. But for simplicity, we always specify the same forgetting factor for $\mathbf{R}_{yy}(k)$ and $\mathbf{R}_{vv}(k)$ in one experiment. Therefore, we do not differentiate the forgetting factors in (5.81) and (5.82).

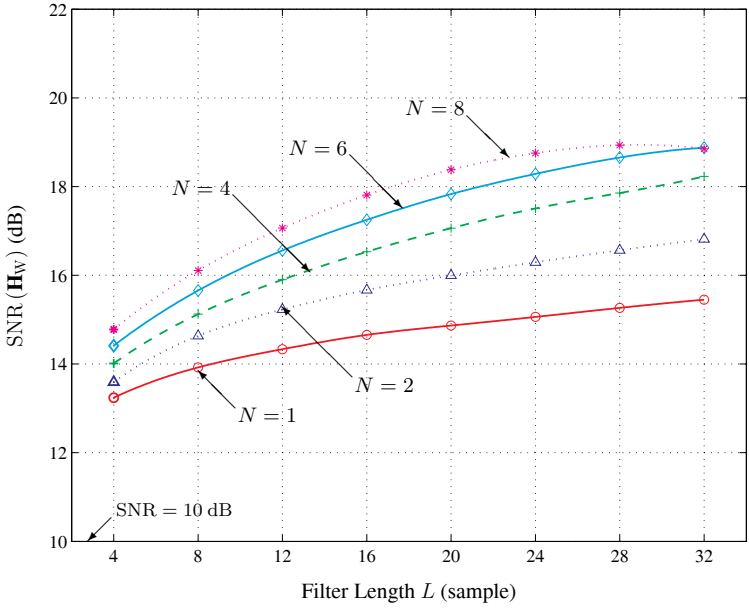
5.10.2 Experimental Results

Experiment 1: Wiener Filter with Various Numbers of Microphones and Filter Lengths.

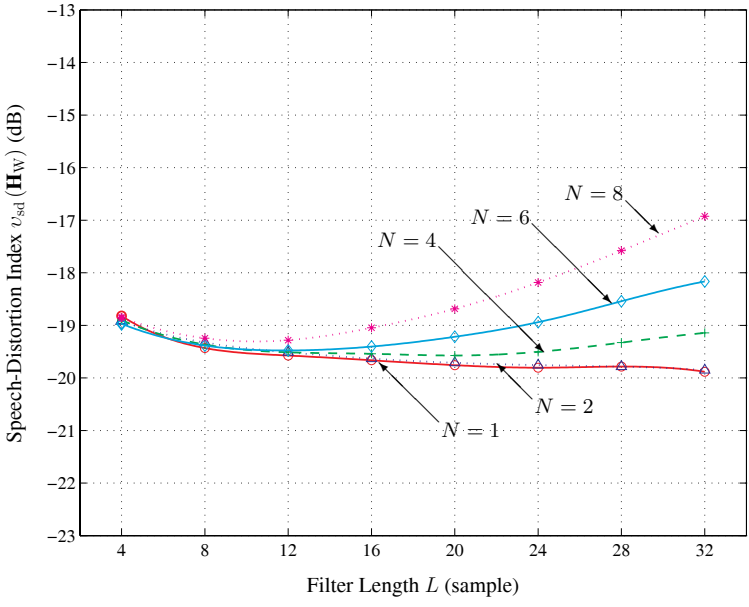
Let us first investigate the Wiener filter algorithm for noise reduction using various numbers of microphones and filter lengths. The performance of the optimal Wiener filter obtained here will be used as a benchmark for comparison with other noise-reduction algorithms in the following experiments.

The experiment was conducted with the acoustic impulse responses being measured for 89% open panels, i.e., $T_{60} = 240$ ms. The source is a female speech signal and we take $\lambda = 0.9975$. The output SNR and speech-distortion index are plotted in Fig. 5.2.

We see from Fig. 5.2 that the output SNR of the Wiener filter is significantly improved by using more microphones and longer filters, which at the



(a)



(b)

Fig. 5.2. Performance of the Wiener filter for noise reduction using various numbers of microphones $N = 1,\ 2,\ 4,\ 6,$ and 8, respectively. (a) Output SNR, and (b) speech-distortion index. Input SNR = 10 dB, room reverberation time $T_{60} = 240$ ms, and the forgetting factor $\lambda = 0.9975$.

same time introduces more speech distortion. This trade-off is more prominent when N is relatively large. By increasing N from 1 to 2, we observe that the change in speech-distortion index is hardly noticeable while the output SNR is boosted by more or less 1 dB. But increasing N from 6 to 8 leads to less than 0.5 dB gain in the SNR as well as approximately 0.5 dB loss in speech distortion. It is implied by this set of results that the Wiener filter is in favor of using multiple, while a small number of, microphones, and a moderate filter length. In particular for an application in which speech distortion is highly concerned, we should avoid to deploy a large array with long Wiener filters.

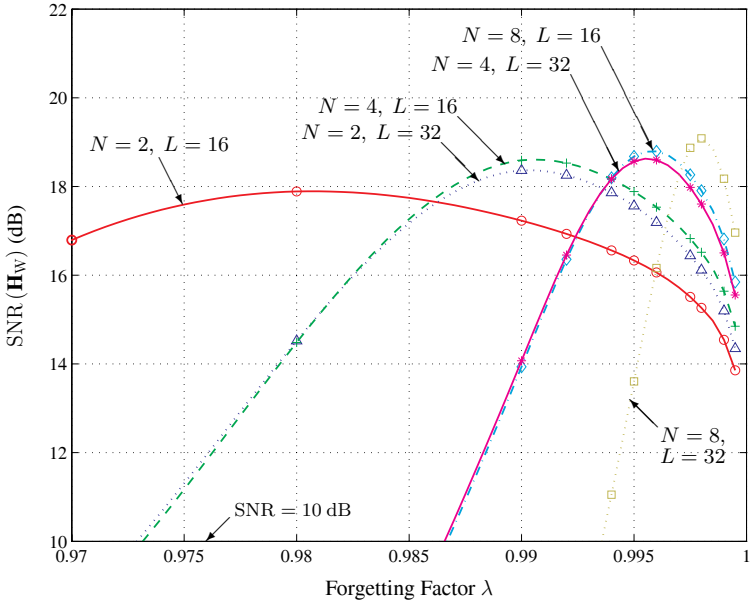
Experiment 2: Effect of the Forgetting Factor on the Performance of the Wiener Filter.

In the development of the Wiener filter as well as other algorithms for noise reduction, we assume the knowledge of $\mathbf{R}_{yy}$ and $\mathbf{R}_{vv}$. As a result, one may unfortunately overlook the importance and underevaluate the difficulty of accurately estimating these statistics (though they are only second order) in practice. Actually the forgetting factor plays a critical role in tuning a noise-reduction algorithm. On one hand, if the forgetting factor is too large (close to 1), the recursive estimate of $\mathbf{R}_{yy}(k)$ according to (5.81) is essentially a long-term average and cannot follow the short-term variation of speech signals. Consequently the potential for greater noise reduction is not fully taken advantage of. On the other hand, if the forgetting factor is too small (much less than 1), then the recursive estimate of $\mathbf{R}_{yy}(k)$ is more likely rank deficient. This leads to the numerical stability problem when computing the inverse of $\mathbf{R}_{yy}(k)$, and hence causes performance degradation. Therefore, a proper forgetting factor is the one that helps achieve the balance between tracking capability and numerical stability. In this experiment, we would like to study this effect of the forgetting factor. We consider the Wiener filter again in the environment of $T_{60} = 240$ ms.

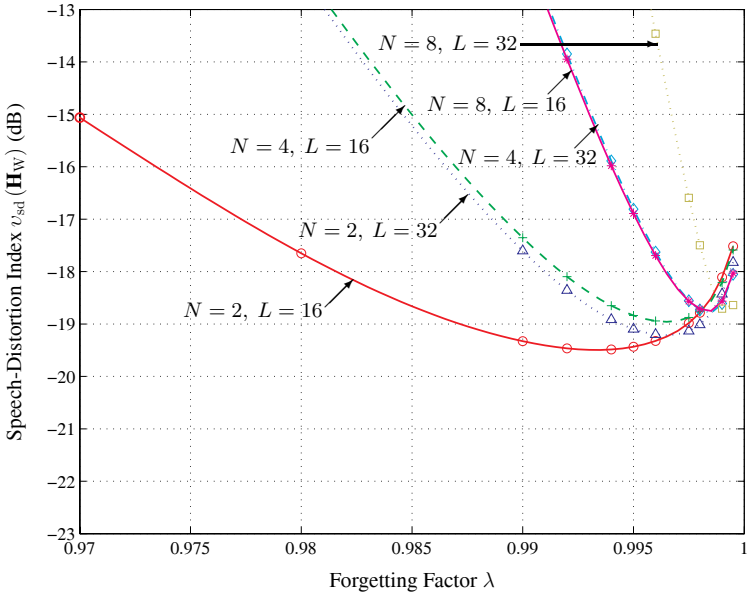
Figure 5.3 depicts the results of six systems under investigation. These curves visibly justify the trade-off effect mentioned above. Note that the size of $\mathbf{R}_{yy}(k)$ is $NL \times NL$. It is clear from Fig. 5.3 that the greater NL and the larger the size of $\mathbf{R}_{yy}(k)$, the greater is the optimal forgetting factor. The Wiener filters with the same value of NL perform almost identically against the forgetting factor regardless of the combination of N and L.

Experiment 3: Effect of Room Reverberation on the Performance of the Wiener Filter.

This experiment was designed to test the Wiener filter in different acoustic environments. We consider a system with $N = 4$ and $\lambda = 0.9975$. Both female speech and white Gaussian noise source signals were evaluated. The room reverberation time $T_{60} = 240$ ms and 580 ms. The experimental results are visualized in Fig. 5.4. We see that the performance of the Wiener filter with

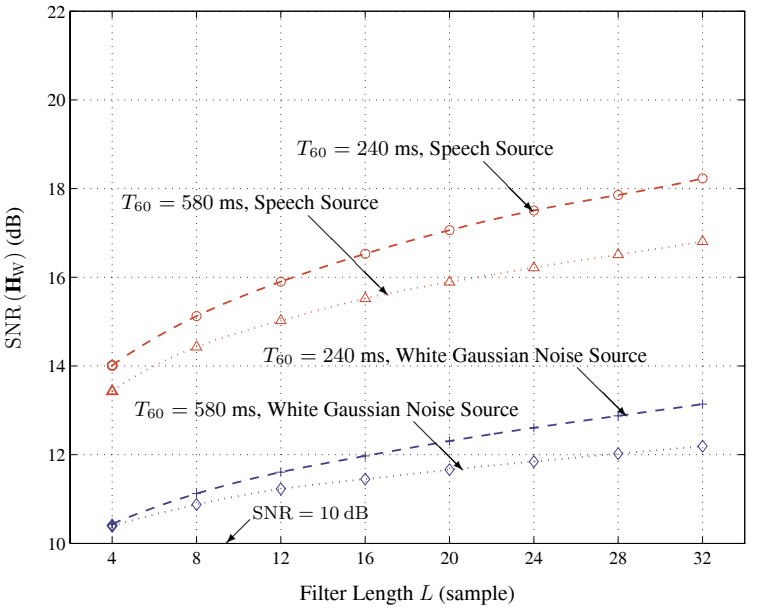


(a)

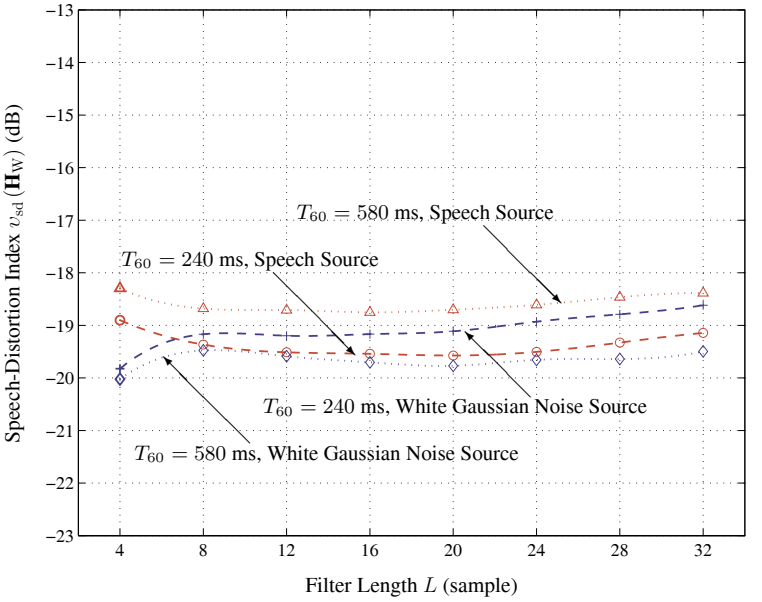


(b)

Fig. 5.3. Effect of the forgetting factor on the performance of the Wiener filter for noise reduction. (a) Output SNR, and (b) speech-distortion index. Input SNR = 10 dB and room reverberation time $T_{60} = 240$ ms.



(a)



(b)

Fig. 5.4. Effect of room reverberation on the performance of the Wiener filter for noise reduction using both speech and white Gaussian noise as the source signal. (a) Output SNR, and (b) speech-distortion index. Input SNR = 10 dB, the number of microphones $N = 4$, and the forgetting factor $\lambda = 0.9975$.

respect to the speech source is much better than that with respect to the white noise source. This is simply because noise is white while speech is predictable in time. A more reverberant channel does not make a speech signal more predictable. Therefore, the Wiener filter performs apparently better in a less reverberant environment ($T_{60} = 240$ ms) than in a more reverberant environment ($T_{60} = 580$ ms). But room reverberation colorizes the white noise source signal, making it somehow predictable in time. Consequently, we see while the output SNR for $T_{60} = 240$ ms is still better than that for $T_{60} = 580$ ms, the distortion is less for $T_{60} = 580$ ms.

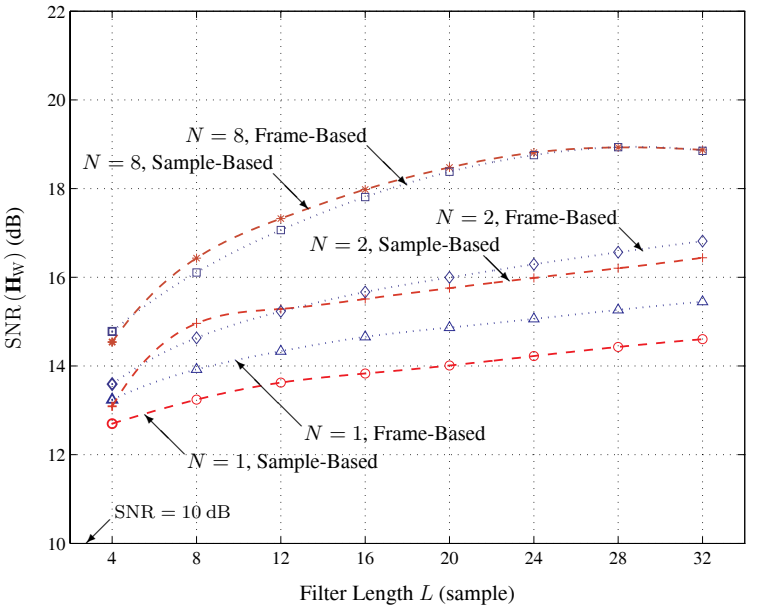
Experiment 4: Performance Comparison Between the Sample- and Frame-Based Implementations of the Wiener Filter.

The Wiener filter algorithm developed in Sect. 5.4 is a frame-based implementation to better fit into the unified framework for noise reduction that is explored in this chapter. However, the traditional sample-based implementation of the Wiener filter can be easily derived following the same principle and procedure. It is given without proof (left to the readers) that the noise-reduction output $z(k)$ from the sample-based implementation is exactly the same as that from the frame-based implementation when k is multiple times of L (the first sample in a frame). But this is not true for the rest of samples in the frame for $L > 1$. In the sample-based implementation the speech signal at one point is always predicted by using the past samples (i.e., via forward prediction), while in the frame-based implementation forward and backward prediction as well as interpolation are all possibly utilized. Since speech is highly correlated with its neighboring (either previous or prospective) samples, it would be difficult to tell, using only intuition, which implementation could yield a better performance. So we intend to quantitatively study it in this experiment.

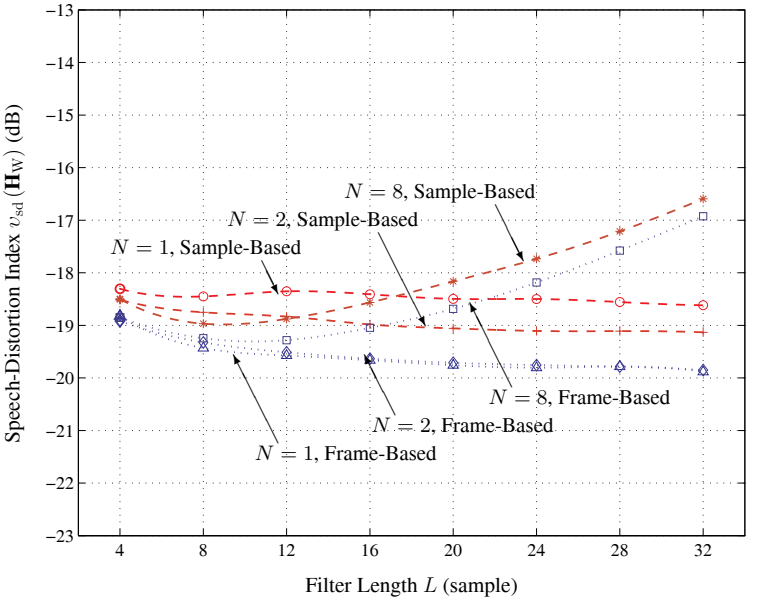
We consider the Wiener filter with $T_{60} = 240$ ms and $\lambda = 0.9975$. The source is the female speech signal. Figure 5.5 shows the results. We observe that for $N = 1$ the frame-based implementation is apparently better than the sample-based in terms of both output SNR and speech distortion. For $N = 2$ and 8, while the output SNR's for the two implementations are comparable, the frame-based produces less speech distortion (approximately 0.5 dB) than the sample-based. Therefore our preference leans to the frame-based implementation. As a matter of fact, the Wiener filters used in the three experiments above are all frame based.

Experiment 5: Performance Evaluation of the Subspace Method.

In the first four experiments, we studied the Wiener filter for noise reduction under various operation conditions. Now we turn to the subspace method. Again, we take $T_{60} = 240$ ms and $\lambda = 0.9975$. The number of microphones is either 2 or 6, and μ varies from 0.5, 1.0, to 2.0. Note that when $\mu = 1$,



(a)



(b)

Fig. 5.5. Performance comparison between the sample- and frame-based implementations of the Wiener filter algorithm for noise reduction. (a) Output SNR, and (b) speech-distortion index. Input SNR = 10 dB, room reverberation time $T_{60} = 240$ ms, and the forgetting factor $\lambda = 0.9975$.

the subspace method is essentially equivalent to the Wiener filter. The results are plotted in Fig. 5.6. It is evident that by decreasing μ, speech distortion is reduced but we gain little noise reduction. In the opposite direction, increasing μ results in low residual noise at the expense of high speech distortion.

Experiment 6: Performance Evaluation of the Spatio-Temporal Prediction Approach.

In the last but probably the most interesting experiment, we tested the novel spatio-temporal prediction approach to noise reduction in comparison with the Wiener filter.

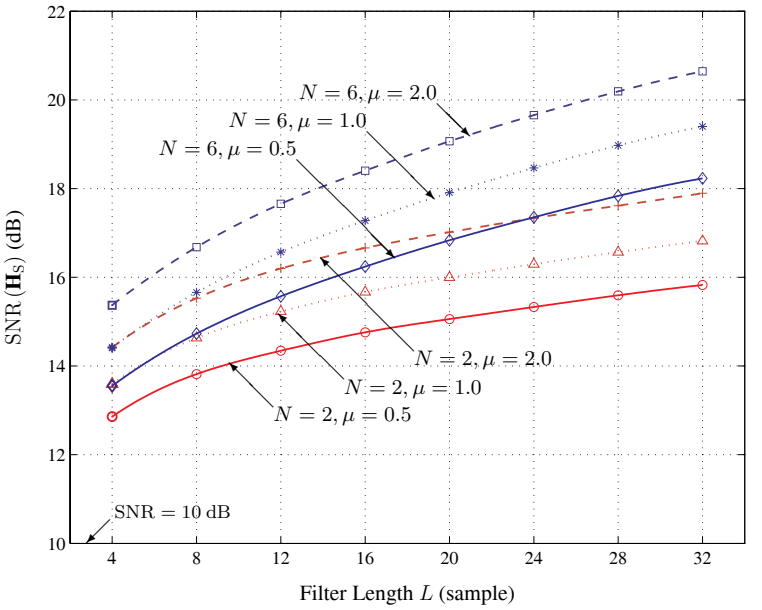
In our study, we learned that the performance of the Wiener filter and the subspace method is limited by the aforementioned numerical stability problem. By inspecting (5.16) and (5.39), we know that in the Wiener filter and subspace algorithms, we need to compute the inverse of $\mathbf{R}_{yy}$, which is of dimension $NL \times NL$. When we intend to use more microphones and longer filters (i.e., larger N and L) for a greater output SNR as well as less speech distortion, the covariance matrix $\mathbf{R}_{yy}$ becomes larger in size, which leads to the following two drawbacks:

- using a short-term average, a larger error can be expected in the estimate $\mathbf{R}_{yy}(k)$. But with a long-term average, the variation of speech statistics cannot be well followed. Both cause performance degradation. The larger $\mathbf{R}_{yy}$, the more prominent is the dilemma;
- the estimate of the covariance matrix $\mathbf{R}_{yy}(k)$ becomes more ill-conditioned (with a larger condition number) when NL gets larger. As a result, it is more problematic to find its inverse.

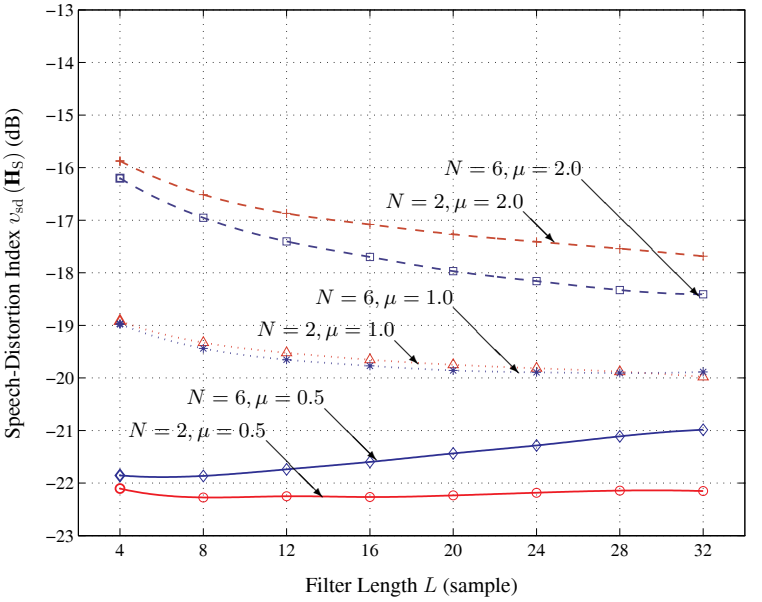
Therefore, as revealed by the results in the previous experiments, we do not gain what we expect from the Wiener filter and subspace algorithms by increasing N and L.

Alternatively, the spatio-temporal prediction approach utilizes the spatial and temporal correlation among the outputs of a microphone array with respect to a speech source separately in two steps. If we look closer at (5.57), we can recognize that the spatio-temporal prediction is proceeded on a pair-by-pair basis. In this procedure, only $\mathbf{R}_{x_1x_1}$ or equivalently $(\mathbf{R}_{y_1y_1} - \mathbf{R}_{v_1v_1})$ needs to be inverted. This matrix is $L \times L$ and does not grow in size with the number of microphones that we use. In addition, from (5.53), we know that $\mathbf{R}_{vv}$ rather than $\mathbf{R}_{yy}$ needs to be inverted in computing $\mathbf{H}_{\mathrm{ST}}$. In most applications, the noise signals are white and relatively more stationary. Consequently, $\mathbf{R}_{vv}$ has a low condition number and can be accurately estimated with a long-term average. Therefore, with the spatio-temporal prediction algorithm, we can use a larger system with more microphones and longer filters for better performance.

Figure 5.7 shows the results of the performance comparison between the spatio-temporal prediction and Wiener filter algorithms, and in Fig. 5.8 we

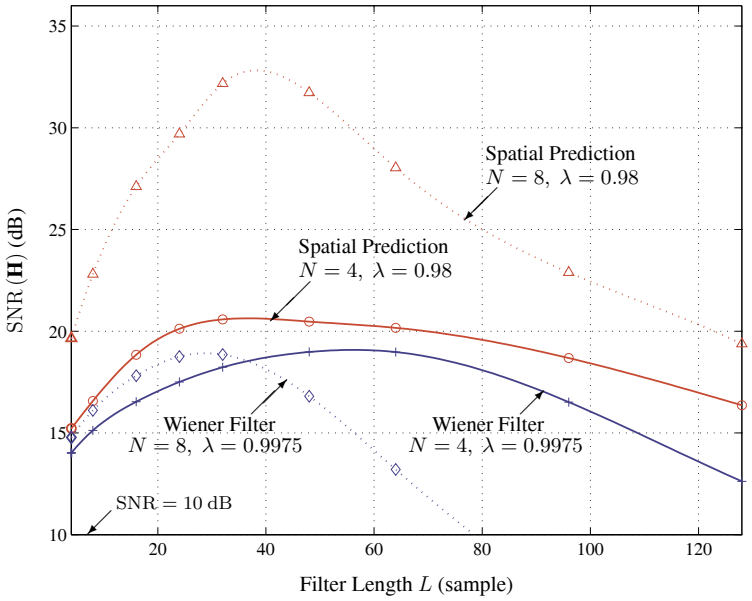


(a)

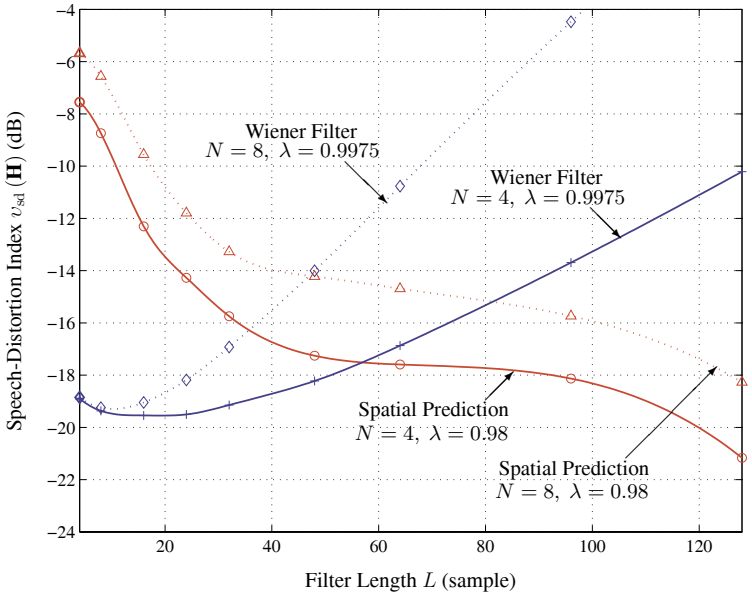


(b)

Fig. 5.6. Performance of the subspace algorithm for noise reduction using different values for μ and various numbers of microphones. (a) Output SNR, and (b) speech-distortion index. Input SNR = 10 dB, room reverberation time T_{60} = 240 ms, and the forgetting factor $\lambda = 0.9975$.

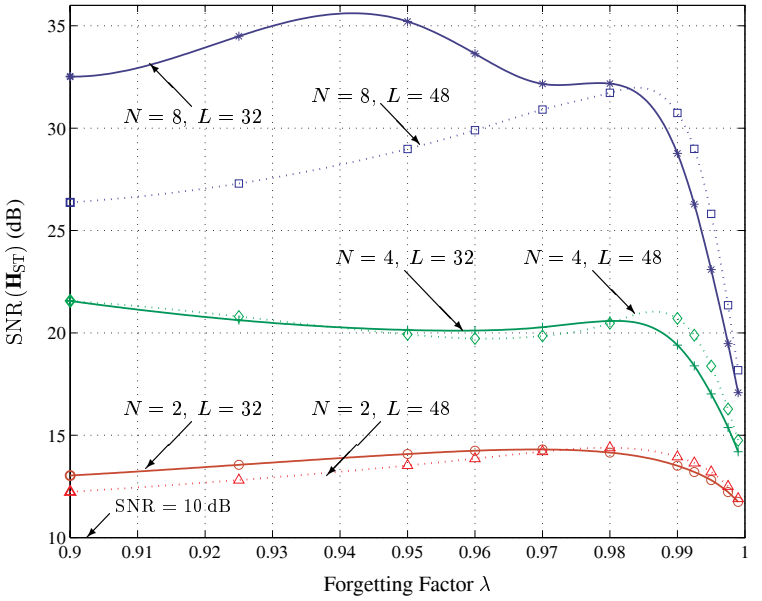


(a)

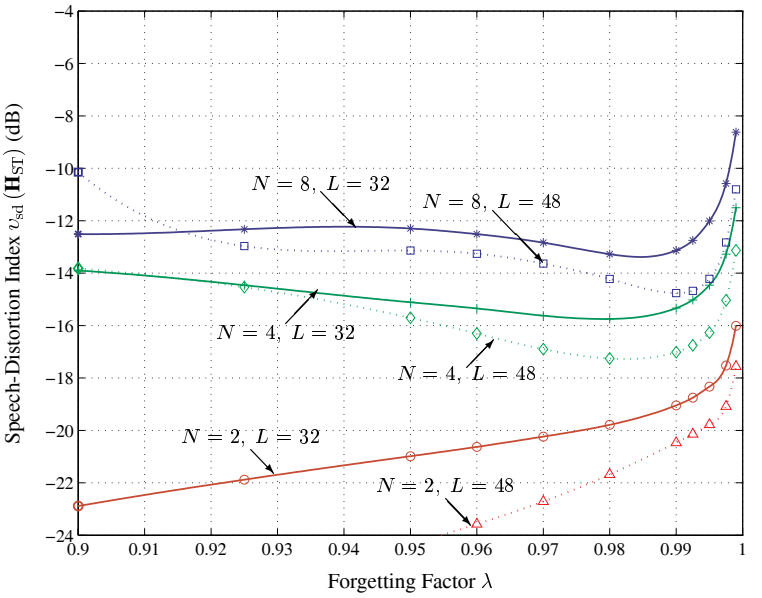


(b)

Fig. 5.7. Performance comparison between the spatio-temporal prediction and the Wiener filter algorithms for noise reduction using four and eight microphones. (a) Output SNR, and (b) speech-distortion index. Input SNR = 10 dB and room reverberation time T_{60} = 240 ms. The forgetting factor $\lambda = 0.9975$ and 0.98 for the Wiener filter and the spatio-temporal prediction algorithms, respectively.



(a)



(b)

Fig. 5.8. Effect of the forgetting factor on the performance of the spatio-temporal prediction algorithm for noise reduction. (a) Output SNR, and (b) speech-distortion index. Input SNR = 10 dB and room reverberation time $T_{60} = 240$ ms.

visualize the performance sensitivity of the spatio-temporal prediction algorithm to the change of the forgetting factor. Note that we use here different scales in both x- and y-axes from those that we have been using in the previous experiments, because we want to explore the use of larger N and L with the spatio-temporal prediction algorithm. We see that the spatio-temporal prediction algorithm yields much higher output SNR's. While its speech distortion is large at small L's, it greatly improves when L increases. Let us compare the best cases for the Wiener filter and spatio-temporal prediction. For $N = 4$, the highest output SNR that the Wiener filter delivers is about 18 dB when $L \approx 56$. In this case, the speech distortion of the Wiener filter is comparable to that of the spatio-temporal prediction algorithm. But the latter can produce approximately 21 dB output SNR, which is 3 dB higher than that with the Wiener filter. In addition, with the spatio-temporal prediction, we can easily meet the requirements imposed by an application. If a very high output SNR is desired with moderate speech distortion, we can take more microphones and relatively small L. On the contrary, if speech distortion is very much concerned with and only some SNR improvement is expected, we can use less microphones and long filters. Finally, comparing Fig. 5.8 to Fig. 5.3, we see that the performance of the spatio-temporal prediction algorithm is not sensitive to λ and is almost a function of L instead of NL. These features make the spatio-temporal prediction algorithm very appealing in practice.

5.11 Conclusions

Noise reduction is a very difficult problem and still remains a challenge today even after forty years of tremendous progress. While some useful and interesting solutions exist in the single-microphone case at the price of distorting the desired speech signal, we will not draw the same conclusion with multiple microphones. From a theoretical point of view, though, it is possible to reduce noise with no speech distortion with a microphone array. However, the derivation of a practical solution is still an open area of research. This chapter has shown the potentials and limitations of various methods. It is clear that the spatio-temporal prediction approach is the most promising one. In the next chapter we will continue our discussion on noise reduction but in the frequency domain.

6

Noncausal (Frequency-Domain) Optimal Filters

6.1 Introduction

The causal and noncausal Wiener filters have played and continue to play a fundamental role in many aspects of signal processing since their invention by Norbert Wiener in the 1940s [234]. If the signal of interest is corrupted by an additive noise, the output of the Wiener filter whose input is the noisy signal (observation) is an optimal estimate, in the mean-square error sense, of the signal of interest. However, this optimal filter is far to be perfect since, as we all know, it distorts the desired signal [15], [40]. Despite this inconvenience, the Wiener filter is popular and widely used in many applications. One of these applications that has adopted this optimal filter for a long time is speech enhancement whose formulation is identical to the formulation of the general problem from which the Wiener filter is derived. As a result, the Wiener filter and many of its variants have significantly contributed for the progress towards a viable solution to the noise-reduction problem [16], [154], [156], [218].

The literature is extremely rich in algorithms for noise reduction in the time and frequency domains (see the introduction of the previous chapter). The focus of this chapter is on the noncausal Wiener filter only (and some versions of it), which is a frequency-domain approach that is always better to use in practice (with some approximations) than a time-domain causal Wiener filter, since it allows an individual control, at each frequency, between noise reduction and speech distortion.

This chapter is organized as follows. In Section 6.2 we define the signal model and clearly formulate the problem. Section 6.3 gives some very important definitions that will help the reader better understand the noise-reduction problem. Section 6.4 develops and studies the classical noncausal Wiener filter. In Section 6.5, the parametric Wiener filtering is explained. Section 6.6 generalizes all the single-channel methods to the multichannel case and shows the fundamental role of the spatial diversity in the derivation of algorithms that do not distort the desired signal, which is extremely important in practice. Finally, we conclude in Section 6.7.

6.2 Signal Model and Problem Formulation

The noise-reduction problem considered in this chapter is to recover the signal of interest (clean speech) $x(k)$ of zero-mean from the noisy observation (microphone signal)

$$y(k) = x(k) + v(k), \tag{6.1}$$

where $v(k)$ is the unwanted additive noise, which is assumed to be a zero-mean random process (white or colored) and uncorrelated with $x(k)$. In the frequency domain, (6.1) can be rewritten as

$$Y(j\omega) = X(j\omega) + V(j\omega), \tag{6.2}$$

where j is the imaginary unit ($j^2 = -1$), and $Y(j\omega)$, $X(j\omega)$, and $V(j\omega)$ are respectively the discrete-time Fourier transforms (DTFTs) of $y(k)$, $x(k)$, and $v(k)$, at angular frequency ω ($-\pi < \omega \leq \pi$). Another possible form for (6.2) is

$$Y(\omega)e^{j\varphi_y(\omega)} = X(\omega)e^{j\varphi_x(\omega)} + V(\omega)e^{j\varphi_v(\omega)}, \tag{6.3}$$

where for any random signal $A(j\omega) = A(\omega)e^{j\varphi_a(\omega)}$, $A(\omega)$ and $\varphi_a(\omega)$ are its amplitude and phase at frequency ω, $A \in \{Y, X, V\}$, $a \in \{y, x, v\}$. We recall that the DTFT and the inverse transform [176] are

$$A(j\omega) = \sum_{k=-\infty}^{\infty} a(k)e^{-j\omega k}, \tag{6.4}$$

$$a(k) = \frac{1}{2\pi}\int_{-\pi}^{\pi} A(j\omega)e^{j\omega k}d\omega. \tag{6.5}$$

Using the power spectral density (PSD) and the fact that $x(k)$ and $v(k)$ are uncorrelated, we get

$$\phi_{yy}(\omega) = \phi_{xx}(\omega) + \phi_{vv}(\omega), \tag{6.6}$$

where

$$\begin{aligned}\phi_{aa}(\omega) &= E\left[|A(j\omega)|^2\right] \\ &= E\left[A^2(\omega)\right]\end{aligned} \tag{6.7}$$

is the PSD of the signal $a(k)$ [for which the DTFT is $A(j\omega)$].

An estimate of $X(j\omega)$ can be obtained by passing $Y(j\omega)$ through a linear filter, i.e.,

$$\begin{aligned}Z(j\omega) &= H(j\omega)Y(j\omega) \\ &= H(j\omega)\left[X(j\omega) + V(j\omega)\right],\end{aligned} \tag{6.8}$$

where $Z(j\omega)$ is the frequency representation of the signal $z(k)$. The PSD of $z(k)$ is then

$$\begin{aligned}\phi_{zz}(\omega) &= |H(j\omega)|^2 \phi_{yy}(\omega) \\ &= |H(j\omega)|^2 [\phi_{xx}(\omega) + \phi_{vv}(\omega)] .\end{aligned} \tag{6.9}$$

Our main concern in the rest of this chapter is the design of the filter $H(j\omega)$ and its study.

6.3 Performance Measures

Like in Chapters 2 and 5, before we discuss the algorithms, we give some very useful definitions that are important for designing properly the filter $H(j\omega)$. These definitions will also help us better understand how noise reduction works in the frequency domain.

The input SNR at frequency ω, that we will call the input narrowband SNR [141], is

$$\text{SNR}(\omega) = \frac{\phi_{xx}(\omega)}{\phi_{vv}(\omega)}. \tag{6.10}$$

We define the input fullband SNR as

$$\begin{aligned}\text{SNR} &= \frac{\int_{-\pi}^{\pi} \phi_{xx}(\omega) d\omega}{\int_{-\pi}^{\pi} \phi_{vv}(\omega) d\omega} \\ &= \frac{\sigma_x^2}{\sigma_v^2},\end{aligned} \tag{6.11}$$

where

$$\sigma_x^2 = E\left[x^2(k)\right] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \phi_{xx}(\omega) d\omega \tag{6.12}$$

and

$$\sigma_v^2 = E\left[v^2(k)\right] = \frac{1}{2\pi} \int_{-\pi}^{\pi} \phi_{vv}(\omega) d\omega \tag{6.13}$$

are the variances of the signals $x(k)$ and $v(k)$, respectively.

By analogy to the time-domain definitions [15], [40], [125], we define the noise-reduction factor at frequency ω as the ratio of the PSD of the noise over the PSD of the residual noise:

$$\begin{aligned}\xi_{\text{nr}} [H(j\omega)] &= \frac{\phi_{vv}(\omega)}{|H(j\omega)|^2 \phi_{vv}(\omega)} \\ &= \frac{1}{|H(j\omega)|^2}.\end{aligned} \tag{6.14}$$

The larger the value of $\xi_{\text{nr}}\left[H(j\omega)\right]$, the more the noise is reduced at frequency ω. After the filtering operation, the residual noise level at frequency ω is expected to be lower than that of the original noise level, therefore this factor should be lower bounded by 1. The fullband noise-reduction factor is

$$\begin{aligned}\xi_{\text{nr}}(H) &= \frac{\int_{-\pi}^{\pi}\phi_{vv}(\omega)d\omega}{\int_{-\pi}^{\pi}\left|H(j\omega)\right|^2\phi_{vv}(\omega)d\omega}\\ &= \frac{\int_{-\pi}^{\pi}\phi_{vv}(\omega)d\omega}{\int_{-\pi}^{\pi}\xi_{\text{nr}}^{-1}\left[H(j\omega)\right]\phi_{vv}(\omega)d\omega}.\end{aligned} \tag{6.15}$$

The previous expression is the ratio of the energy of the noise over the weighted energy of the noise with the weighting $\xi_{\text{nr}}^{-1}\left[H(j\omega)\right]$. Same as in (6.14), $\xi_{\text{nr}}(H)$ is expected to be lower bounded by 1. Indeed, if $\xi_{\text{nr}}\left[H(j\omega)\right] \geq 1, \forall\omega$, we deduce from (6.15) that $\xi_{\text{nr}}(H) \geq 1$.

The filtering operation distorts the speech signal, so we define the narrowband speech-distortion index as

$$\begin{aligned}\upsilon_{\text{sd}}\left[H(j\omega)\right] &= \frac{E\left[\left|X(j\omega) - H(j\omega)X(j\omega)\right|^2\right]}{\phi_{xx}(\omega)}\\ &= \left|1 - H(j\omega)\right|^2.\end{aligned} \tag{6.16}$$

This speech-distortion index is lower bounded by 0 and expected to be upper bounded by 1 for optimal filters. The higher the value of $\upsilon_{\text{sd}}\left[H(j\omega)\right]$, the more the speech is distorted at frequency ω. The fullband speech-distortion index is

$$\begin{aligned}\upsilon_{\text{sd}}(H) &= \frac{\int_{-\pi}^{\pi}E\left[\left|X(j\omega) - H(j\omega)X(j\omega)\right|^2\right]d\omega}{\int_{-\pi}^{\pi}\phi_{xx}(\omega)d\omega}\\ &= \frac{\int_{-\pi}^{\pi}\phi_{xx}(\omega)\left|1 - H(j\omega)\right|^2 d\omega}{\int_{-\pi}^{\pi}\phi_{xx}(\omega)d\omega}\\ &= \frac{\int_{-\pi}^{\pi}\upsilon_{\text{sd}}\left[H(j\omega)\right]\phi_{xx}(\omega)d\omega}{\int_{-\pi}^{\pi}\phi_{xx}(\omega)d\omega}.\end{aligned} \tag{6.17}$$

Equation (6.17) is the ratio of the weighted energy of the speech with the weighting $\upsilon_{\text{sd}}\left[H(j\omega)\right]$ over the energy of the speech. If $\upsilon_{\text{sd}}\left[H(j\omega)\right] \leq 1,\ \forall\omega$, we see from (6.17) that $\upsilon_{\text{sd}}(H) \leq 1$.

It is interesting to notice that the narrowband noise-reduction factor and speech-distortion index depend only on the filter $H(j\omega)$ while the same measures from a fullband point of view depend also on the PSDs of the noise and speech. This quite surprising observation shows that these two measures behave differently locally and globally. The nature of the signals has no effect locally but it has its importance, obviously, globally.

After the filtering operation [eq. (6.9)], the output SNR at frequency ω is

$$\begin{aligned} \text{oSNR}(\omega) &= \frac{|H(j\omega)|^2 \phi_{xx}(\omega)}{|H(j\omega)|^2 \phi_{vv}(\omega)} \\ &= \text{SNR}(\omega). \end{aligned} \tag{6.18}$$

The previous expression shows that the filtering operation in (6.8) does not affect the SNR locally. The output fullband SNR is defined as

$$\begin{aligned} \text{oSNR}(H) &= \frac{\int_{-\pi}^{\pi} |H(j\omega)|^2 \phi_{xx}(\omega) d\omega}{\int_{-\pi}^{\pi} |H(j\omega)|^2 \phi_{vv}(\omega) d\omega} \\ &= \frac{\int_{-\pi}^{\pi} \xi_{\text{nr}}^{-1}\left[H(j\omega)\right] \phi_{xx}(\omega) d\omega}{\int_{-\pi}^{\pi} \xi_{\text{nr}}^{-1}\left[H(j\omega)\right] \phi_{vv}(\omega) d\omega} \\ &\neq \text{SNR}. \end{aligned} \tag{6.19}$$

The output fullband SNR is the ratio of the weighted energy of the speech over the weighted energy of the noise with the same weighting $\xi_{\text{nr}}^{-1}\left[H(j\omega)\right]$. Contrary to the output narrowband SNR, the output fullband SNR is affected by the filter $H(j\omega)$, which is obviously a desirable thing to have. Expression (6.19) shows that the noise-reduction factor at each frequency and the nature of the signals have an important impact on the output SNR. Also, we can see from (6.18)–(6.19) that if

- $\text{SNR}(\omega) = 1,\ \forall\omega$ (the speech and noise are identical), then $\text{oSNR}(H) = 1$ (no improvement),
- $\text{SNR}(\omega) < 1,\ \forall\omega$, then $\text{oSNR}(H) < 1$ (if the SNR in every frequency band is less than 0 dB, then the output fullband SNR can never exceed 0 dB),
- $\text{SNR}(\omega) > 1,\ \forall\omega$, then $\text{oSNR}(H) > 1$.

Before finishing this section, we give the definition of a measure that will be extremely useful in the study of the filter $H(j\omega)$. Let $a(k)$ and $b(k)$ be two zero-mean stationary random processes with $A(j\omega)$ and $B(j\omega)$ as their respective DTFTs, we define the complex coherence [210] as

$$\gamma_{ab}(j\omega) = \frac{\phi_{ab}(j\omega)}{\sqrt{\phi_{aa}(\omega)\phi_{bb}(\omega)}}, \tag{6.20}$$

where

$$\phi_{ab}(j\omega) = E\left[A(j\omega)B^*(j\omega)\right] \tag{6.21}$$

is the cross-spectrum between the two signals $a(k)$ and $b(k)$, and $\phi_{aa}(\omega)$ and $\phi_{bb}(\omega)$ are their respective PSDs. The magnitude squared coherence (MSC) function, $|\gamma_{ab}(j\omega)|^2$, has this important property:

$$0 \leq |\gamma_{ab}(j\omega)|^2 \leq 1. \tag{6.22}$$

The MSC function gives an indication on the strength of the linear relationship, as a function of the frequency, between the two random variables $a(k)$ and $b(k)$.

Another important property is that if $b(k)$ is related to $a(k)$ the following way

$$b(k) = a(k) + c(k), \tag{6.23}$$

where $c(k)$ is a zero-mean stationary random process uncorrelated with $a(k)$, then the complex coherence

$$\begin{aligned} \gamma_{ab}(j\omega) &= \sqrt{\frac{\phi_{aa}(\omega)}{\phi_{bb}(\omega)}} \\ &= \gamma_{ab}(\omega) \end{aligned} \tag{6.24}$$

is always real.

6.4 Noncausal Wiener Filter

In this section we are going to derive and study the frequency-domain (noncausal) single-channel Wiener filter.

Let us define the frequency-domain error signal between the clean speech and its estimate:

$$\begin{aligned} \mathcal{E}(j\omega) &= X(j\omega) - Z(j\omega) \\ &= X(j\omega) - H(j\omega)Y(j\omega). \end{aligned} \tag{6.25}$$

The frequency-domain MSE is

$$J\left[H(j\omega)\right] = E\left[\left|\mathcal{E}(j\omega)\right|^2\right]. \tag{6.26}$$

Taking the gradient of $J\left[H(j\omega)\right]$ with respect to $H^*(j\omega)$ and equating the result to 0 lead to

$$-E\left\{Y^*(j\omega)\left[X(j\omega) - H_{\mathrm{W}}(j\omega)Y(j\omega)\right]\right\} = 0. \tag{6.27}$$

Hence

$$\phi_{yy}(\omega)H_{\mathrm{W}}(j\omega) = \phi_{xy}(j\omega). \tag{6.28}$$

But

$$\begin{aligned} \phi_{xy}(j\omega) &= E\left[X(j\omega)Y^*(j\omega)\right] \\ &= \phi_{xx}(\omega), \end{aligned} \tag{6.29}$$

therefore the optimal filter can be put into the following forms:

$$\begin{aligned} H_{\mathrm{W}}(j\omega) &= \frac{\phi_{xx}(\omega)}{\phi_{yy}(\omega)} \\ &= 1 - \frac{\phi_{vv}(\omega)}{\phi_{yy}(\omega)}. \end{aligned} \tag{6.30}$$

We see that the optimal Wiener filter is always real and positive. Therefore, from now on we will drop the imaginary unit from $H_{\mathrm{W}}(j\omega)$, i.e., $H_{\mathrm{W}}(\omega)$, to accentuate the fact that the Wiener filter is a real number.

The optimal estimate of the frequency-domain clean speech, in the MSE sense, is then

$$\begin{aligned} Z_{\mathrm{W}}(j\omega) &= H_{\mathrm{W}}(\omega)Y(j\omega) \\ &= Y(j\omega) - \frac{Y(j\omega)}{\phi_{yy}(\omega)}\phi_{vv}(\omega), \end{aligned} \tag{6.31}$$

and in the time domain:

$$z_{\mathrm{W}}(k) = y(k) - \frac{1}{2\pi}\int_{-\pi}^{\pi} \frac{\phi_{vv}(\omega)}{\phi_{yy}(\omega)} Y(j\omega)e^{j\omega k}d\omega. \tag{6.32}$$

Property 1. We have

$$\gamma_{xy}^2(\omega) + \gamma_{vy}^2(\omega) = 1, \tag{6.33}$$

where $\gamma_{xy}^2(\omega)$ is the MSC function[1] between $x(k)$ and $y(k)$, and $\gamma_{vy}^2(\omega)$ is the MSC function[2] between $v(k)$ and $y(k)$.

Proof. Indeed, we can easily check that

$$\begin{aligned} \gamma_{xy}^2(\omega) &= \frac{\phi_{xx}(\omega)}{\phi_{yy}(\omega)} \\ &= \frac{\mathrm{SNR}(\omega)}{1 + \mathrm{SNR}(\omega)}, \end{aligned} \tag{6.34}$$

and

$$\begin{aligned} \gamma_{vy}^2(\omega) &= \frac{\phi_{vv}(\omega)}{\phi_{yy}(\omega)} \\ &= \frac{1}{1 + \mathrm{SNR}(\omega)}. \end{aligned} \tag{6.35}$$

Therefore, adding (6.34) and (6.35) we find (6.33).

[1] Notice that $\gamma_{xy}(\omega)$ is always real.
[2] Notice that $\gamma_{vy}(\omega)$ is always real.

Property 1 shows that the sum of the two MSC functions is always constant and equal to 1. So if one increases the other decreases.

Property 2. We have

$$H_{\mathrm{W}}(\omega) = \gamma_{xy}^2(\omega) \tag{6.36}$$

$$= 1 - \gamma_{vy}^2(\omega). \tag{6.37}$$

These fundamental forms of the Wiener filter, although obvious, do not seem to be known in the literature. They show that they are simply related to two MSC functions. Since $0 \leq |\gamma_{ab}(j\omega)|^2 \leq 1$, then $0 \leq H_{\mathrm{W}}(\omega) \leq 1$. The Wiener filter acts like a gain function. When the level of noise is high $[\gamma_{vy}^2(\omega) \approx 1]$, then $H_{\mathrm{W}}(\omega)$ is close to 0 since there is a large amount of noise that has to be removed. When the level of noise is low $[\gamma_{vy}^2(\omega) \approx 0]$, then $H_{\mathrm{W}}(\omega)$ is close to 1 and is not going to affect much the signals since there is little noise that needs to be removed.

We deduce the narrowband noise-reduction factor and speech-distortion index

$$\xi_{\mathrm{nr}}\left[H_{\mathrm{W}}(\omega)\right] = \frac{1}{\gamma_{xy}^4(\omega)} \geq 1, \tag{6.38}$$

$$\upsilon_{\mathrm{sd}}\left[H_{\mathrm{W}}(\omega)\right] = \gamma_{vy}^4(\omega) \leq 1, \tag{6.39}$$

and the fullband noise-reduction factor and speech-distortion index

$$\xi_{\mathrm{nr}}(H_{\mathrm{W}}) = \frac{\int_{-\pi}^{\pi} \phi_{vv}(\omega)d\omega}{\int_{-\pi}^{\pi} \gamma_{xy}^4(\omega)\phi_{vv}(\omega)d\omega} \geq 1, \tag{6.40}$$

$$\upsilon_{\mathrm{sd}}(H_{\mathrm{W}}) = \frac{\int_{-\pi}^{\pi} \gamma_{vy}^4(\omega)\phi_{xx}(\omega)d\omega}{\int_{-\pi}^{\pi} \phi_{xx}(\omega)d\omega} \leq 1. \tag{6.41}$$

We see clearly how noise reduction and speech distortion depend on the two coherence functions $\gamma_{xy}(\omega)$ and $\gamma_{vy}(\omega)$ in the noncausal Wiener filter. When $\gamma_{xy}(\omega)$ increases, $\xi_{\mathrm{nr}}(H_{\mathrm{W}})$ decreases; at the same time $\gamma_{vy}(\omega)$ decreases and so does $\upsilon_{\mathrm{sd}}(H_{\mathrm{W}})$.

For any real filter $H(\omega)$, we can verify that the narrowband and fullband MSEs are

$$J\left[H(\omega)\right] = \phi_{xx}(\omega)\left[1 - H(\omega)\right]^2 + \phi_{vv}(\omega)H^2(\omega) \tag{6.42}$$

and

$$J\left(H\right) = \frac{1}{2\pi}\int_{-\pi}^{\pi} J\left[H(\omega)\right]d\omega \tag{6.43}$$

$$= \frac{1}{2\pi}\int_{-\pi}^{\pi} \phi_{xx}(\omega)\left[1 - H(\omega)\right]^2 d\omega + \frac{1}{2\pi}\int_{-\pi}^{\pi} \phi_{vv}(\omega)H^2(\omega)d\omega.$$

We define the narrowband and fullband normalized MSEs (NMSEs) as

$$\begin{aligned}\tilde{J}\left[H(\omega)\right] &= \frac{J\left[H(\omega)\right]}{\phi_{vv}(\omega)} \\ &= \mathrm{SNR}(\omega)\left[1-H(\omega)\right]^2 + H^2(\omega) \\ &= \mathrm{SNR}(\omega)\cdot \upsilon_{\mathrm{sd}}\left[H(\omega)\right] + \frac{1}{\xi_{\mathrm{nr}}\left[H(\omega)\right]}\end{aligned} \tag{6.44}$$

and

$$\begin{aligned}\tilde{J}\left(H\right) &= 2\pi \frac{J\left(H\right)}{\int_{-\pi}^{\pi}\phi_{vv}(\omega)d\omega} \\ &= \frac{\int_{-\pi}^{\pi}\phi_{xx}(\omega)\left[1-H(\omega)\right]^2 d\omega}{\int_{-\pi}^{\pi}\phi_{vv}(\omega)d\omega} + \frac{\int_{-\pi}^{\pi}\phi_{vv}(\omega)H^2(\omega)d\omega}{\int_{-\pi}^{\pi}\phi_{vv}(\omega)d\omega} \\ &= \mathrm{SNR}\cdot \upsilon_{\mathrm{sd}}(H) + \frac{1}{\xi_{\mathrm{nr}}(H)}.\end{aligned} \tag{6.45}$$

The two NMSEs have the same form. They both depend on the same variables. But the narrowband NMSE depends on the narrowband variables while the fullband NMSE depends on the fullband variables. They also have the same form as the time-domain NMSE of the causal Wiener filter [40].

The narrowband minimum NMSE is then

$$\begin{aligned}\tilde{J}\left[H_{\mathrm{W}}(\omega)\right] &= \gamma_{xy}^2(\omega) = 1-\gamma_{vy}^2(\omega) \\ &= H_{\mathrm{W}}(\omega).\end{aligned} \tag{6.46}$$

Expression (6.46) has a simple form and depends only on the coherence function between $x(k)$ and $y(k)$ [or between $v(k)$ and $y(k)$]. This minimum NMSE is also a linear function of $H_{\mathrm{W}}(\omega)$.

Property 3. With the optimal noncausal Wiener filter given in (6.30), the output fullband SNR [eq. (6.19)] is always greater than or at least equal to the input fullband SNR [eq. (6.11)], i.e., $\mathrm{oSNR}(H_{\mathrm{W}}) \geq \mathrm{SNR}$.
Proof. See [42], [125].

Property 3 is fundamental. It shows that the frequency-domain Wiener filter is able to improve the output fullband SNR of a noisy observed signal.

Very often in practice, the ensemble averages are unknown, so it is convenient to approximate the PSDs used in the Wiener filter by sample estimates [56], [218]:

$$\begin{aligned}\hat{H}_{\mathrm{W}}(\omega) &= 1-\frac{V^2(\omega)}{Y^2(\omega)} \\ &= \hat{\gamma}_{vy}^2(\omega).\end{aligned} \tag{6.47}$$

This form of the Wiener filter is the starting point of so many spectrum-based noise reduction techniques [154], [156], [218].

6.5 Parametric Wiener Filtering

Some applications may need aggressive noise reduction. Other applications on the contrary may require little speech distortion (so less aggressive noise reduction). An easy way to control the compromise between noise reduction and speech distortion is via the parametric Wiener filtering [70], [153]:

$$H_{\mathrm{G}}(\omega) = \left[1 - \gamma_{vy}^{\beta_1}(\omega)\right]^{\beta_2}, \tag{6.48}$$

where β_1 and β_2 are two positive parameters that allow the control of this compromise. For $(\beta_1, \beta_2) = (2, 1)$, we get the noncausal Wiener filter developed in the previous section. Taking $(\beta_1, \beta_2) = (2, 1/2)$, leads to

$$\begin{aligned} H_{\mathrm{P}}(\omega) &= \sqrt{1 - \gamma_{vy}^2(\omega)} \\ &= \gamma_{xy}(\omega), \end{aligned} \tag{6.49}$$

which is the power subtraction method studied in [68], [70], [153], [161], [208]. The pair $(\beta_1, \beta_2) = (1, 1)$ gives the magnitude subtraction method [22], [24], [199], [200], [228]:

$$\begin{aligned} H_{\mathrm{M}}(\omega) &= 1 - \gamma_{vy}(\omega) \\ &= 1 - \sqrt{1 - \gamma_{xy}^2(\omega)}. \end{aligned} \tag{6.50}$$

We can verify that the narrowband noise-reduction factors for the power subtraction and magnitude subtraction methods are

$$\xi_{\mathrm{nr}}\left[H_{\mathrm{P}}(\omega)\right] = \frac{1}{\gamma_{xy}^2(\omega)}, \tag{6.51}$$

$$\xi_{\mathrm{nr}}\left[H_{\mathrm{M}}(\omega)\right] = \frac{1}{\left[1 - \sqrt{1 - \gamma_{xy}^2(\omega)}\right]^2}, \tag{6.52}$$

and the corresponding narrowband speech-distortion indices are

$$\upsilon_{\mathrm{sd}}\left[H_{\mathrm{P}}(\omega)\right] = \left[1 - \sqrt{1 - \gamma_{vy}^2(\omega)}\right]^2, \tag{6.53}$$

$$\upsilon_{\mathrm{sd}}\left[H_{\mathrm{M}}(\omega)\right] = \gamma_{vy}^2(\omega). \tag{6.54}$$

We can also easily check that

$$\xi_{\mathrm{nr}}\left[H_{\mathrm{M}}(\omega)\right] \geq \xi_{\mathrm{nr}}\left[H_{\mathrm{W}}(\omega)\right] \geq \xi_{\mathrm{nr}}\left[H_{\mathrm{P}}(\omega)\right], \tag{6.55}$$

$$\upsilon_{\mathrm{sd}}\left[H_{\mathrm{P}}(\omega)\right] \leq \upsilon_{\mathrm{sd}}\left[H_{\mathrm{W}}(\omega)\right] \leq \upsilon_{\mathrm{sd}}\left[H_{\mathrm{M}}(\omega)\right]. \tag{6.56}$$

The two previous inequalities are very important from a practical point of view. They show that, among the three methods, the magnitude subtraction

is the most aggressive one as far as noise reduction is concerned, a very well-known fact in the literature [56], but at the same time it's the one that will likely distorts most the speech signal. The smoother approach is the power subtraction while the Wiener filter is between the two others in terms of speech distortion and noise reduction. Many other variants of these algorithms can be found in [100], [205].

Another straightforward way to derive parametric filters is from the narrowband NMSE, i.e. (6.44), which can be rewritten as follows:

$$[1 + \mathrm{SNR}(\omega)]\, H^2(\omega) - 2 \cdot \mathrm{SNR}(\omega) H(\omega) + \mathrm{SNR}(\omega) - \tilde{J}\,[H(\omega)] = 0, \tag{6.57}$$

which is a quadratic equation with respect to the filter $H(\omega)$. The solution is then

$$\begin{aligned} H(\omega) &= \frac{\mathrm{SNR}(\omega) \pm \sqrt{\Delta(\omega)}}{1 + \mathrm{SNR}(\omega)} \\ &= 1 - \gamma_{vy}^2(\omega) \pm \gamma_{vy}^2(\omega)\sqrt{\Delta(\omega)}, \end{aligned} \tag{6.58}$$

where

$$\begin{aligned} \Delta(\omega) &= \tilde{J}\,[H(\omega)] + \mathrm{SNR}(\omega) \left\{ \tilde{J}\,[H(\omega)] - 1 \right\} \\ &= \frac{\tilde{J}\,[H(\omega)] - 1 + \gamma_{vy}^2(\omega)}{\gamma_{vy}^2(\omega)}. \end{aligned} \tag{6.59}$$

For the filter $H(\omega)$ to be real and $H(\omega) \leq 1$ (no signal amplification), it requires that

$$1 - \gamma_{vy}^2(\omega) \leq \tilde{J}\,[H(\omega)] \leq 1. \tag{6.60}$$

Actually, $\tilde{J}\,[H(\omega)] = 1 - \gamma_{vy}^2(\omega)$ corresponds to the Wiener filter, $H_{\mathrm{W}}(\omega)$, and $\tilde{J}\,[H(\omega)] = 1$ to the unit gain filter $H(\omega) = 1$. As a result, $\Delta(\omega) \leq 1$. Now let's take

$$\beta = \pm\gamma_{vy}^2(\omega)\sqrt{\Delta(\omega)}, \tag{6.61}$$

the parametric filter is then

$$H(\omega) = 1 - \gamma_{vy}^2(\omega) + \beta, \tag{6.62}$$

where β is chosen between -1 and 1 in such a way that $0 \leq H(\omega) \leq 1$. It is easy to see that if

- $\beta = 0$, we get the Wiener filter,
- $\beta > 0$, we obtain a filter that reduces less the level of noise than the Wiener filter (so less speech distortion),

- $\beta < 0$, we have a filter that reduces more the level of noise than the Wiener filter (so more speech distortion).

We can also check that taking

$$\beta = \sqrt{1 - \gamma_{vy}^2(\omega)} - 1 + \gamma_{vy}^2(\omega) > 0 \tag{6.63}$$

leads to the power subtraction method and

$$\beta = \gamma_{vy}(\omega)\left[\gamma_{vy}(\omega) - 1\right] < 0 \tag{6.64}$$

gives the magnitude subtraction approach.

The parametric form given in (6.62) is arguably more interesting and more intuitive to use than the form shown in (6.48) since it depends on one parameter only (instead of two for the latter) and depending on its value whether it's positive or negative we know exactly if the corresponding filter will reduce less or more the level of noise as compared to the Wiener filter.

It is clear from this study that speech distortion is unavoidable in the single-channel case. Parametric Wiener filtering can help better control the compromise between noise reduction and speech distortion in many applications but this approach has obviously its limitations. In the next section we will study the multichannel case and see if there are other options for a better compromise.

6.6 Generalization to the Multichannel Case

The multichannel case consists of utilizing multiple microphones instead of just one. We expect that the spatial diversity will give more degrees of freedom for possible good solutions to the noise-reduction problem. We start by first explaining the spatial signal model.

6.6.1 Signal Model

Suppose that we have an array consisting of N sensors and a desired source signal $s(k)$ in a room. The received signals are expressed as

$$\begin{aligned} y_n(k) &= g_n * s(k) + v_n(k) \\ &= x_n(k) + v_n(k),\ n = 1, 2, \ldots, N, \end{aligned} \tag{6.65}$$

where g_n is the impulse response from the unknown source $s(k)$ to the nth microphone and $v_n(k)$ is the noise at microphone n. We assume that the signals $x_n(k)$ and $v_n(k)$ are uncorrelated and zero-mean. Without loss of generality, we consider the first microphone as the reference. Our main objective in this section is, again, noise reduction; hence we will try to recover $x_1(k)$ the best

way we can in some sense by observing not only one microphone signal but N of them. We do not attempt here to recover $s(k)$ (i.e., speech dereverberation).

In the frequency domain, (6.65) can be rewritten as

$$\begin{aligned} Y_n(j\omega) &= G_n(j\omega)S(j\omega) + V_n(j\omega) \\ &= X_n(j\omega) + V_n(j\omega),\ n = 1, 2, \ldots, N, \end{aligned} \tag{6.66}$$

where $Y_n(j\omega)$, $S(j\omega)$, $G_n(j\omega)$, $X_n(j\omega) = G_n(j\omega)S(j\omega)$, and $V_n(j\omega)$ are the DTFTs of $y_n(k)$, $s(k)$, g_n, $x_n(k)$, and $v_n(k)$, respectively. Therefore, the PSD of $y_n(k)$ is

$$\begin{aligned} \phi_{y_n y_n}(\omega) &= \phi_{x_n x_n}(\omega) + \phi_{v_n v_n}(\omega) \\ &= |G_n(j\omega)|^2 \phi_{ss}(\omega) + \phi_{v_n v_n}(\omega),\ n = 1, 2, \ldots, N. \end{aligned} \tag{6.67}$$

A linear estimate of $X_1(j\omega)$ with the N observations can be obtained as follows:

$$\begin{aligned} Z(j\omega) &= H_1^*(j\omega)Y_1(j\omega) + H_2^*(j\omega)Y_2(j\omega) + \cdots + H_N^*(j\omega)Y_N(j\omega) \\ &= \mathbf{h}^H(j\omega)\mathbf{y}(j\omega) \\ &= \mathbf{h}^H(j\omega)\left[\mathbf{x}(j\omega) + \mathbf{v}(j\omega)\right], \end{aligned} \tag{6.68}$$

where

$$\begin{aligned} \mathbf{y}(j\omega) &= \left[Y_1(j\omega)\ Y_2(j\omega) \cdots Y_N(j\omega) \right]^T, \\ \mathbf{x}(j\omega) &= S(j\omega)\left[G_1(j\omega)\ G_2(j\omega) \cdots G_N(j\omega) \right]^T \\ &= S(j\omega)\mathbf{g}(j\omega), \end{aligned}$$

$\mathbf{v}(j\omega)$ is defined in a similar way to $\mathbf{y}(j\omega)$, and

$$\mathbf{h}(j\omega) = \left[H_1(j\omega)\ H_2(j\omega) \cdots H_N(j\omega) \right]^T$$

is a vector containing the N noncausal filters to be designed. The PSD of $z(k)$ is then

$$\phi_{zz}(\omega) = \mathbf{h}^H(j\omega)\mathbf{\Phi}_{xx}(j\omega)\mathbf{h}(j\omega) + \mathbf{h}^H(j\omega)\mathbf{\Phi}_{vv}(j\omega)\mathbf{h}(j\omega), \tag{6.69}$$

where

$$\begin{aligned} \mathbf{\Phi}_{xx}(j\omega) &= E\left[\mathbf{x}(j\omega)\mathbf{x}^H(j\omega)\right] \\ &= \phi_{ss}(\omega)\mathbf{g}(j\omega)\mathbf{g}^H(j\omega), \end{aligned} \tag{6.70}$$

$$\mathbf{\Phi}_{vv}(j\omega) = E\left[\mathbf{v}(j\omega)\mathbf{v}^H(j\omega)\right], \tag{6.71}$$

are the PSD matrices of the signals $x_n(k)$ and $v_n(k)$, respectively. Notice that the rank of the matrix $\mathbf{\Phi}_{xx}(j\omega)$ is always equal to 1.

In the rest of this section, we will study the design of the filter vector $\mathbf{h}(j\omega)$ but we first give some useful definitions.

6.6.2 Definitions

In this subsection we briefly generalize some definitions of Section 6.3 to the multichannel case. Since the first microphone is chosen as the reference, all definitions will be given with respect to that reference.

The input narrowband and fullband SNRs are

$$\mathrm{SNR}(\omega) = \frac{\phi_{x_1x_1}(\omega)}{\phi_{v_1v_1}(\omega)}, \tag{6.72}$$

$$\mathrm{SNR} = \frac{\int_{-\pi}^{\pi} \phi_{x_1x_1}(\omega)d\omega}{\int_{-\pi}^{\pi} \phi_{v_1v_1}(\omega)d\omega}. \tag{6.73}$$

We define the narrowband and fullband multichannel noise-reduction factors as

$$\xi_{\mathrm{nr}}\left[\mathbf{h}(j\omega)\right] = \frac{\phi_{v_1v_1}(\omega)}{\mathbf{h}^H(j\omega)\mathbf{\Phi}_{vv}(j\omega)\mathbf{h}(j\omega)}, \tag{6.74}$$

$$\begin{aligned}\xi_{\mathrm{nr}}(\mathbf{h}) &= \frac{\int_{-\pi}^{\pi} \phi_{v_1v_1}(\omega)d\omega}{\int_{-\pi}^{\pi} \mathbf{h}^H(j\omega)\mathbf{\Phi}_{vv}(j\omega)\mathbf{h}(j\omega)d\omega} \\ &= \frac{\int_{-\pi}^{\pi} \phi_{v_1v_1}(\omega)d\omega}{\int_{-\pi}^{\pi} \xi_{\mathrm{nr}}^{-1}\left[\mathbf{h}(j\omega)\right]\phi_{v_1v_1}(\omega)d\omega}.\end{aligned} \tag{6.75}$$

Contrary to the narrowband single-channel noise-reduction factor, the multichannel version depends on the PSD of the noise.

We define the narrowband and fullband multichannel speech-distortion indices as

$$\begin{aligned}\upsilon_{\mathrm{sd}}\left[\mathbf{h}(j\omega)\right] &= \frac{E\left[\left|X_1(j\omega) - \mathbf{h}^H(j\omega)\mathbf{x}(j\omega)\right|^2\right]}{\phi_{x_1x_1}(\omega)} \\ &= \frac{\left[\mathbf{u} - \mathbf{h}(j\omega)\right]^H \mathbf{\Phi}_{xx}(j\omega)\left[\mathbf{u} - \mathbf{h}(j\omega)\right]}{\phi_{x_1x_1}(\omega)},\end{aligned} \tag{6.76}$$

$$\begin{aligned}\upsilon_{\mathrm{sd}}(\mathbf{h}) &= \frac{\int_{-\pi}^{\pi} E\left[\left|X_1(j\omega) - \mathbf{h}^H(j\omega)\mathbf{x}(j\omega)\right|^2\right] d\omega}{\int_{-\pi}^{\pi} \phi_{x_1x_1}(\omega)d\omega} \\ &= \frac{\int_{-\pi}^{\pi} \left[\mathbf{u} - \mathbf{h}(j\omega)\right]^H \mathbf{\Phi}_{xx}(j\omega)\left[\mathbf{u} - \mathbf{h}(j\omega)\right] d\omega}{\int_{-\pi}^{\pi} \phi_{x_1x_1}(\omega)d\omega} \\ &= \frac{\int_{-\pi}^{\pi} \upsilon_{\mathrm{sd}}\left[\mathbf{h}(j\omega)\right]\phi_{x_1x_1}(\omega)d\omega}{\int_{-\pi}^{\pi} \phi_{x_1x_1}(\omega)d\omega}.\end{aligned} \tag{6.77}$$

The narrowband multichannel speech-distortion index depends on the PSD of the speech and on the filter vector contrary to its single-channel counterpart which depends only on the filter.

The output narrowband and fullband SNRs are

$$\text{oSNR}\left[\mathbf{h}(j\omega)\right] = \frac{\mathbf{h}^H(j\omega)\mathbf{\Phi}_{xx}(j\omega)\mathbf{h}(j\omega)}{\mathbf{h}^H(j\omega)\mathbf{\Phi}_{vv}(j\omega)\mathbf{h}(j\omega)}, \tag{6.78}$$

$$\text{oSNR}(\mathbf{h}) = \frac{\int_{-\pi}^{\pi}\mathbf{h}^H(j\omega)\mathbf{\Phi}_{xx}(j\omega)\mathbf{h}(j\omega)d\omega}{\int_{-\pi}^{\pi}\mathbf{h}^H(j\omega)\mathbf{\Phi}_{vv}(j\omega)\mathbf{h}(j\omega)d\omega}. \tag{6.79}$$

It is interesting to see that now the output narrowband SNR, which depends on the filter vector $\mathbf{h}(j\omega)$ and PSDs of the speech and noise, is not equal to the input narrowband SNR. This is a major difference from the single-channel case. As a consequence, the spatial diversity can help improve the output SNR. Also, we can see from (6.78)–(6.79) that if

- $\text{oSNR}\left[\mathbf{h}(j\omega)\right] = 1,\ \forall\omega$, then $\text{oSNR}(\mathbf{h}) = 1$,
- $\text{oSNR}\left[\mathbf{h}(j\omega)\right] < 1,\ \forall\omega$, then $\text{oSNR}(\mathbf{h}) < 1$,
- $\text{oSNR}\left[\mathbf{h}(j\omega)\right] > 1,\ \forall\omega$, then $\text{oSNR}(\mathbf{h}) > 1$.

6.6.3 Multichannel Wiener Filter

To derive the Wiener filter, we first need to write the error signal

$$\begin{aligned}\mathcal{E}(j\omega) &= X_1(j\omega) - Z(j\omega)\\ &= X_1(j\omega) - \mathbf{h}^H(j\omega)\mathbf{y}(j\omega)\\ &= \left[\mathbf{u} - \mathbf{h}(j\omega)\right]^H\mathbf{x}(j\omega) - \mathbf{h}^H(j\omega)\mathbf{v}(j\omega),\end{aligned} \tag{6.80}$$

where

$$\mathbf{u} = \begin{bmatrix}1\ 0\cdots 0\ 0\end{bmatrix}^T \tag{6.81}$$

is a vector of length N. The corresponding MSE is

$$J\left[\mathbf{h}(j\omega)\right] = E\left[\left|\mathcal{E}(j\omega)\right|^2\right]. \tag{6.82}$$

The minimization of (6.82) with respect to $\mathbf{h}(j\omega)$ leads to

$$\mathbf{\Phi}_{yy}(j\omega)\mathbf{h}_{\text{W}}(j\omega) = \mathbf{\Phi}_{yx}(j\omega)\mathbf{u}, \tag{6.83}$$

where

$$\begin{aligned}\mathbf{\Phi}_{yy}(j\omega) &= E\left[\mathbf{y}(j\omega)\mathbf{y}^H(j\omega)\right]\\ &= \phi_{ss}(\omega)\mathbf{g}(j\omega)\mathbf{g}^H(j\omega) + \mathbf{\Phi}_{vv}(j\omega)\end{aligned} \tag{6.84}$$

is the PSD matrix of the signals $y_n(k)$ and

$$\begin{aligned}\mathbf{\Phi}_{yx}(j\omega) &= E\left[\mathbf{y}(j\omega)\mathbf{x}^H(j\omega)\right]\\ &= \mathbf{\Phi}_{xx}(j\omega)\end{aligned} \tag{6.85}$$

is the cross-spectral matrix between the signals $y_n(k)$ and $x_n(k)$.

Therefore, the optimal filter can be put into the following forms:

$$\begin{aligned}\mathbf{h}_{\mathrm{W}}(j\omega) &= \mathbf{\Phi}_{yy}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\mathbf{u} \\ &= \left[\mathbf{I}_{N\times N} - \mathbf{\Phi}_{yy}^{-1}(j\omega)\mathbf{\Phi}_{vv}(j\omega)\right]\mathbf{u}. \end{aligned} \tag{6.86}$$

We can make two important observations. The first one is that the multichannel Wiener filter is complex contrary to its single-channel counterpart which is always real. Obviously, the phase has a role to play in the multichannel case since the spatial information is involved and the desired signal does not necessarily arrive with the same phase at the different microphones. The second observation is that a necessary condition for the matrix $\mathbf{\Phi}_{yy}(j\omega)$ to be full rank is that the matrix $\mathbf{\Phi}_{vv}(j\omega)$ is also full rank. In other words, for the multichannel Wiener filter to be unique the noise should not be completely coherent at the microphones.

Determining the inverse of $\mathbf{\Phi}_{yy}(j\omega)$ from (6.84) with the Woodbury's identity

$$\left[\mathbf{\Phi}_{vv}(j\omega) + \phi_{ss}(\omega)\mathbf{g}(j\omega)\mathbf{g}^H(j\omega)\right]^{-1} = \tag{6.87}$$

$$\mathbf{\Phi}_{vv}^{-1}(j\omega) - \frac{\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{g}(j\omega)\mathbf{g}^H(j\omega)\mathbf{\Phi}_{vv}^{-1}(j\omega)}{\phi_{ss}^{-1}(\omega) + \mathbf{g}^H(j\omega)\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{g}(j\omega)}$$

$$= \mathbf{\Phi}_{vv}^{-1}(j\omega) - \frac{\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\mathbf{\Phi}_{vv}^{-1}(j\omega)}{1 + \mathrm{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]} \tag{6.88}$$

and substituting the result into (6.86), leads to other interesting formulations of the Wiener filter:

$$\mathbf{h}_{\mathrm{W}}(j\omega) = \frac{\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)}{1 + \mathrm{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]}\mathbf{u} \tag{6.89}$$

$$= \frac{\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{yy}(j\omega) - \mathbf{I}_{N\times N}}{1 - N + \mathrm{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{yy}(j\omega)\right]}\mathbf{u} \tag{6.90}$$

$$= \left\{\mathbf{I}_{N\times N} - \frac{\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)}{1 + \mathrm{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]}\right\} \times \mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\mathbf{u}. \tag{6.91}$$

We deduce the narrowband noise-reduction factor and speech distortion index for the multichannel Wiener filter:

$$\xi_{\mathrm{nr}}\left[\mathbf{h}_{\mathrm{W}}(j\omega)\right] = \frac{\left\{1 + \mathrm{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]\right\}^2}{\mathrm{SNR}(\omega)\mathrm{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]}, \tag{6.92}$$

$$\upsilon_{\mathrm{sd}}\left[\mathbf{h}_{\mathrm{W}}(j\omega)\right] = \frac{1}{\left\{1 + \mathrm{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]\right\}^2}. \tag{6.93}$$

For any complex filter vector $\mathbf{h}(j\omega)$, we can verify that the narrowband and fullband MSEs are

$$J\left[\mathbf{h}(j\omega)\right] = \left[\mathbf{u} - \mathbf{h}(j\omega)\right]^H \mathbf{\Phi}_{xx}(j\omega)\left[\mathbf{u} - \mathbf{h}(j\omega)\right] + \mathbf{h}^H(j\omega)\mathbf{\Phi}_{vv}(j\omega)\mathbf{h}(j\omega) \tag{6.94}$$

and

$$J\left(\mathbf{h}\right) = \frac{1}{2\pi}\int_{-\pi}^{\pi}\left[\mathbf{u} - \mathbf{h}(j\omega)\right]^H \mathbf{\Phi}_{xx}(j\omega)\left[\mathbf{u} - \mathbf{h}(j\omega)\right]d\omega + \frac{1}{2\pi}\int_{-\pi}^{\pi}\mathbf{h}^H(j\omega)\mathbf{\Phi}_{vv}(j\omega)\mathbf{h}(j\omega)d\omega. \tag{6.95}$$

Therefore, the narrowband and fullband NMSEs are

$$\tilde{J}\left[\mathbf{h}(j\omega)\right] = \mathrm{SNR}(\omega)\cdot\upsilon_{\mathrm{sd}}\left[\mathbf{h}(j\omega)\right] + \frac{1}{\xi_{\mathrm{nr}}\left[\mathbf{h}(j\omega)\right]} \tag{6.96}$$

and

$$\tilde{J}\left(\mathbf{h}\right) = \mathrm{SNR}\cdot\upsilon_{\mathrm{sd}}\left(\mathbf{h}\right) + \frac{1}{\xi_{\mathrm{nr}}\left(\mathbf{h}\right)}. \tag{6.97}$$

Using (6.92) and (6.93), we deduce the narrowband NMSE for the Wiener filter (minimum NMSE):

$$\tilde{J}\left[\mathbf{h}_{\mathrm{W}}(j\omega)\right] = \frac{\mathrm{SNR}(\omega)}{1 + \mathrm{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]}. \tag{6.98}$$

Property 4. With the noncausal multichannel Wiener filter given in (6.86), the output narrowband SNR [eq. (6.78)] is always greater than or at least equal to the input narrowband SNR [eq. (6.72)], i.e., $\mathrm{oSNR}\left[\mathbf{h}_{\mathrm{W}}(j\omega)\right] \geq \mathrm{SNR}(\omega)$.

This is a fundamental difference from the single-channel case, where the input narrowband SNR is always identical to the output narrowband SNR.

Proof. We can use the same proofs given in [42], [62], [125] to show this property.

A more explicit form of the output narrowband SNR with the Wiener filter is

$$\mathrm{oSNR}\left[\mathbf{h}_{\mathrm{W}}(j\omega)\right] = \mathrm{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]. \tag{6.99}$$

Using (6.92), (6.93), and (6.99) we can verify the relation

$$\mathrm{SNR}(\omega)\cdot\mathrm{oSNR}\left[\mathbf{h}_{\mathrm{W}}(j\omega)\right]\cdot\xi_{\mathrm{nr}}\left[\mathbf{h}_{\mathrm{W}}(j\omega)\right]\cdot\upsilon_{\mathrm{sd}}\left[\mathbf{h}_{\mathrm{W}}(j\omega)\right] = 1, \tag{6.100}$$

showing the importance of the four involved measures.

Property 5. With the noncausal multichannel Wiener filter given in (6.86), the output fullband SNR [eq. (6.79)] is always greater than or at least equal to the input fullband SNR [eq. (6.73)], i.e., $\text{oSNR}\left(\mathbf{h}_{\text{W}}\right) \geq \text{SNR}$.

Proof. We can use the same proofs given in [42], [62], [125] to show this property.

To summarize, the noncausal multichannel Wiener filter has the potential to improve both the output narrowband and fullband SNRs, while the noncausal single-channel Wiener filter has the potential to improve only the output fullband SNR.

The parametric Wiener filtering developed in Section 6.5 can be generalized to the multichannel case by using the same ideas proposed in [59], [60]. The derived parametric multichannel Wiener filter is then

$$\mathbf{h}_{\text{G}}(j\omega) = \left[\mathbf{\Phi}_{xx}(j\omega) + \beta_0 \mathbf{\Phi}_{vv}(j\omega)\right]^{-1} \mathbf{\Phi}_{xx}(j\omega)\mathbf{u}, \tag{6.101}$$

where the real $\beta_0 \geq 0$ is the tradeoff parameter between noise reduction and speech distortion. If $\beta_0 > 1$, the residual noise level is reduced at the expense of increased speech distortion. On the contrary, if $\beta_0 < 1$, speech distortion is reduced at the expense of decreased noise reduction [59], [69], [209]. A more sophisticated approach can be developed by replacing β_0 with a perceptual filter [113].

In the single-channel case ($N = 1$), (6.101) reduces to

$$\begin{aligned} H_{\text{G}}(\omega) &= \frac{1 - \gamma_{vy}^2(\omega)}{1 - (1 - \beta_0)\gamma_{vy}^2(\omega)} \\ &\approx 1 - \gamma_{vy}^2(\omega) + \gamma_{vy}^2(\omega)(1 - \beta_0)\left[1 - \gamma_{vy}^2(\omega)\right] \\ &\approx 1 - \gamma_{vy}^2(\omega) + \beta, \end{aligned} \tag{6.102}$$

which works in a similar way to (6.62).

6.6.4 Spatial Maximum SNR Filter

The minimization of the MSE criterion [eq. (6.82)] leads to the Wiener filter. Another criterion, instead, is the output narrowband SNR, $\text{oSNR}\left[\mathbf{h}(j\omega)\right]$, defined in (6.78) that we can maximize, since this measure is the most relevant one as far as noise reduction is concerned.

Maximizing $\text{oSNR}\left[\mathbf{h}(j\omega)\right]$ is equivalent to solving the generalized eigenvalue problem

$$\mathbf{\Phi}_{xx}(j\omega)\mathbf{h}(j\omega) = \lambda(\omega)\mathbf{\Phi}_{vv}(j\omega)\mathbf{h}(j\omega). \tag{6.103}$$

The optimal solution to this well-known problem is $\mathbf{h}_{\max}(j\omega)$, the eigenvector corresponding to the maximum eigenvalue, $\lambda_{\max}(\omega)$, of the matrix $\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)$. In this case we have

$$\text{oSNR}\left[\mathbf{h}_{\max}(j\omega)\right] = \lambda_{\max}(\omega). \tag{6.104}$$

It is clear that $c\mathbf{h}_{\max}(j\omega)$, for any scalar c, is also a solution of (6.103). Usually we choose the eigenvector that has the unit norm, i.e., $\mathbf{h}_{\max}^H(j\omega)\mathbf{h}_{\max}(j\omega) = 1$. This is the convention we adopt here.

We already know that the rank of the matrix $\mathbf{\Phi}_{xx}(j\omega)$ is equal to 1. Therefore, the matrix $\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)$ has only one nonzero eigenvalue corresponding to $\lambda_{\max}(\omega)$. Furthermore it is easy to verify, using (6.89), that

$$\mathbf{\Phi}_{xx}(j\omega)\mathbf{h}_{\mathrm{W}}(j\omega) = \mathrm{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]\mathbf{\Phi}_{vv}(j\omega)\mathbf{h}_{\mathrm{W}}(j\omega). \tag{6.105}$$

Therefore, the Wiener filter, $\mathbf{h}_{\mathrm{W}}(j\omega)$, is also a solution to our problem. As a result

$$\mathbf{h}_{\max}(j\omega) = \frac{\mathbf{h}_{\mathrm{W}}(j\omega)}{\sqrt{\mathbf{h}_{\mathrm{W}}^H(j\omega)\mathbf{h}_{\mathrm{W}}(j\omega)}}, \tag{6.106}$$

$$\lambda_{\max}(\omega) = \mathrm{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]. \tag{6.107}$$

Surprisingly, the maximum SNR filter does not exist in the noncausal single-channel case but does exist in the time domain and is different, in general, from the Wiener filter.

We can conclude that minimizing the MSE criterion is equivalent to maximizing the output SNR at frequency ω (locally), up to a scaling factor. However, the two approaches are very different from a fullband point of view or in practice. Remember, these optimizations (MSE and max SNR) are done for each frequency independently of the others. As a result, the scaling factors (norms) of the Wiener vectors at the different frequencies are not constant. While locally the two filters (Wiener and maximum SNR) give the same output SNR, globally they do not perform the same for noise reduction. Indeed, it is easy to check that

$$\mathrm{oSNR}\left[\mathbf{h}_{\mathrm{W}}(j\omega)\right] = \mathrm{oSNR}\left[\mathbf{h}_{\max}(j\omega)\right] \tag{6.108}$$

but

$$\mathrm{oSNR}\left(\mathbf{h}_{\mathrm{W}}\right) \neq \mathrm{oSNR}\left(\mathbf{h}_{\max}\right) \tag{6.109}$$

unless, of course, we normalize the vector $\mathbf{h}_{\mathrm{W}}(j\omega)$ in such a way that its norm is 1.

The two filters distort the speech signal since

$$\upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\mathrm{W}}\right) \neq 0, \tag{6.110}$$

$$\upsilon_{\mathrm{sd}}\left(\mathbf{h}_{\max}\right) \neq 0. \tag{6.111}$$

Contrary to the time-domain methods, the frequency-domain algorithms are affected by the scaling factor. This problem is somewhat similar to the convolutive blind source separation (BSS) in the frequency domain where separation can be obtained up to a scaling factor at each frequency [159], [182]. It is then essential to find appropriate solutions to this problem, which will be discussed in the next two sections.

6.6.5 Minimum Variance Distortionless Response Filter

The minimum variance distortionless response (MVDR) filter [35], [148], [149], [216] results from the optimization of a criterion with a constraint which tries to minimize the level of noise of the noisy signals without distorting the desired signal. From the error signal given in (6.80), it's clear that the constraint should be taken in such a way that

$$[\mathbf{u} - \mathbf{h}(j\omega)]^H \mathbf{x}(j\omega) = 0. \tag{6.112}$$

Replacing $\mathbf{x}(j\omega) = S(j\omega)\mathbf{g}(j\omega)$ in the previous equation gives

$$\mathbf{h}^H(j\omega)\mathbf{g}(j\omega) = G_1(j\omega). \tag{6.113}$$

The MVDR problem for choosing the weights is thus written as

$$\min_{\mathbf{h}(j\omega)} \mathbf{h}^H(j\omega)\mathbf{\Phi}_{vv}(j\omega)\mathbf{h}(j\omega) \quad \text{subject to} \quad \mathbf{h}^H(j\omega)\mathbf{g}(j\omega) = G_1(j\omega). \tag{6.114}$$

Using Lagrange multipliers, we easily find the MVDR filter:

$$\mathbf{h}_{\text{MVDR}}(j\omega) = G_1^*(j\omega)\frac{\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{g}(j\omega)}{\mathbf{g}^H(j\omega)\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{g}(j\omega)}, \tag{6.115}$$

which can be put in other more interesting forms:

$$\begin{aligned}\mathbf{h}_{\text{MVDR}}(j\omega) &= \frac{\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)}{\text{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]}\mathbf{u} \\ &= \frac{\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{yy}(j\omega) - \mathbf{I}_{N\times N}}{\text{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{yy}(j\omega)\right] - N}\mathbf{u}.\end{aligned} \tag{6.116}$$

It can be easily verified that

$$\mathbf{h}_{\text{W}}(j\omega) = c(\omega)\mathbf{h}_{\text{MVDR}}(j\omega), \tag{6.117}$$

$$\mathbf{h}_{\max}(j\omega) = \frac{\mathbf{h}_{\text{MVDR}}(j\omega)}{\sqrt{\mathbf{h}_{\text{MVDR}}^H(j\omega)\mathbf{h}_{\text{MVDR}}(j\omega)}}, \tag{6.118}$$

where

$$c(\omega) = \frac{\text{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]}{1 + \text{tr}\left[\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{\Phi}_{xx}(j\omega)\right]}. \tag{6.119}$$

Again, the three fundamental filters $\mathbf{h}_{\text{W}}(j\omega)$, $\mathbf{h}_{\max}(j\omega)$, and $\mathbf{h}_{\text{MVDR}}(j\omega)$ are equivalent up to a scaling factor [81]; thus

$$\text{oSNR}\left[\mathbf{h}_{\text{MVDR}}(j\omega)\right] = \text{oSNR}\left[\mathbf{h}_{\text{W}}(j\omega)\right] = \text{oSNR}\left[\mathbf{h}_{\max}(j\omega)\right]. \tag{6.120}$$

But this time

$$\upsilon_{\text{sd}}\left[\mathbf{h}_{\text{MVDR}}(j\omega)\right] = \upsilon_{\text{sd}}\left(\mathbf{h}_{\text{MVDR}}\right) = 0. \tag{6.121}$$

This makes the scaling factor of the MVDR filter optimal in the sense that it does not distort the speech signal.

We can also check that the narrowband noise-reduction factor is

$$\xi_{\text{nr}}\left[\mathbf{h}_{\text{MVDR}}(j\omega)\right] = \frac{\text{oSNR}\left[\mathbf{h}_{\text{MVDR}}(j\omega)\right]}{\text{SNR}(\omega)}. \tag{6.122}$$

The form of the MVDR filter given in (6.115) is equivalent to the transfer function generalized sidelobe canceler (TF-GSC) proposed by Gannot et al. [79], [80]. The major inconvenience of this algorithm is the blind estimation of the vector $G_1^{-1}(j\omega)\mathbf{g}(j\omega)$ (transfer functions) which is not easy to do in practice without the insights given in (6.116). The same authors try to take advantage of the nonstationarity of the speech for its estimation but this estimator may not be very robust or accurate.

The form of the MVDR filter shown in (6.116) is not exploited in the literature which is really surprising since it's, and by far, much more practical than (6.115) and it does not require the estimation of the channel impulse response. This is a relief, since as we all know blind estimation is always a very difficult problem.

To summarize, the MVDR filter as proposed in (6.116) solves the scaling factor problem encountered in the Wiener and maximum SNR filters and does not require the knowledge of the acoustic channel like the GSC implementation does [81].

6.6.6 Distortionless Multichannel Wiener Filter

In this subsection we derive a distortionless multichannel Wiener filter in two steps: the first step finds the constraint with another noncausal filter while the second step finds an optimal estimator of this noncausal filter.

Assume that we can find a noncausal filter, $W_n(j\omega)$, such that

$$X_n(j\omega) = W_n(j\omega)X_1(j\omega),\ n = 2, \ldots, N. \tag{6.123}$$

We will show later how to find this optimal filter.

Substituting (6.123) into (6.80), we get

$$\mathcal{E}(j\omega) = \left[1 - \mathbf{h}^H(j\omega)\mathbf{w}(j\omega)\right] X_1(j\omega) - \mathbf{h}^H(j\omega)\mathbf{v}(j\omega), \tag{6.124}$$

where

$$\mathbf{w}(j\omega) = \left[1\ W_2(j\omega)\ \cdots\ W_N(j\omega)\right]^T.$$

In order not to distort the desired signal, we should solve the following optimization problem:

$$\min_{\mathbf{h}(j\omega)} \mathbf{h}^H(j\omega)\mathbf{\Phi}_{vv}(j\omega)\mathbf{h}(j\omega) \quad \text{subject to} \quad \mathbf{h}^H(j\omega)\mathbf{w}(j\omega) = 1, \tag{6.125}$$

from which we deduce the optimal distortionless Wiener (DW) filter:

$$\mathbf{h}_{\text{DW}}(j\omega) = \frac{\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{w}(j\omega)}{\mathbf{w}^H(j\omega)\mathbf{\Phi}_{vv}^{-1}(j\omega)\mathbf{w}(j\omega)}. \tag{6.126}$$

The second step consists of finding the noncausal filter $W_n(j\omega)$. An optimal estimator, in the Wiener sense, can be obtained by minimizing the following cost function

$$J\left[W_n(j\omega)\right] = E\left[\left|X_n(j\omega) - W_n(j\omega)X_1(j\omega)\right|^2\right]. \tag{6.127}$$

We easily find the optimal Wiener filter:

$$W_{n,\text{W}}(j\omega) = \frac{\phi_{x_1x_1}(\omega)}{\phi_{x_nx_1}(j\omega)}, \ n = 2, \ldots, N, \tag{6.128}$$

where

$$\phi_{x_nx_1}(j\omega) = E\left[X_n(j\omega)X_1^*(j\omega)\right] \tag{6.129}$$

is the cross-spectrum between the signals $x_n(k)$ and $x_1(k)$.

Also, we can write the Wiener filter, $W_{n,\text{W}}(j\omega)$, in terms of the acoustic channels:

$$W_{n,\text{W}}(j\omega) = \frac{G_1(j\omega)}{G_n(j\omega)}, \ n = 2, \ldots, N. \tag{6.130}$$

Using this form in (6.126), we obtain

$$\mathbf{h}_{\text{DW}}(j\omega) = \mathbf{h}_{\text{MVDR}}(j\omega). \tag{6.131}$$

Thus, the DW and MVDR filters are identical. Like the MVDR filter, the DW filter (which is a two step approach) solves the scaling factor problem. Another advantage of this method compared to the TF-GSC is that it does not require the knowledge of the transfer functions explicitly. A time-domain version of this algorithm can be found in [21], [44]. (See also Chapters 4 and 5.)

6.7 Conclusions

This chapter was dedicated to the noncausal (frequency-domain) optimal filter for both the single- and multi-channel cases. We have given some important definitions and emphasized the differences between the narrowband and full-band variables. This distinction gives more insights into the understanding of

the algorithms in the frequency domain. We have also seen that while in all the single-channel algorithms there is always a compromise between noise reduction and speech distortion, for the multichannel filters when well designed it's possible to have a good amount of noise reduction without distorting the desired signal. For example, an interesting form of the MVDR filter was presented that can be implemented as easily as the popular magnitude spectral subtraction method but with no speech distortion.

7

Microphone Arrays from a MIMO Perspective

7.1 Introduction

As seen throughout the text, the major functionality of a microphone-array system is to reduce noise, thereby enhancing a desired information-bearing speech signal. The term noise, in general, refers to any unwanted signal that interferes with measurement, processing, and communication of the desired speech signal. This broad-sense definition of noise, however, is too encompassing as it masks many important technical aspects of the real problem. To enable better modeling and removal of the effects of noise in the context of microphone array processing, it is advantageous to break the general definition into the following three subcategories: additive noise originating from various ambient sound sources, interfering signals from concurrent competing sources, and reverberation caused by multipath propagation introduced by an enclosure. We have seen from the previous chapters that the use of a microphone array together with proper beamforming techniques can reduce the effect of additive noise. This chapter continues to explore beamforming techniques, with a focus on interference suppression and speech dereverberation. Different from the traditional way of treating beamforming as purely spatial filtering, this chapter studies the problem from a more physically meaningful multiple-input multiple-output (MIMO) signal processing perspective. A general framework based on the MIMO channel impulse responses will be developed. Under this framework, we study different algorithms including their underlying principles and intrinsic connections. We also analyze the bounds for the beamforming filter length, which govern the performance of beamforming in terms of speech dereverberation and interference suppression. In addition, we discuss, from the channel condition point of view, what are the necessary conditions for different beamforming algorithms to work.

This chapter is organized as follows. Section 7.2 presents the four signal models (depending on the inputs and outputs) and the problem description. In Section 7.3, the two-element microphone array is studied. Section 7.4 studies the general case of a microphone array with any number of elements. Sec-

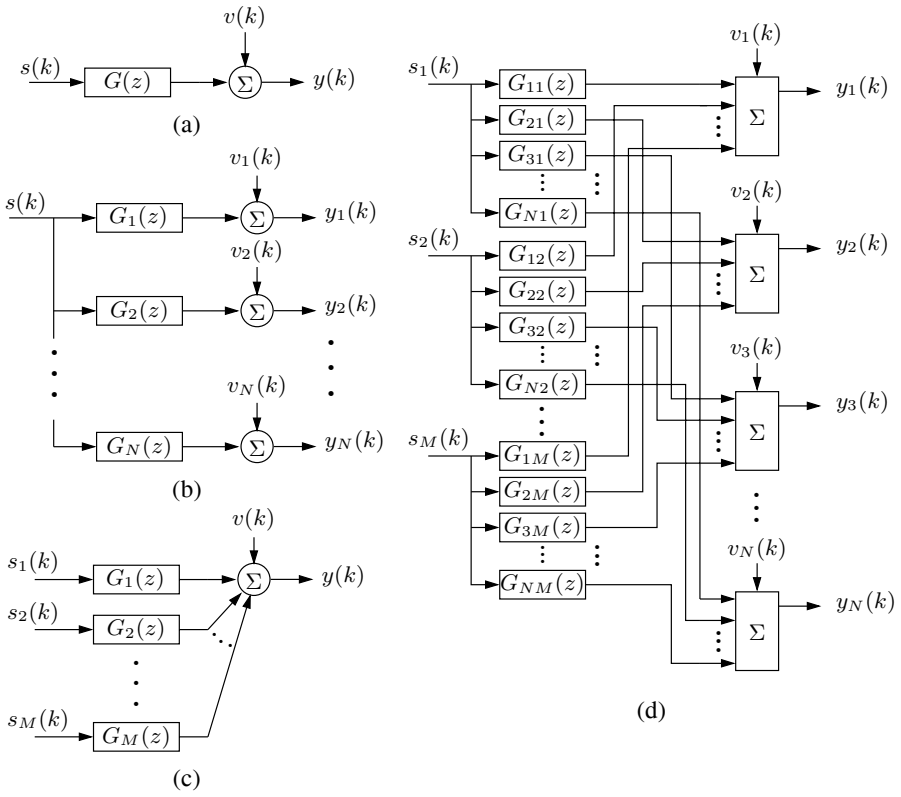


Fig. 7.1. Illustration of four distinct types of systems. (a) A single-input single-output (SISO) system. (b) A single-input multiple-output (SIMO) system. (c) A multiple-input single-output (MISO) system. (d) A multiple-input multiple-output (MIMO) system.

tion 7.5 gives some experimental results. Finally, some conclusions will be provided in Section 7.6.

7.2 Signal Models and Problem Description

Throughout the text, we have presented several signal models to describe a microphone-array system in different wave-propagation situations. To enable a better understanding of how beamforming can be formulated to suppress interference and dereverberate speech, it is advantageous to divide the signal models into four basic classes according to the number of inputs and outputs. Such classification is now well accepted and is the basis of many interesting studies in different areas of control and signal processing.

7.2.1 SISO Model

The first class is the single-input single-output (SISO) system, as shown in Fig. 7.1(a). The output signal is given by

$$y(k) = g * s(k) + v(k), \tag{7.1}$$

where g is the channel impulse response, $s(k)$ is the source signal at time k, and $v(k)$ is the additive noise at the output. Here we assume that the system is linear and shift-invariant. The channel impulse response is delineated usually with an FIR filter rather than an IIR filter. In vector/matrix form, the SISO signal model (7.1) is written as

$$y(k) = \mathbf{g}^T \mathbf{s}(k) + v(k), \tag{7.2}$$

where

$$\begin{aligned} \mathbf{g} &= \left[g_0 \; g_1 \; \cdots \; g_{L_g-1} \right]^T, \\ \mathbf{s}(k) &= \left[s(k) \; s(k-1) \; \cdots \; s(k-L_g+1) \right]^T, \end{aligned}$$

and L_g is the channel length.

Using the z-transform, the SISO signal model (7.2) is described as follows

$$Y(z) = G(z)S(z) + V(z), \tag{7.3}$$

where $Y(z)$, $S(z)$, and $V(z)$ are the z-transforms of $y(k)$, $s(k)$, and $v(k)$, respectively, and $G(z) = \sum_{l=0}^{L_g-1} g_l z^{-l}$.

The SISO model is simple and is probably the most widely used and studied model in communications, signal processing, and control.

7.2.2 SIMO Model

The diagram of a single-input multiple-output (SIMO) system is illustrated by Fig. 7.1(b), in which there are N outputs from the same source as input and the nth output is expressed as

$$y_n(k) = \mathbf{g}_n^T \mathbf{s}(k) + v_n(k), \;\; n = 1, 2, \ldots, N, \tag{7.4}$$

where $\mathbf{g}_n$ and $v_n(k)$ are defined in a similar way to those in (7.2), and L_g is the length of the longest channel impulse response in this SIMO system. A more comprehensive expression of the SIMO model is given by

$$\mathbf{y}_{\mathrm{a}}(k) = \mathbf{G}\mathbf{s}(k) + \mathbf{v}_{\mathrm{a}}(k), \tag{7.5}$$

where

$$\mathbf{y}_{\mathrm{a}}(k) = \left[\, y_1(k)\ y_2(k) \cdots y_N(k) \,\right]^T,$$
$$\mathbf{G} = \begin{bmatrix} g_{1,0} & g_{1,1} & \cdots & g_{1,L_g-1} \\ g_{2,0} & g_{2,1} & \cdots & g_{2,L_g-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{N,0} & g_{N,1} & \cdots & g_{N,L_g-1} \end{bmatrix}_{N \times L_g},$$
$$\mathbf{v}_{\mathrm{a}}(k) = \left[\, v_1(k)\ v_2(k) \cdots v_N(k) \,\right]^T.$$

The SIMO model (7.5) is described in the z-transform domain as

$$\mathbf{y}_{\mathrm{a}}(z) = \mathbf{g}(z)S(z) + \mathbf{v}_{\mathrm{a}}(z), \tag{7.6}$$

where

$$\mathbf{y}_{\mathrm{a}}(z) = \left[\, Y_1(z)\ Y_2(z) \cdots Y_N(z) \,\right]^T,$$
$$\mathbf{g}(z) = \left[\, G_1(z)\ G_2(z) \cdots G_N(z) \,\right]^T,$$
$$G_n(z) = \sum_{l=0}^{L_g-1} g_{n,l} z^{-l},\ n = 1, 2, \ldots, N,$$
$$\mathbf{v}_{\mathrm{a}}(z) = \left[\, V_1(z)\ V_2(z) \cdots V_N(z) \,\right]^T.$$

7.2.3 MISO Model

In the third type of systems as drawn in Fig. 7.1(c), we suppose that there are M sources but only one output whose signal is then expressed as

$$\begin{aligned} y(k) &= \sum_{m=1}^{M} \mathbf{g}_m^T \mathbf{s}_m(k) + v(k), \\ &= \mathbf{g}^T \mathbf{s}_{ML_g}(k) + v(k), \end{aligned} \tag{7.7}$$

where

$$\mathbf{g} = \left[\, \mathbf{g}_1^T\ \mathbf{g}_2^T \cdots \mathbf{g}_M^T \,\right]^T,$$
$$\mathbf{g}_m = \left[\, g_{m,0}\ g_{m,1} \cdots g_{m,L_g-1} \,\right]^T,$$
$$\mathbf{s}_{ML_g}(k) = \left[\, \mathbf{s}_1^T(k)\ \mathbf{s}_2^T(k) \cdots \mathbf{s}_M^T(k) \,\right]^T,$$
$$\mathbf{s}_m(k) = \left[\, s_m(k)\ s_m(k-1) \cdots s_m(k-L_g+1) \,\right]^T.$$

In the z-transform domain, the multiple-input single-output (MISO) model is given by

$$Y(z) = \mathbf{g}^T(z)\mathbf{s}(z) + V(z), \tag{7.8}$$

where

$$\mathbf{g}(z) = \left[G_1(z)\ G_2(z) \cdots G_M(z) \right]^T,$$
$$G_m(z) = \sum_{l=0}^{L_g-1} g_{m,l} z^{-l},\ m = 1, 2, \ldots, M,$$
$$\mathbf{s}(z) = \left[S_1(z)\ S_2(z) \cdots S_M(z) \right]^T.$$

Note that $\mathbf{g}(z)$ defined here is slightly different from that in (7.6). We do not deliberately distinguish them.

7.2.4 MIMO Model

Figure 7.1(d) depicts a multiple-input multiple-output (MIMO) system. A MIMO system with M inputs and N outputs is referred to as an $M \times N$ system. At time k, we have

$$\mathbf{y}_{\mathrm{a}}(k) = \mathbf{G}\mathbf{s}_{ML_g}(k) + \mathbf{v}_{\mathrm{a}}(k), \tag{7.9}$$

where

$$\mathbf{y}_{\mathrm{a}}(k) = \left[y_1(k)\ y_2(k) \cdots y_N(k) \right]^T,$$
$$\mathbf{G} = \left[\mathbf{G}_1\ \mathbf{G}_2 \cdots \mathbf{G}_M \right],$$
$$\mathbf{G}_m = \begin{bmatrix} g_{1m,0} & g_{1m,1} & \cdots & g_{1m,L_g-1} \\ g_{2m,0} & g_{2m,1} & \cdots & g_{2m,L_g-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{Nm,0} & g_{Nm,1} & \cdots & g_{Nm,L-1} \end{bmatrix}_{N \times L_g},$$
$$m = 1, 2, \ldots, M,$$
$$\mathbf{v}_{\mathrm{a}}(k) = \left[v_1(k)\ v_2(k) \cdots v_N(k) \right]^T,$$

g_{nm} ($n = 1, 2, \ldots, N$, $m = 1, 2, \ldots, M$) is the impulse response of the channel from input m to output n, and $\mathbf{s}(k)$ is defined similarly to that in (7.7). Again, we have the model presented in the z-transform domain as

$$\mathbf{y}_{\mathrm{a}}(z) = \mathbf{G}(z)\mathbf{s}(z) + \mathbf{v}_{\mathrm{a}}(z), \tag{7.10}$$

where

$$\mathbf{G}(z) = \begin{bmatrix} G_{11}(z) & G_{12}(z) & \cdots & G_{1M}(z) \\ G_{21}(z) & G_{22}(z) & \cdots & G_{2M}(z) \\ \vdots & \vdots & \ddots & \vdots \\ G_{N1}(z) & G_{N2}(z) & \cdots & G_{NM}(z) \end{bmatrix},$$
$$G_{nm}(z) = \sum_{l=0}^{L_g-1} g_{nm,l} z^{-l},\ n = 1, 2, \ldots, N,\ m = 1, 2, \ldots, M.$$

Clearly the MIMO system is the most general model and all other three systems can be treated as special examples of a MIMO system.

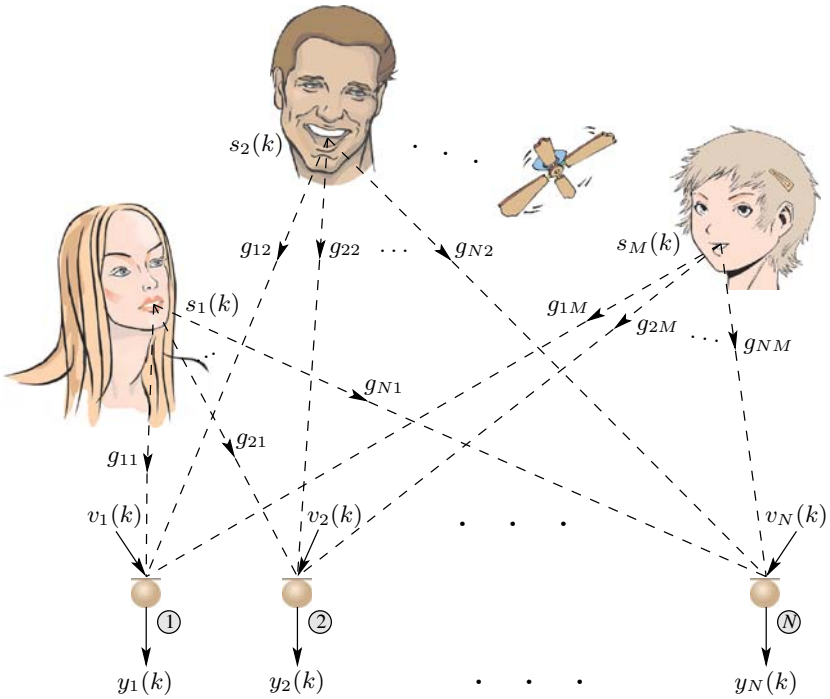


Fig. 7.2. Illustration of a microphone array system.

7.2.5 Problem Description

The problem considered in this chapter is illustrated in Fig. 7.2, where we have M sources in the sound field and we use N microphones to collect signals. We assume that the number of microphones used is greater than, or at least equal to the number of sound sources, i.e., $N \geq M$. Hence, the appropriate signal model is the MIMO system explained in Subsection 7.2.4. Some of the sources can be interferers. Since the additive noise case was studied in Chapters 4 and 5, we will neglect the background noise in the rest of this chapter, i.e., considering $v_n(k) = 0$. Our objective is then the extraction, from the observation signals, of some of the M radiating sources.

7.3 Two-Element Microphone Array

For ease of comprehending the fundamental principles, let us first consider the simple case where there are only two sources and two microphones. In this situation, the output signal at the nth microphone and at time k, is written as

$$y_n(k) = \mathbf{g}_{n1}^T \mathbf{s}_1(k) + \mathbf{g}_{n2}^T \mathbf{s}_2(k), \ n = 1, 2. \tag{7.11}$$

We consider that $s_1(k)$ is the signal of interest (speech source, for example) while $s_2(k)$ is the interference (noise source). Given the observations $y_n(k)$, the objective of this two-element microphone array is to recover $s_1(k)$. This would involve two processing operations: dereverberation and interference suppression. Suppose that we can achieve an estimate of $s_1(k)$ by applying two filters to the two microphone outputs, i.e.,

$$z(k) = \mathbf{h}_1^T \mathbf{y}_1(k) + \mathbf{h}_2^T \mathbf{y}_2(k), \tag{7.12}$$

where

$$\mathbf{h}_n = \begin{bmatrix} h_{n,0} \; h_{n,1} \cdots h_{n,L_h-1} \end{bmatrix}^T, \; n = 1, 2,$$

are two filters of length L_h and

$$\mathbf{y}_n(k) = \begin{bmatrix} y_n(k) \; y_n(k-1) \cdots y_n(k-L_h+1) \end{bmatrix}^T, \; n = 1, 2.$$

A legitimate question then arises: is it possible to find $\mathbf{h}_1$ and $\mathbf{h}_2$ in such a way that $z(k) = s_1(k-\tau)$ (where τ is a delay constant)? In other words, is it possible to perfectly recover $s_1(k)$ (up to a constant delay)? We will answer this question in the following subsections.

7.3.1 Least-Squares Approach

First, let us rewrite the microphone signals in the following vector/matrix form

$$\mathbf{y}_n(k) = \mathbf{G}_{n1}\mathbf{s}_{L,1}(k) + \mathbf{G}_{n2}\mathbf{s}_{L,2}(k), \; n = 1, 2, \tag{7.13}$$

where

$$\mathbf{G}_{nm} = \begin{bmatrix} g_{nm,0} & \cdots & g_{nm,L_g-1} & 0 & 0 & \cdots & 0 \\ 0 & g_{nm,0} & \cdots & g_{nm,L_g-1} & 0 & \cdots & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ 0 & 0 & \cdots & 0 & g_{nm,0} & \cdots & g_{nm,L_g-1} \end{bmatrix}, \quad n, m = 1, 2,$$

is a Sylvester matrix of size $L_h \times L$, with $L = L_g + L_h - 1$, and

$$\mathbf{s}_{L,m}(k) = \begin{bmatrix} s_m(k) \; s_m(k-1) \cdots s_m(k-L+1) \end{bmatrix}^T, \; m = 1, 2.$$

Substituting (7.13) into (7.12), we find that

$$z(k) = \left[\mathbf{h}_1^T\mathbf{G}_{11} + \mathbf{h}_2^T\mathbf{G}_{21}\right]\mathbf{s}_{L,1}(k) + \left[\mathbf{h}_1^T\mathbf{G}_{12} + \mathbf{h}_2^T\mathbf{G}_{22}\right]\mathbf{s}_{L,2}(k). \tag{7.14}$$

In order to perfectly recover $s_1(k)$, the following two conditions have to be met

$$\mathbf{G}_{11}^T\mathbf{h}_1 + \mathbf{G}_{21}^T\mathbf{h}_2 = \mathbf{u}, \tag{7.15}$$
$$\mathbf{G}_{12}^T\mathbf{h}_1 + \mathbf{G}_{22}^T\mathbf{h}_2 = \mathbf{0}_{L\times 1}, \tag{7.16}$$

where

$$\mathbf{u} = \left[0 \cdots 0\ 1\ 0 \cdots 0\right]^T$$

is a vector of length L, whose τth component is equal to 1. In matrix/vector form, the two previous conditions are

$$\mathbf{G}^T\mathbf{h} = \mathbf{u}', \tag{7.17}$$

where

$$\begin{aligned}\mathbf{G} &= \begin{bmatrix}\mathbf{G}_{11}\ \mathbf{G}_{12}\\ \mathbf{G}_{21}\ \mathbf{G}_{22}\end{bmatrix}\\ &= \left[\mathbf{G}_{:1}\ \mathbf{G}_{:2}\right],\\ \mathbf{h} &= \left[\mathbf{h}_1^T\ \mathbf{h}_2^T\right]^T,\\ \mathbf{u}' &= \left[\mathbf{u}^T\ \mathbf{0}_{L\times 1}^T\right]^T.\end{aligned}$$

Let us assume that the matrix $\mathbf{G}^T$ has full column rank. Since the number of its rows is always greater than the number of its columns, the best estimator we can derive from (7.17) is the least-squares (LS) filter

$$\begin{aligned}\mathbf{h}_{\text{LS}} &= \left[\mathbf{G}\mathbf{G}^T\right]^{-1}\mathbf{G}\mathbf{u}' \\ &= \left[\mathbf{h}_{\text{LS},1}^T\ \mathbf{h}_{\text{LS},2}^T\right]^T.\end{aligned} \tag{7.18}$$

This solution may not be good enough in practice for several reasons. First, we do not know how to determine L_h, the length of the LS filters $\mathbf{h}_{\text{LS},1}$ and $\mathbf{h}_{\text{LS},2}$. Second, the whole impulse response matrix $\mathbf{G}$ must be known to find the optimal filter in the LS sense, and thus there is very little flexibility with this method. In addition, it does not seem easy to quantify the amount of dereverberation and interference suppression separately.

7.3.2 Frost Algorithm

The Frost algorithm, also known as the linearly constrained minimum-variance (LCMV) filter (see Chapter 4), is another interesting structure for beamforming [76].

If we concatenate the two observation vectors together, we obtain

$$\begin{aligned}\mathbf{y}(k) &= \left[\mathbf{y}_1^T(k)\ \mathbf{y}_2^T(k)\right]^T\\ &= \mathbf{G}\begin{bmatrix}\mathbf{s}_{L,1}(k)\\ \mathbf{s}_{L,2}(k)\end{bmatrix} = \mathbf{G}\mathbf{s}_{2L}(k)\end{aligned}$$

and the covariance matrix of the observation signals is

$$\mathbf{R}_{yy} = E\left[\mathbf{y}(k)\mathbf{y}^T(k)\right] = \mathbf{G}\mathbf{R}_{ss}\mathbf{G}^T, \tag{7.19}$$

where $\mathbf{R}_{ss} = E\left[\mathbf{s}_{2L}(k)\mathbf{s}_{2L}^T(k)\right]$. In order for $\mathbf{R}_{yy}$ to be invertible, $\mathbf{R}_{ss}$ has to be invertible and $\mathbf{G}^T$ must have full column rank. In the rest, we assume that $\mathbf{R}_{yy}$ is nonsingular.

In the LCMV approach we would like to minimize the energy, $E\left[z^2(k)\right] = \mathbf{h}^T\mathbf{R}_{yy}\mathbf{h}$, at the outputs of the microphones without distorting the signal $s_1(k)$. This is equivalent to the optimization problem

$$\min_{\mathbf{h}} \mathbf{h}^T\mathbf{R}_{yy}\mathbf{h} \quad \text{subject to} \quad \mathbf{G}_{:1}^T\mathbf{h} = \mathbf{u}. \tag{7.20}$$

From (7.20), we see that this method will perfectly dereverberate the signal of interest (assuming that $\mathbf{G}_{:1}$ is known or is accurately estimated), while at the same time it will minimize the effect of the interference source, $s_2(k)$.

The problem in (7.20) can be solved by using a Lagrange multiplier to adjoin the constraints to the objective function. The solution can be easily deduced as

$$\begin{aligned}\mathbf{h}_{\text{LCMV}} &= \mathbf{R}_{yy}^{-1}\mathbf{G}_{:1}\left[\mathbf{G}_{:1}^T\mathbf{R}_{yy}^{-1}\mathbf{G}_{:1}\right]^{-1}\mathbf{u} \\ &= \left[\mathbf{h}_{\text{LCMV},1}^T \ \mathbf{h}_{\text{LCMV},2}^T\right]^T.\end{aligned} \tag{7.21}$$

In the previous expression, we assumed that the matrix $\left[\mathbf{G}_{:1}^T\mathbf{R}_{yy}^{-1}\mathbf{G}_{:1}\right]$ is nonsingular. A close inspection shows that two conditions need to be satisfied in order for this matrix to be invertible. The first one is $2L_h \geq L$, which implies that $L_h \geq L_g - 1$. This is very interesting since it tells us how to choose the minimum length of the two filters $\mathbf{h}_{\text{LCMV},1}$ and $\mathbf{h}_{\text{LCMV},2}$, which is something not seen from the LS approach. The second one is that $\mathbf{G}_{:1}$ has to have full column rank. If these two conditions are met, the LCMV filter exists and is unique. Note that in this approach, only the impulse responses from the desired source, i.e., $s_1(k)$ to the microphones, need to be known. In other words, only $\mathbf{G}_{:1}$ needs to be known, but not $\mathbf{G}_{:2}$.

We can always take the minimum required length for L_h, i.e. $L_h = L_g - 1$. In this case, $\mathbf{G}_{:1}$ is a square matrix and (7.21) becomes

$$\begin{aligned}\mathbf{h}_{\text{LCMV}} &= \left[\mathbf{G}_{:1}^T\right]^{-1}\mathbf{u} \\ &= \left[\mathbf{G}_{11}^T \ \mathbf{G}_{21}^T\right]^{-1}\mathbf{u},\end{aligned} \tag{7.22}$$

which does not depend on $\mathbf{R}_{yy}$. Expression (7.22) is exactly the multiple input/output inverse theorem (MINT) [166]. So for $L_h = L_g - 1$, we estimate $s_1(k)$ by dereverberating the observation signals $y_n(k)$ without much concern for the noise source $s_2(k)$. We assumed in this particular case that the square

matrix $\mathbf{G}_{:1}$ has full rank, which is equivalent to saying that the two polynomials formed from g_{11} and g_{21} share no common zeros. As L_h is increased compared to L_g, we still perfectly dereverberate the signal $s_1(k)$, while at the same time reduce the effect of the interference signal.

It's quite remarkable that the MINT method is a particular case of the Frost algorithm. However, this result should not come as a surprise since the motivation behind the two approaches is similar.

7.3.3 Generalized Sidelobe Canceller Structure

The generalized sidelobe canceller (GSC) transforms the LCMV algorithm from a constrained optimization problem into an unconstrained form. Therefore, the GSC and LCMV beamformers are essentially the same while the GSC has some implementation advantages [32], [33], [94], [95], [133], [230]. Given the channel impulse responses, the GSC method can be formulated by dividing the filter vector $\mathbf{h}$ into two components operating on orthogonal subspaces, as illustrated in Fig. 7.3. Here we assume that $L_h > L_g - 1$ so that the dimension of the nullspace of $\mathbf{G}_{:1}^T$ is not equal to zero. Mathematically, in the GSC structure, we have

$$\mathbf{h} = \mathbf{f} - \mathbf{B}\mathbf{w}, \tag{7.23}$$

where

$$\mathbf{f} = \mathbf{G}_{:1}\left[\mathbf{G}_{:1}^T\mathbf{G}_{:1}\right]^{-1}\mathbf{u} \tag{7.24}$$

is the minimum-norm solution of $\mathbf{G}_{:1}^T\mathbf{f} = \mathbf{u}$, $\mathbf{B}$ is the so-called blocking matrix that spans the nullspace of $\mathbf{G}_{:1}^T$, i.e. $\mathbf{G}_{:1}^T\mathbf{B} = \mathbf{0}_{L\times(2L_h-L)}$, and $\mathbf{w}$ is a weighting vector. The size of $\mathbf{B}$ is $2L_h \times (2L_h - L)$, where $2L_h - L$ is the dimension of the nullspace of $\mathbf{G}_{:1}^T$. Therefore, the length of $\mathbf{w}$ is $2L_h - L$.

The GSC approach is formulated as the following unconstrained optimization problem

$$\min_{\mathbf{w}} \left(\mathbf{f} - \mathbf{B}\mathbf{w}\right)^T \mathbf{R}_{yy} \left(\mathbf{f} - \mathbf{B}\mathbf{w}\right). \tag{7.25}$$

The solution is

$$\mathbf{w}_{\mathrm{GSC}} = \left[\mathbf{B}^T\mathbf{R}_{yy}\mathbf{B}\right]^{-1}\mathbf{B}^T\mathbf{R}_{yy}\mathbf{f}. \tag{7.26}$$

Equation (7.25) is equivalent to the minimization of $E\left[e^2(k)\right]$, where

$$e(k) = \mathbf{y}^T(k)\mathbf{f} - \mathbf{y}^T(k)\mathbf{B}\mathbf{w} \tag{7.27}$$

is the error signal between the outputs of the two filters $\mathbf{f}$ and $\mathbf{B}\mathbf{w}$.

In [28] (see also Chapter 2), it is shown that

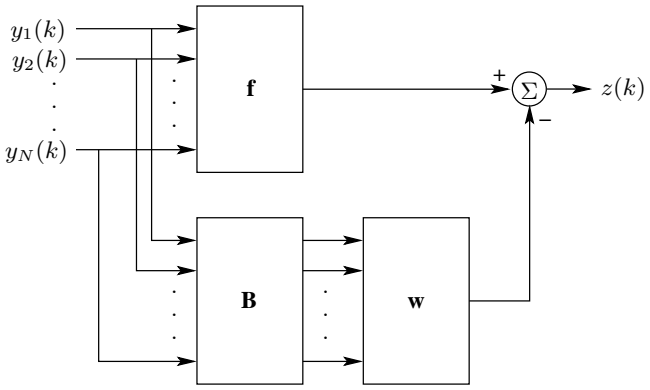


Fig. 7.3. The structure of a generalized sidelobe canceller.

$$\begin{aligned}\mathbf{h}_{\mathrm{LCMV}} &= \mathbf{R}_{yy}^{-1}\mathbf{G}_{:1}\left[\mathbf{G}_{:1}^{T}\mathbf{R}_{yy}^{-1}\mathbf{G}_{:1}\right]^{-1}\mathbf{u} \\ &= \left\{\mathbf{I}_{2L_h\times 2L_h} - \mathbf{B}\left[\mathbf{B}^{T}\mathbf{R}_{yy}\mathbf{B}\right]^{-1}\mathbf{B}^{T}\mathbf{R}_{yy}\right\}\mathbf{f} \\ &= \mathbf{h}_{\mathrm{GSC}} \end{aligned} \tag{7.28}$$

so the LCMV and GSC algorithms are equivalent.

Expressions (7.23) and (7.28) have a very nice physical interpretation [compared to (7.21)]. The LCMV filter $\mathbf{h}_{\mathrm{LCMV}}$ is the sum of two orthogonal vectors $\mathbf{f}$ and $-\mathbf{B}\mathbf{w}_{\mathrm{GSC}}$, which serve for different purposes. The objective of the first vector, $\mathbf{f}$, is to perform dereverberation on the signals $g_{11} * s_1$ and $g_{21} * s_1$, while the objective of the second vector, $-\mathbf{B}\mathbf{w}_{\mathrm{GSC}}$, is to reduce the effect of the interference $s_2(k)$. Increasing the length L_h of the filters $\mathbf{h}_{\mathrm{LCMV},1}$ and $\mathbf{h}_{\mathrm{LCMV},2}$ from its minimum value $L_g - 1$ will not change anything to the dereverberation part. However, increasing L_h will augment the dimension of the nullspace of $\mathbf{G}_{:1}^{T}$, and hence the length of $\mathbf{w}_{\mathrm{GSC}}$. As a result, better interference suppression is expected. It is obvious, from a theoretical point of view, that perfect dereverberation is possible (if $\mathbf{G}_{:1}$ is known or can be accurately estimated) but perfect interference suppression is not. In practice, if the two impulse responses g_{11} and g_{21} can be estimated, we can expect good dereverberation but interference suppression may be limited for the simple reason that it will be very hard to make L_h much larger than L_g (the length of the impulse responses g_{11} and g_{21}). In other words, as reverberation of the room increases, interference suppression decreases. This result was shown experimentally in [23], [92]. One possible way for improvement is to process the observation signals in two steps: the LCMV filter for dereverberation (first step) followed by a Wiener filter for noise reduction (second step); see, for examples, the methods proposed in [48], [160], [162], and [242]. This post-filtering approach may be effective from a noise reduction point of view but it will distort the desired signal $s_1(k)$.

7.4 N-Element Microphone Array

We now study the more general case of N microphones and M sources, with $M \leq N$. Without loss of generality, we assume that the first P ($P > 0$) signals, i.e., $s_p(k),\ p = 1, 2, \ldots, P$, are the desired sources while the other Q ($Q > 0$) source signals $s_{P+q}(k),\ q = 1, 2, \ldots, Q$, are the interferers, where $P+Q = M$. Given the observation signals $y_n(k),\ n = 1, 2, \ldots, N$, the objective of the array processing is to extract the signals $s_p(k),\ p = 1, 2, \ldots, P$. This implies dereverberation for the P desired sources and suppression of the Q interference signals.

Let

$$z_p(k) = \sum_{n=1}^{N} \mathbf{h}_{pn}^T \mathbf{y}_n(k),\ p = 1, 2, \ldots, P, \tag{7.29}$$

where

$$\mathbf{h}_{pn} = \left[h_{pn,0}\ h_{pn,1} \cdots h_{pn,L_h-1} \right]^T,\ p = 1, 2, \ldots, P,\ n = 1, 2, \ldots, N,$$

are PN filters of length L_h. We ask again the same question: is it possible to find $\mathbf{h}_{pn}$ in such a way that $z_p(k) = s_p(k - \tau_p)$ (where τ_p is some delay constant)? In other words, is it possible to perfectly recover $s_p(k)$ (up to a constant delay)? We discuss the possible solutions to this question in the succeeding subsections.

7.4.1 Least-Squares and MINT Approaches

The microphone signals can be rewritten in the following form

$$\mathbf{y}_n(k) = \sum_{m=1}^{M} \mathbf{G}_{nm} \mathbf{s}_{L,m}(k),\ n = 1, 2, \ldots, N. \tag{7.30}$$

Substituting (7.30) into (7.29), we find that

$$z_p(k) = \sum_{m=1}^{M} \left[\sum_{n=1}^{N} \mathbf{h}_{pn}^T \mathbf{G}_{nm} \right] \mathbf{s}_{L,m}(k),\ p = 1, 2, \ldots, P. \tag{7.31}$$

From the above expression, we see that in order to perfectly recover $s_p(k)$ the following M conditions have to be satisfied

$$\sum_{n=1}^{N} \mathbf{G}_{np}^T \mathbf{h}_{pn} = \mathbf{u}_p, \tag{7.32}$$

$$\sum_{n=1}^{N} \mathbf{G}_{nm}^T \mathbf{h}_{pn} = \mathbf{0}_{L\times 1},\ m = 1, 2, \ldots, M,\ m \neq p, \tag{7.33}$$

where

$$\mathbf{u}_p = \left[\, 0 \cdots 0\ 1\ 0 \cdots 0 \,\right]^T$$

is a vector of length L, whose τ_pth component is equal to 1. In matrix/vector form, the M previous conditions are

$$\mathbf{G}^T \mathbf{h}_{p:} = \mathbf{u}'_p, \tag{7.34}$$

where

$$\begin{aligned}
\mathbf{G} &= \begin{bmatrix} \mathbf{G}_{11} & \mathbf{G}_{12} & \cdots & \mathbf{G}_{1M} \\ \mathbf{G}_{21} & \mathbf{G}_{22} & \cdots & \mathbf{G}_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{G}_{N1} & \mathbf{G}_{N2} & \cdots & \mathbf{G}_{NM} \end{bmatrix} \\
&= \left[\, \mathbf{G}_{:1}\ \mathbf{G}_{:2} \cdots \mathbf{G}_{:M} \,\right], \\
\mathbf{h}_{p:} &= \left[\, \mathbf{h}_{p1}^T\ \mathbf{h}_{p2}^T \cdots \mathbf{h}_{pN}^T \,\right]^T, \\
\mathbf{u}'_p &= \Big[\ \underbrace{\mathbf{0}_{L\times 1}^T \quad \cdots \quad \mathbf{0}_{L\times 1}^T}_{(p-1)L} \quad \mathbf{u}_p^T \quad \underbrace{\mathbf{0}_{L\times 1}^T \quad \cdots \quad \mathbf{0}_{L\times 1}^T}_{(M-p)L} \ \Big]^T.
\end{aligned}$$

The channel matrix $\mathbf{G}$ is of size $NL_h \times ML$. Depending on the values of N and M, we have two cases, i.e., $N = M$ and $N > M$.

Case 1: $N = M$.

In this case, $ML = NL = NL_h + NL_g - N$. Since $L_g > 1$, we have $ML > NL_h$. This means that the number of rows of $\mathbf{G}^T$ is always larger than its number of columns. If we assume that the matrix $\mathbf{G}^T$ has full column rank, the LS solution for (7.34) is

$$\mathbf{h}_{\mathrm{LS},p:} = \left[\mathbf{G}\mathbf{G}^T\right]^{-1} \mathbf{G}\mathbf{u}'_p. \tag{7.35}$$

Here again, like in Section 7.3.1, we have no idea on how to choose L_h.

Case 2: $N > M$.

With more microphones than sources, is it possible to find a better solution than the LS one? Let $M = N - K$, $K > 0$. In fact, requiring $\mathbf{G}^T$ to have a number of rows that is equal to or larger than its number of columns, we find this time an upper bound for L_h:

$$L_h \leq \left(\frac{N}{K} - 1\right)(L_g - 1)\,. \tag{7.36}$$

If we take

$$L_h = \left(\frac{N}{K} - 1\right)(L_g - 1)\,, \tag{7.37}$$

and if L_h is an integer, $\mathbf{G}^T$ is now a square matrix. Therefore

$$\mathbf{h}_{\text{MINT},p:} = \left[\mathbf{G}^T\right]^{-1} \mathbf{u}'_p. \tag{7.38}$$

This is identical to the MINT method [123], [166], which can perfectly recover the signal of interest $s_p(k)$ if $\mathbf{G}$ is known or can be accurately estimated. Of course, we supposed that $\mathbf{G}^T$ has full rank, which is equivalent to saying that the polynomials formed from $g_{1m}, g_{2m}, \ldots, g_{Nm},\ m = 1, 2, \ldots, M$, share no common zeroes.

It is very interesting to see that, if we have more microphones than sources, we have more flexibility in estimation of the signals of interest and have a better idea for the choice of L_h.

7.4.2 Frost Algorithm

Following (7.30), if we concatenate the N observation vectors together, we get

$$\begin{aligned}\mathbf{y}(k) &= \left[\mathbf{y}_1^T(k)\ \mathbf{y}_2^T(k)\ \cdots\ \mathbf{y}_N^T(k)\right]^T \\ &= \mathbf{G}\mathbf{s}_{ML}(k),\end{aligned}$$

where

$$\mathbf{s}_{ML}(k) = \left[\mathbf{s}_{L,1}^T(k)\ \mathbf{s}_{L,2}^T(k)\ \cdots\ \mathbf{s}_{L,M}^T(k)\right]^T.$$

The covariance matrix corresponding to $\mathbf{y}(k)$ is

$$\mathbf{R}_{yy} = E\left[\mathbf{y}(k)\mathbf{y}^T(k)\right] = \mathbf{G}\mathbf{R}_{ss}\mathbf{G}^T, \tag{7.39}$$

with $\mathbf{R}_{ss} = E\left[\mathbf{s}_{ML}(k)\mathbf{s}_{ML}^T(k)\right]$. We suppose that $\mathbf{R}_{yy}$ is invertible, which is equivalent to stating that the $\mathbf{R}_{ss}$ matrix is of full rank and $\mathbf{G}^T$ matrix has full column rank. We are now ready to study two interesting cases.

Case 1: Partial Knowledge of the Impulse Response Matrix.

In this case, we wish to extract the source $s_p(k)$ with only the knowledge of $\mathbf{G}_{:p}$, i.e., the impulse responses from that source to the N microphones. With this information, the LCMV filter is obtained by solving the following problem

$$\min_{\mathbf{h}_{p:}} \mathbf{h}_{p:}^T \mathbf{R}_{yy} \mathbf{h}_{p:} \quad \text{subject to} \quad \mathbf{G}_{:p}^T \mathbf{h}_{p:} = \mathbf{u}_p. \tag{7.40}$$

Hence

$$\mathbf{h}_{\text{LCMV1},p:} = \mathbf{R}_{yy}^{-1}\mathbf{G}_{:p}\left[\mathbf{G}_{:p}^T\mathbf{R}_{yy}^{-1}\mathbf{G}_{:p}\right]^{-1}\mathbf{u}_p. \tag{7.41}$$

We refer to this approach as the LCMV1, where a necessary condition for $\left[\mathbf{G}_{:p}^T\mathbf{R}_{yy}^{-1}\mathbf{G}_{:p}\right]$ to be nonsingular is to have $NL_h \geq L$, which implies that

$$L_h \geq \frac{L_g - 1}{N - 1}. \tag{7.42}$$

An important thing to observe is that the minimum length required for the filters $\mathbf{h}_{\text{LCMV1},pn},\ n = 1, 2, \ldots, N$, decreases as the number of microphones increases. As a consequence, the Frost filter has the potential to significantly reduce the effect of the interferers with a large number of microphones.

If we take the minimum required length for L_h, i.e., $L_h = (L_g - 1)/(N - 1)$ and assume that L_h is an integer, $\mathbf{G}_{:p}$ turns to be a square matrix and (7.41) becomes

$$\begin{aligned}\mathbf{h}_{\text{LCMV1},p:} &= \left[\mathbf{G}_{:p}^T\right]^{-1}\mathbf{u}_p \\ &= \left[\,\mathbf{G}_{1p}^T\ \mathbf{G}_{2p}^T\cdots\mathbf{G}_{Np}^T\,\right]^{-1}\mathbf{u}_p,\end{aligned} \tag{7.43}$$

which is the MINT method [166]. We assumed in (7.43) that $\mathbf{G}_{:p}$ has full rank, which is equivalent to saying that the N polynomials formed from $g_{1p}, g_{2p}, \ldots, g_{Np}$ share no common zeros. Mathematically, this condition is expressed as follows

$$\begin{aligned}&\gcd\left[G_{1p}(z), G_{2p}(z), \cdots, G_{Np}(z)\right] = 1 \\ &\Leftrightarrow\ \exists\ H_{p1}(z), H_{p2}(z), \cdots, H_{pN}(z) :\ \sum_{n=1}^{N} G_{np}(z)H_{pn}(z) = 1,\end{aligned} \tag{7.44}$$

where $\gcd[\cdot]$ denotes the greatest common divisor of the polynomials involved and, $G_{np}(z)$ and $H_{pn}(z)$ are the z-transforms of g_{np} and h_{pn}, respectively. This is known as the Bezout theorem.

From (7.39), we can deduce that a necessary condition for $\mathbf{R}_{yy}$ to be invertible is to have $NL_h \leq ML$. When $M = N$, i.e., the number of sources is equal to the number of microphones, this condition is always true, which means that there is no upper bound for L_h. When $N > M$, assume that $M = N - K$, $K > 0$, this condition becomes

$$L_h \leq \left(\frac{N}{K} - 1\right)(L_g - 1)\,. \tag{7.45}$$

Combining (7.45) and (7.42), we see how L_h is bounded, i.e.,

$$\frac{L_g - 1}{N - 1} \leq L_h \leq \left(\frac{N}{K} - 1\right)(L_g - 1)\,. \tag{7.46}$$

Case 2: Full Knowledge of the Impulse Response Matrix and $N > M$.

Here, we wish to extract source $s_p(k)$ with the full knowledge of the impulse response matrix $\mathbf{G}$, with $M = N - K$, $K > 0$. Taking all this information into account in our optimization problem

$$\min_{\mathbf{h}_{p:}} \mathbf{h}_{p:}^T\mathbf{R}_{yy}\mathbf{h}_{p:} \quad \text{subject to} \quad \mathbf{G}^T\mathbf{h}_{p:} = \mathbf{u}'_p, \tag{7.47}$$

we find the solution

$$\mathbf{h}_{\text{LCMV2},p:} = \mathbf{R}_{yy}^{-1}\mathbf{G}\left[\mathbf{G}^T\mathbf{R}_{yy}^{-1}\mathbf{G}\right]^{-1}\mathbf{u}_p'. \tag{7.48}$$

We refer to this approach as the LCMV2, where we assume that both $\mathbf{R}_{yy}$ and $\left[\mathbf{G}^T\mathbf{R}_{yy}^{-1}\mathbf{G}\right]$ are nonsingular and their inverse matrices exist. From the previous analysis, we know that in order for $\mathbf{R}_{yy}$ to be invertible the condition in (7.45) has to be true. Also, a necessary condition for $\left[\mathbf{G}^T\mathbf{R}_{yy}^{-1}\mathbf{G}\right]$ to be nonsingular is to have $NL_h \geq ML$, which implies that

$$L_h \geq \left(\frac{N}{K} - 1\right)(L_g - 1). \tag{7.49}$$

Therefore, the only condition for (7.48) to exist is that

$$L_h = \left(\frac{N}{K} - 1\right)(L_g - 1), \tag{7.50}$$

and this value needs to be an integer. In this case, $\mathbf{G}$ is a square matrix and (7.48) becomes

$$\mathbf{h}_{\text{LCMV2},p:} = \left[\mathbf{G}^T\right]^{-1}\mathbf{u}_p', \tag{7.51}$$

which is also the MINT solution [166]. Also, it is shown in [123] how to convert an $M \times N$ MIMO system (with $M < N$) into M interference-free SIMO systems. The MINT method is then applied in each one of these SIMO systems to remove the channel effect. So this two-step approach (see Chapter 8) is equivalent to the LCMV2.

7.4.3 Generalized Sidelobe Canceller Structure

The GSC structure [94] makes sense only for the LCMV1 filter. We need to take $L_h > (L_g - 1)/(N - 1)$ in order that the dimension of the nullspace of $\mathbf{G}_{:p}^T$ is not equal to zero. We already know that the GSC method solves exactly the same problem as the Frost algorithm by decomposing the filter $\mathbf{h}_{p:}$ into two orthogonal components [32], [133], [230]:

$$\mathbf{h}_{p:} = \mathbf{f}_p - \mathbf{B}_p\mathbf{w}_p, \tag{7.52}$$

where

$$\mathbf{f}_p = \mathbf{G}_{:p}\left[\mathbf{G}_{:p}^T\mathbf{G}_{:p}\right]^{-1}\mathbf{u}_p \tag{7.53}$$

is the minimum-norm solution of $\mathbf{G}_{:p}^T\mathbf{f}_p = \mathbf{u}_p$ and $\mathbf{B}_p$ is the blocking matrix that spans the nullspace of $\mathbf{G}_{:p}^T$, i.e. $\mathbf{G}_{:p}^T\mathbf{B}_p = \mathbf{0}_{L\times(NL_h-L)}$. The size of $\mathbf{B}_p$ is $NL_h \times (NL_h - L)$, where $NL_h - L$ is the dimension of the nullspace of $\mathbf{G}_{:p}^T$.

Therefore, $\mathbf{w}_p$ is a vector of length $L_w = NL_h - L = (N-1)L_h - L_g + 1$, which is obtained from the following unconstrained optimization problem

$$\min_{\mathbf{w}_p} \left(\mathbf{f}_p - \mathbf{B}_p\mathbf{w}_p\right)^T \mathbf{R}_{yy} \left(\mathbf{f}_p - \mathbf{B}_p\mathbf{w}_p\right), \tag{7.54}$$

and the solution is

$$\mathbf{w}_{\mathrm{GSC},p} = \left[\mathbf{B}_p^T\mathbf{R}_{yy}\mathbf{B}_p\right]^{-1}\mathbf{B}_p^T\mathbf{R}_{yy}\mathbf{f}_p. \tag{7.55}$$

Our discussion is going to focus on two situations. The first one is when the number of microphones is equal to the number of sources[1] ($N = M$). In this case, we know from the previous subsection that there is no upper bound for L_h. This implies that the length of $\mathbf{w}_{\mathrm{GSC},p}$ can be taken as large as we wish. As a result, we can expect better interference suppression as L_h is increased. By increasing the number of microphones (with $N = M$), the minimum length required for L_h will decrease compared to L_g, which is a very good thing because in practice acoustic impulse responses can be very long.

Our second situation is when we have more microphones than sources. Assume that $M = N - K$, $K > 0$. Using (7.46) and the fact that $L_w = (N-1)L_h - L_g + 1$, we can easily deduce the bounds for the length of $\mathbf{w}_{\mathrm{GSC},p}$:

$$0 < L_w \leq \frac{N}{K}(N-K-1)(L_g-1) \leq \frac{N}{K}(N-K-1)(N-1)L_h. \tag{7.56}$$

This means that there is a limit to interference suppression. Consider the scenario where we have one desired source only ($P = 1$) and Q interferers. We have $M = Q + 1 = N - K$ and (7.56) is now:

$$0 < L_w \leq \frac{NQ}{N-Q-1}(L_g-1) \leq \frac{N(N-1)Q}{N-Q-1}L_h. \tag{7.57}$$

We see from (7.57) that the upper bound of L_w depends on three factors: the reverberation condition (L_g), the number of interference sources (Q), and the number of microphones (N). When Q and N are fixed, if the length of the room impulse response L_g increases, this indicates that the environment is more reverberant and the interference suppression problem will become more difficult. So we have to increase L_w to compensate for the additional reflections. In case that L_g and N remain the same, but the number of interference sources Q increases, this implies that we have more interferers to cope with so we have to use a larger L_w. Now suppose that L_g and Q remain the same, if we increase the number of microphones, this will allow us to use a larger

[1] There is no distinction here between the interference and desired sources. By extracting the signal of interest $s_p(k)$ from the rest, the algorithm will see the other desired sources as interferences. We assume that all sources are active at the same time; if it's not the case, we will be in a situation where we have more microphones than sources.

value for L_w. We should, however, make the distinction between this case and the former two situations. When we have more microphones, we achieve more realizations of the source signals. So we can increase L_w to augment the interference-suppression performance. But in the former two situations, we would expect some degree of performance degradation since the problem becomes more difficult to solve as L_g and Q increase.

7.4.4 Minimum Variance Distortionless Response Approach

The minimum variance distortionless response (MVDR) method, due to Capon [35], [149] is a particular case of the LCMV1. The MVDR applies only one constraint

$$\mathbf{g}_{:p}^T(\kappa_p)\mathbf{h}_{p:} = 1, \tag{7.58}$$

where $\mathbf{g}_{:p}(\kappa_p)$ is the κ_pth column of the matrix $\mathbf{G}_{:p}$. The aim of this constraint is to align the desired source signal, $s_p(k)$, at the output of the beamformer. Hence, in the MVDR approach, we have the following optimization problem:

$$\min_{\mathbf{h}_{p:}} \mathbf{h}_{p:}^T\mathbf{R}_{yy}\mathbf{h}_{p:} \quad \text{subject to} \quad \mathbf{g}_{:p}^T(\kappa_p)\mathbf{h}_{p:} = 1, \tag{7.59}$$

whose solution is

$$\mathbf{h}_{\text{MVDR},p:} = \frac{\mathbf{R}_{yy}^{-1}\mathbf{g}_{:p}(\kappa_p)}{\mathbf{g}_{:p}^T(\kappa_p)\mathbf{R}_{yy}^{-1}\mathbf{g}_{:p}(\kappa_p)}. \tag{7.60}$$

The minimum required length for the filters $\mathbf{h}_{\text{MVDR},pn}$ is $L_h = \kappa_p$. In this case, the performance of the MVDR beamformer is similar to that of the classical delay-and-sum beamformer. As L_h is increased compared to κ_p, the signal of interest will still be aligned at the output of the beamformer, while other signals will tend to be attenuated.

This method can be very useful in practice, since it does not require the full knowledge of the impulse responses but only the relative delays among microphones. However, an adaptive implementation of the MVDR may cancel the desired signal [30], [49], [50], [53], [219], [220], [241].

7.5 Simulations

The section compares different algorithms via simulations in a realistic acoustic environment.

7.5.1 Acoustic Environments and Experimental Setup

Same as in Section 5.10, the experiments were conducted with the acoustic impulse responses measured in the varechoic chamber at Bell Labs. The layout

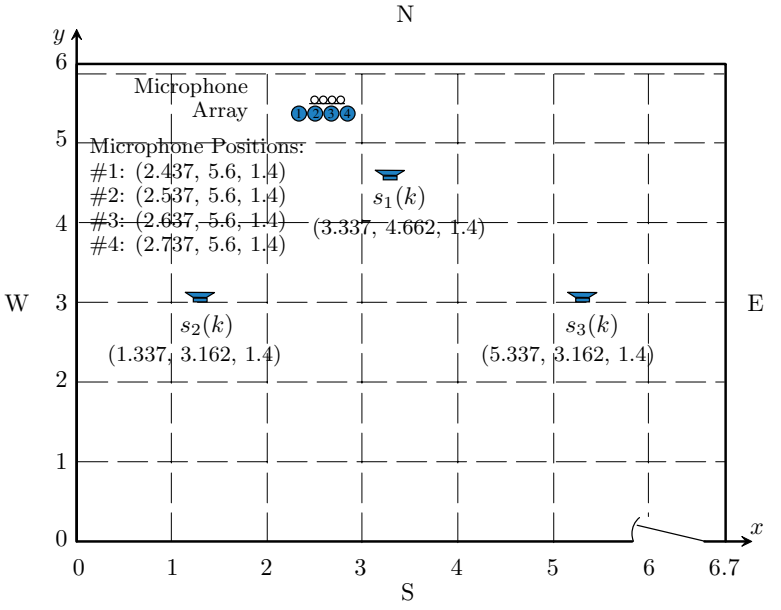


Fig. 7.4. Layout of the experimental setup in the varechoic chamber (coordinate values measured in meters). The three sources are placed, respectively, at (3.337, 4.662, 1.6), (1.337, 3.162, 1.6), and (5.337, 3.162, 1.6). The four microphones in the linear array are located, respectively, at (2.437, 5.6, 1.4), (2.537, 5.6, 1.4), (2.637, 5.6, 1.4), and (2.737, 5.6, 1.4).

of the experimental setup is illustrated in Fig. 7.4, where a linear array which consists of 4 omni-directional microphones were employed with their positions being, respectively, at (2.437, 5.6, 1.4), (2.537, 5.6, 1.4), (2.637, 5.6, 1.4), and (2.737, 5.6, 1.4) (coordinate values measured in meters). We have three sources in the sound field: one target $s_1(k)$ is located at (3.337, 4.662, 1.6), and two interferers, $s_2(k)$ and $s_3(k)$, are placed at (1.337, 3.162, 1.6) and (5.337, 3.162, 1.6) respectively. The objective of this study is to investigate how the desired signal $s_1(k)$ can be dereverberated and the two interference sources, $s_2(k)$ and $s_3(k)$, can be suppressed or cancelled when four microphones are used. We consider the reverberation condition with the 60-dB reverberation time $T_{60} = 310$ ms. The impulse response from each source to each microphone was measured originally at 48 kHz, and then downsampled to 8 kHz. The microphone outputs are computed by convolving the source signal with the corresponding channel impulse responses.

To visualize the performance of different beamforming algorithms, we first conduct a simple experiment where all the impulse responses are truncated to only 64 points (the zeros commonly shared by all the impulse responses at the beginning are removed). All the three source signals are prerecorded speech

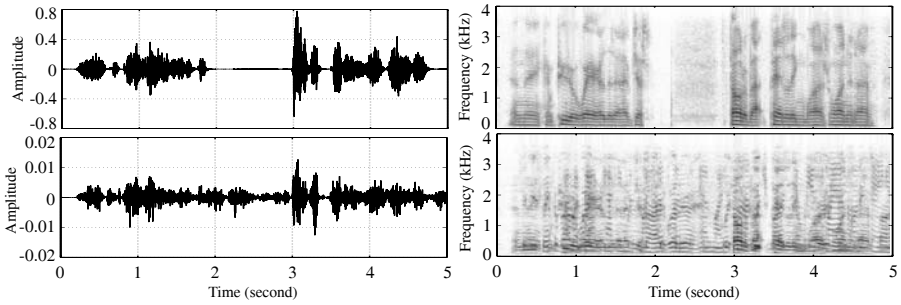


Fig. 7.5. Time sequence and the corresponding spectrogram of: the desired source signal $s_1(k)$ from a male speaker (the upper trace) and the output of microphone 1, i.e., $x_1(k)$ (the lower trace).

sampled at 8 kHz where $s_1(k)$ is from a male speaker and both $s_2(k)$ and $s_3(k)$ are from a same female speaker. The waveform and spectrogram of the first 5 seconds of $s_1(k)$ are shown in Fig 7.5. The microphone outputs are obtained by convolving the three source signals with the corresponding impulse responses. Figure 7.5 also plots the first 5 seconds of the signal observed at the first microphone.

To extract $s_1(k)$, we need to estimate the filter $\mathbf{h}_1$. This would require knowledge about the impulse responses from the three sources to the four microphones. In this experiment, we assume that the impulse responses are known *a priori*, so the results in this case demonstrate the upper limit of each algorithm in a given condition. Another parameter that has to be determined is the length of the $\mathbf{h}_1$ filter, i.e., L_h. Throughout the text, we have analyzed the bounds of L_h for different algorithms. In this experiment, L_h is chosen as its maximum value that can be taken according to (7.37), (7.45), (7.50), and (7.56) and is set to the same for all the algorithms. Note that with this optimum choice of L_h, the pseudoinverse of the channel matrix is equal to its normal inverse. So under this condition, the LS and LCMV2 methods will produce the same results. In addition, we already see from Section 7.4 that LCMV2 and MINT are the same. The outputs of the different beamformers are plotted in Fig. 7.6.

It can be seen from Fig. 7.6 that both the LS and LCMV2 (MINT) approaches have achieved almost perfect interference suppression and speech dereverberation. However, the outputs of the LCMV1 and GSC still consist of a small amount of interference signals. Apparently, the LCMV1 and GSC are less effective than the LS and LCMV2 (MINT) techniques in terms of interference suppression. This is comprehensible since the LCMV1 and GSC employ only the channel information from the desired source to the microphones while both the LS and LCMV2 (MINT) techniques use not only the impulse responses from the desired source but also those from all the interferers. In addition, we see that the MVDR is inferior to all the other studied

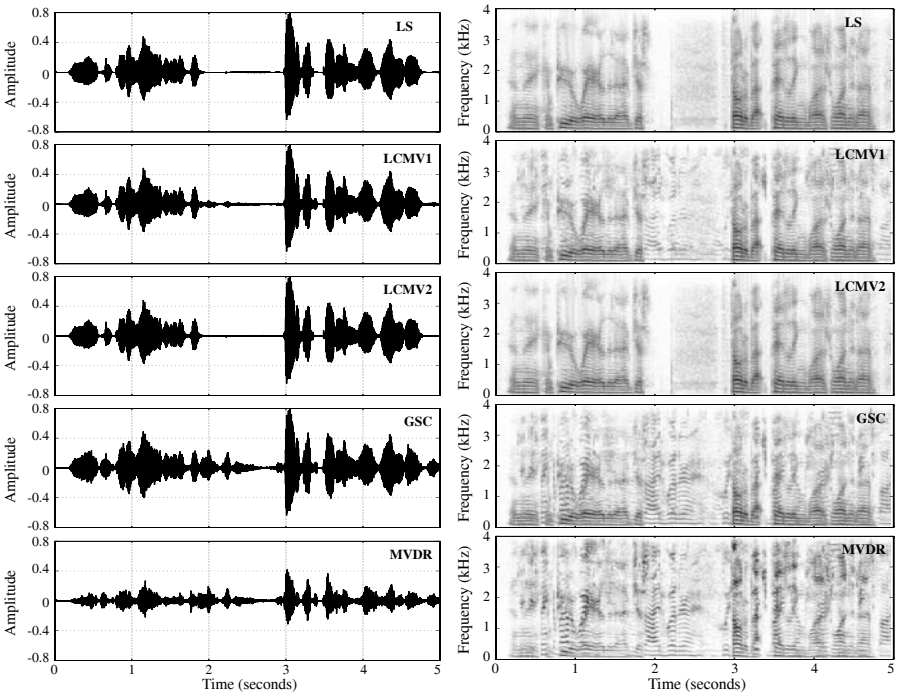


Fig. 7.6. Time sequence and the corresponding spectrogram of different beamforming algorithms, where $L_g = 64$ and $L_h = 189$ for all the algorithms. Note that under this condition, the LS, LCMV2, and MINT methods are theoretically the same.

techniques in performance. Such a result is not surprising since the MVDR poses less constraints as compared to the other techniques.

To quantitatively assess the performance of interference suppression and speech dereverberation, we now evaluate two criteria, namely signal-to-interference ratio (SIR) and speech spectral distortion. For the notion of SIR, see [129]. Here, though we have M sources, our interest is in extracting only the target signal, i.e., the first source $s_1(k)$, so the average input SIR at microphone n is defined as

$$\mathrm{SIR}_n^{\mathrm{in}} = \frac{E\left\{[g_{n1} * s_1(k)]^2\right\}}{\sum_{m=2}^{M} E\left\{[g_{nm} * s_m(k)]^2\right\}}, \quad n = 1, 2, \ldots, N. \tag{7.61}$$

The overall average input SIR is then given by

$$\mathrm{SIR}^{\mathrm{in}} = \frac{1}{N} \sum_{n=1}^{N} \mathrm{SIR}_n^{\mathrm{in}}. \tag{7.62}$$

The output SIR is defined using the same principle but the expression will be slightly more complicated. For a concise presentation, we denote the impulse

Table 7.1. Performance of interference suppression and speech dereverberation using different beamforming algorithms where the MIMO impulse responses are known *a priori*.

			LS		LCMV1		LCMV2 (MINT)		GSC		MVDR	
SIR^{in} (dB)	L_g	L_h	SIR^{o} (dB)	IS	SIR^{o} (dB)	IS	SIR^{o} (dB)	IS	SIR^{o} (dB)	IS	SIR^{o} (dB)	IS
		189*	187.6	0.00	18.0	0.00	187.6	0.00	14.5	0.00	4.8	6.28
−9.2	64	150	9.3	0.02	9.1	0.00	×	×	9.1	0.00	4.3	6.65
		100	7.2	0.08	−0.5	0.00	×	×	−0.5	0.00	3.4	7.86
		50	4.5	0.13	−8.0	0.00	×	×	−8.0	0.00	2.7	8.17
		381*	171.3	0.00	9.6	0.00	171.3	0.00	4.1	0.00	4.2	6.86
−8.1	128	360	24.7	0.01	3.9	0.00	×	×	3.9	0.00	4.2	6.86
		320	14.3	0.01	2.8	0.00	×	×	2.8	0.00	4.2	6.75
		200	3.8	0.13	−3.9	0.00	×	×	−3.9	0.00	3.3	7.22
		765*	117.2	0.00	7.9	0.00	117.2	0.00	1.5	0.00	4.4	7.68
−8.3	256	700	24.8	0.03	1.3	0.00	×	×	1.3	0.00	4.4	7.56
		600	11.2	0.23	0.1	0.00	×	×	0.1	0.00	4.5	7.38
		300	4.0	0.15	−6.7	0.00	×	×	−6.7	0.00	3.0	9.07

NOTES: *: the maximum value that the L_h can take for the condition;
×: the L_h cannot take this value for the method in the given condition.

response of the equivalent channel between the mth source and the beamforming output as f_m, which can be expressed as

$$f_m = \sum_{n=1}^{N} h_{1n} * g_{nm}, \tag{7.63}$$

where h_{1n} is the filter between microphone n and the beamforming output, and g_{nm} is the impulse response between source m and microphone n. The output SIR can then be written as

$$\text{SIR}^{\text{o}} = \frac{E\left\{[f_1 * s_1(k)]^2\right\}}{\sum_{m=2}^{M} E\left\{[f_m * s_m(k)]^2\right\}}. \tag{7.64}$$

If we express both SIR^{o} and SIR^{in} in decibels, the difference between the two reflects the performance of interference suppression.

To evaluate speech dereverberation, we investigate the IS distance [38], [131], [185], [187] between $s_1(k)$ and $s_1(k) * f_m$, which evaluates the amount of reverberation present in the estimated speech signal after beamforming. The smaller the IS distance, the more effective will be the beamforming algorithm in dereverberation.

Table 7.1 summarizes the experimental results, where the source signals are the same as used in the previous experiment. The following observations can be made:

- As the length of the impulse responses, i.e., L_g, increases, the maximum achievable (with the maximum L_h) gain in SIR decreases. This occurs to all the algorithms. Such a result should not come as a surprise. As L_g increases, each microphone receives more reflections (with longer delays) from both the desired and interference sources. Consequently, the received speech becomes more distorted and the estimation problem tends to be more difficult.
- In the ideal condition where impulse responses are known and L_h is set to its maximum value, both the LS and LCMV2 (MINT) techniques can achieve almost perfect interference suppression and speech dereverberation. The SIR gains are more than 100 dB and the IS distances are approximately zero. Similar to the LS and LCMV2 (MINT) methods, the LCMV1 and GSC can also perform perfect speech dereverberation, but their interference suppression performance is limited. The underlying reason for this has been explained earlier on. Briefly, it is because the LCMV1 and GSC do not use the channel information from the interferers to the microphones.
- In each reverberant condition (a fixed L_g), if we reduce the length of the $\mathbf{h}_1$ filter, the amount of interference suppression decreases significantly for all the methods except for the MVDR. Therefore, if we want a reasonable amount of interference suppression, the length of the filter $\mathbf{h}_1$ should be set to a large value. However, this length is upper bounded, as explained in Section 7.4.
- The IS distances obtained by the LS, LCMV1, LCMV2 (MINT), and GSC methods are close to zero, indicating that these techniques have accomplished good speech dereverberation. This coincides with the theoretical analysis made throughout the text.
- In terms of interference rejection, the MVDR method is very robust to the changes of both L_g and L_h. When L_h is small, this method can even achieve more interference suppression than the other four approaches. However, the values of the IS distance with this method are very large. Therefore, we may have to use dereverberation techniques in order to further reduce speech distortion.

In the preceding experiments, we assumed that the impulse responses were known *a priori*. In real applications, it is very difficult if not impossible to know the true impulse responses. Therefore, we have to estimate such information based on the data observed at the microphones. In our application scenario, the source signals are generally not accessible, so the estimation of channel impulse responses have to be done in a blind manner. However, blind identification of a MIMO system is a very difficult problem and no effective solution is available thus far, particularly for acoustic applications. Fortunately, in natural communication environments, not all the sources are active at the same time. In many time periods, the observation signal is occupied exclusively by a single source. If we can detect those periods, the MIMO identification prob-

Table 7.2. Performance of interference suppression and speech dereverberation with different beamforming algorithms where the channel impulse responses are estimated using a blind technique.

				LS		LCMV1		LCMV2 (MINT)		GSC		MVDR	
SIR^{in} (dB)	L_g	$L_{\hat{g}}$	L_h	SIR^{o} (dB)	IS	SIR^{o} (dB)	IS	SIR^{o} (dB)	IS	SIR^{o} (dB)	IS	SIR^{o} (dB)	IS
−9.23	64	64	189	140.1	0.0	14.5	0.0	140.1	0.0	14.5	0.0	4.8	6.3
		50	147	−5.9	6.1	8.3	0.6	×	×	9.0	0.5	4.3	7.0
−8.04	128	128	381	133.1	0.0	4.1	0.0	133.1	0.0	4.1	0.0	4.2	6.9
		100	297	−4.7	5.9	4.9	0.9	×	×	3.6	0.9	4.1	7.1

NOTES: L_g: the length of true impulse responses;
$L_{\hat{g}}$: the length of the channel impulse responses used during blind channel identification.

lem can be converted to a SIMO identification problem in each time period. This is assumed to be the case in our study and the channel impulse responses are estimated using the techniques developed in [123]. After the estimation of channel impulse responses, we can recover the desired source signals by beamforming. The results for this experiment are shown in Table 7.2 where we studied two situations. While in the first one, we assume that we know the length of the true impulse responses during blind channel identification, in the second case, the length of the modeling filter i.e., $L_{\hat{g}}$, during blind channel identification is set to less than L_g. Evidently, the second case is more realistic since in reality the real impulse responses can be very long, but we cannot use a very long modeling filter due to many practical limitations.

Comparing Tables 7.2 and 7.1, one can see that, when $L_{\hat{g}} = L_g$, all the techniques suffer some but not significant performance degradation. However, if $L_{\hat{g}}$ is less than L_g, which is true in most real applications, the LS and LCMV2 (MINT) suffer significant performance degradation in both interference suppression and speech dereverberation. The reason may be explained as follows. In our case, we truncated the impulse response to either 64 or 128 points. Due to the strong reverberation, the tail of the truncated impulse responses consists of significant energy. As a result, dramatic errors were introduced during channel identification when decreasing $L_{\hat{g}}$. This in turn degrades the performance of beamforming. However, comparing with the LS and LCMV2 (MINT), we see that the LCMV1 and GSC suffer some but not serious deterioration. We also noticed a very interesting property of the MVDR approach from Tables 7.2 that its performance does not deteriorate much as $L_{\hat{g}}$ decreases. This robust feature is due to the fact that the MVDR poses less constraints than the other studied methods. But, as we noticed before, the MVDR suffers dramatic signal distortion, as indicated by its large IS dis-

tances. So further dereverberation techniques may have to be considered after the MVDR processing if possible.

7.6 Conclusions

This chapter was concerned with interference suppression and speech dereverberation using microphone arrays. We developed a general framework for microphone array beamforming, in which beamforming is treated as a MIMO signal processing problem. Under this general framework, we analyzed the lower and upper bounds for the length of the beamforming filter, which governs the performance of beamforming in terms of speech dereverberation and interference suppression. We discussed the intrinsic relationships among the most classical beamforming techniques and explained, from the channel condition point of view, what are the necessary conditions for the different beamforming techniques to work. Theoretical analysis as well as experimental results showed that the impulse responses from both the desired sources and the interferers have to be employed in order to achieve good interference suppression and speech dereverberation. In practice, however, the true impulse responses are in general not accessible. Therefore, we have to estimate them with blind techniques. But these techniques, as of today, are still not very accurate and lack robustness. As a result, microphone-array beamforming algorithms will be affected. As to what degree the impulse responses mismatch would affect the beamforming algorithms, it is worth of further investigation.

8

Sequential Separation and Dereverberation: the Two-Stage Approach

8.1 Introduction

This chapter will continue the discussion started in the previous chapter on source extraction (or separation) and speech dereverberation with classical approaches. The same MIMO framework will be used for analysis. But instead of trying to determine a solution in one step, we will present a two-stage approach for sequential separation and dereverberation. This will help the reader better comprehend the interactions between spatial and temporal processings in a microphone array system.

8.2 Signal Model and Problem Description

The problem of source separation and speech dereverberation has been clearly described in Section 7.2. But for the self containment of this chapter and for the convenience of the readers, we decide to briefly repeat the signal model in the following.

We consider an N-element microphone array in a reverberant acoustic environment in which there are M sound sources. This is an $M \times N$ MIMO system. As shown in Fig. 8.1, the nth microphone output is expressed as

$$y_n(k) = \sum_{m=1}^{M} g_{nm} * s_m(k) + v_n(k), \quad n = 1, 2, \ldots, N. \tag{8.1}$$

The objective of separation and dereverberation is to retrieve the source signals $s_m(k)$ $(m = 1, 2, \ldots, M)$ by applying a set of filters h_{mn} $(m = 1, 2, \ldots, M,\ n = 1, 2, \ldots, N)$ to the microphone outputs $y_n(k)$ $(n = 1, 2, \ldots, N)$, as illustrated by Fig. 8.1. In the absence of additive noise, the resulting signal of separation and dereverberation is obtained as

$$\mathbf{z}_{\mathrm{a}}(k) = \mathbf{HGs}_{ML}(k), \tag{8.2}$$

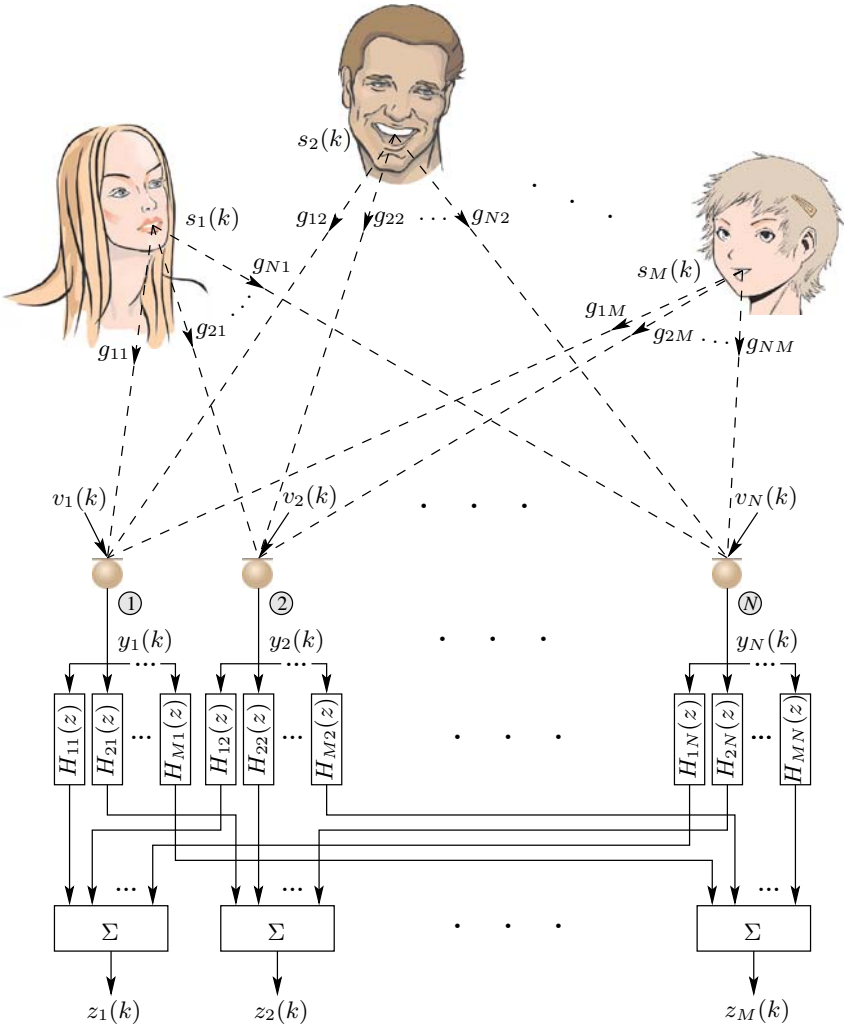


Fig. 8.1. Illustration of source separation and speech dereverberation.

where

$$\mathbf{z}_{\mathrm{a}}(k) = \left[z_1(k)\ z_2(k) \cdots z_M(k) \right]^T,$$

$$\mathbf{H} = \begin{bmatrix} \mathbf{h}_{11}^T & \mathbf{h}_{12}^T & \cdots & \mathbf{h}_{1N}^T \\ \mathbf{h}_{21}^T & \mathbf{h}_{22}^T & \cdots & \mathbf{h}_{2N}^T \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{h}_{M1}^T & \mathbf{h}_{M2}^T & \cdots & \mathbf{h}_{MN}^T \end{bmatrix}_{M \times NL_h},$$

$$\mathbf{h}_{mn} = \left[h_{mn,0}\ h_{mn,1} \cdots h_{mn,L_h-1} \right]^T,$$

$$
\begin{aligned}
&m = 1, 2, \ldots, M, \quad n = 1, 2, \ldots, N, \\
\mathbf{G} &= \begin{bmatrix} \mathbf{G}_{11} & \mathbf{G}_{12} & \cdots & \mathbf{G}_{1M} \\ \mathbf{G}_{21} & \mathbf{G}_{22} & \cdots & \mathbf{G}_{2M} \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{G}_{N1} & \mathbf{G}_{N2} & \cdots & \mathbf{G}_{NM} \end{bmatrix}_{NL_h \times ML}, \\
\mathbf{G}_{nm} &= \begin{bmatrix} g_{nm,0} & \cdots & g_{nm,L_g-1} & 0 & \cdots & 0 \\ 0 & g_{nm,0} & \cdots & g_{nm,L_g-1} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & 0 & g_{nm,0} & \cdots & g_{nm,L_g-1} \end{bmatrix}_{L_h \times L}, \\
&n = 1, 2, \ldots, N, \quad m = 1, 2, \ldots, M, \\
\mathbf{s}_{ML}(k) &= \left[\mathbf{s}_{L,1}^T(k) \; \mathbf{s}_{L,2}^T(k) \cdots \mathbf{s}_{L,M}^T(k) \right]^T, \\
\mathbf{s}_{L,m}(k) &= \left[s_m(k) \; s_m(k-1) \cdots s_m(k-L+1) \right]^T, \quad m = 1, 2, \ldots, M,
\end{aligned}
$$

L_g is the length of the longest channel impulse response in the acoustic MIMO system, L_h is the length of the separation-and-dereverberation filters, and $L = L_g + L_h - 1$.

Since we aim to make

$$
z_m(k) = s_m(k - \tau_m), \quad m = 1, 2, \ldots, M, \tag{8.3}
$$

where τ_m is a constant delay, the conditions for separation and dereverberation are deduced as

$$
\mathbf{HG} = \mathbf{U} = \begin{bmatrix} \mathbf{u}_{11}^T & \mathbf{0}_{L\times 1}^T & \cdots & \mathbf{0}_{L\times 1}^T \\ \mathbf{0}_{L\times 1}^T & \mathbf{u}_{22}^T & \cdots & \mathbf{0}_{L\times 1}^T \\ \vdots & \vdots & \ddots & \vdots \\ \mathbf{0}_{L\times 1}^T & \mathbf{0}_{L\times 1}^T & \cdots & \mathbf{u}_{MM}^T \end{bmatrix}, \tag{8.4}
$$

where $\mathbf{u}_{mm} = \left[0 \cdots 0 \; 1 \; 0 \cdots 0 \right]^T$ is a vector of length L, whose τ_mth component is equal to 1.

While in Section 7.4 we have exhaustively explored all the possible cases for solving (8.4), such a one-step algorithm by direct inverse of the channel matrix $\mathbf{G}$ does not tell us too much about the interactions between separation and dereverberation. In the following sections, we will develop a procedure which shows that separation and dereverberation are separable under some conditions that are commonly met in practical acoustic MIMO systems.

Before we proceed, we want to present again the MIMO signal model in the z-transform domain as follows

$$
\mathbf{y}_{\mathrm{a}}(z) = \mathbf{G}(z)\mathbf{s}(z) + \mathbf{v}_{\mathrm{a}}(z), \tag{8.5}
$$

where

$$
\mathbf{y}_{\mathrm{a}}(z) = \left[Y_1(z) \; Y_2(z) \cdots Y_N(z) \right]^T,
$$

$$\mathbf{G}(z) = \begin{bmatrix} G_{11}(z) & G_{12}(z) & \cdots & G_{1M}(z) \\ G_{21}(z) & G_{22}(z) & \cdots & G_{2M}(z) \\ \vdots & \vdots & \ddots & \vdots \\ G_{N1}(z) & G_{N2}(z) & \cdots & G_{NM}(z) \end{bmatrix},$$

$$G_{nm}(z) = \sum_{l=0}^{L_g-1} g_{nm,l} z^{-l}, \quad n = 1, 2, \ldots, N, \ m = 1, 2, \ldots, M,$$

$$\mathbf{s}(z) = \begin{bmatrix} S_1(z) & S_2(z) & \cdots & S_M(z) \end{bmatrix}^T,$$

$$\mathbf{v}_{\mathrm{a}}(z) = \begin{bmatrix} V_1(z) & V_2(z) & \cdots & V_N(z) \end{bmatrix}^T.$$

As the reader will see, this z-domain expression is more extensively used in this chapter.

8.3 Source Separation

In this section, we intend to show that interference from competing sources and reverberation can be separated from the microphone outputs. We begin the development with the example of a simple 2×3 MIMO system and then extend it to the more general case for $M \times N$ systems.

8.3.1 2×3 MIMO System

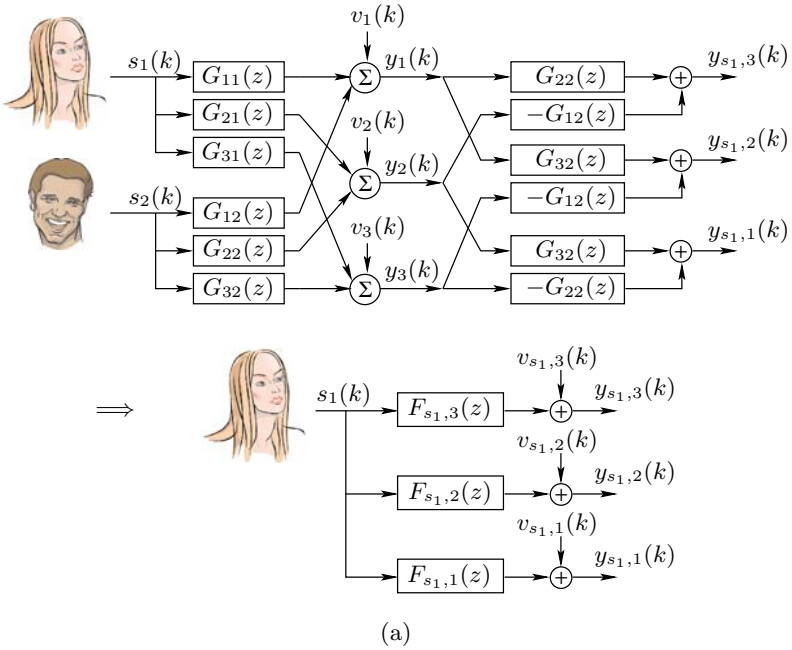
For a 2×3 system, the co-channel interference (CCI) due to the simultaneous existence of two competing sources can be cancelled by using two microphone outputs at a time. For instance, we can remove the interference in $Y_1(z)$ and $Y_2(z)$ caused by $S_2(z)$ (from the perspective of the first source) as follows:

$$\begin{aligned} & Y_1(z)G_{22}(z) - Y_2(z)G_{12}(z) \\ & = \left[G_{11}(z)G_{22}(z) - G_{21}(z)G_{12}(z)\right] S_1(z) + \\ & \quad \left[G_{22}(z)V_1(z) - G_{12}(z)V_2(z)\right]. \end{aligned} \tag{8.6}$$

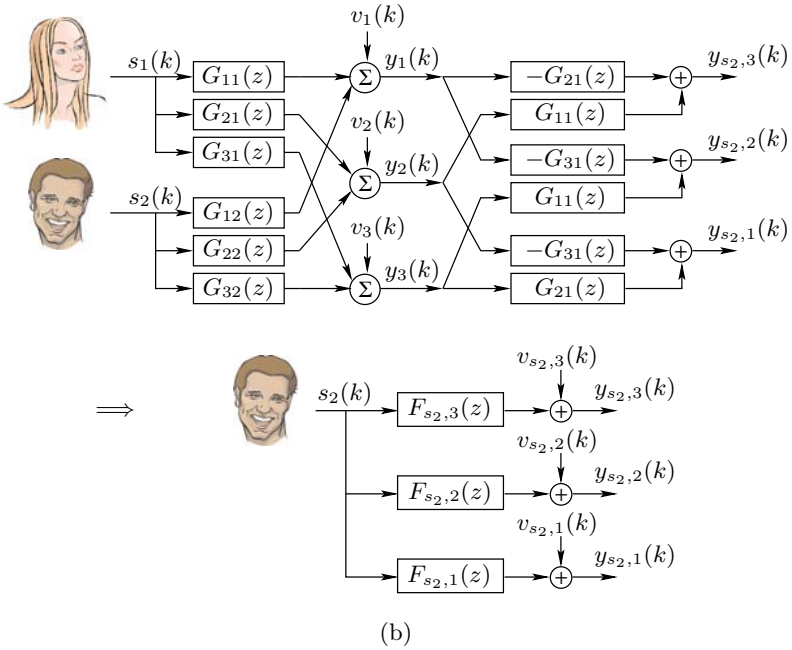
Similarly, the interference caused by $S_1(z)$ (from the perspective of the second source) in these two outputs can also be cancelled. Therefore, by selecting different pairs from the three microphone outputs, we can obtain 6 CCI-free signals and then can construct two separate 1×3 SIMO systems with $s_1(k)$ and $s_2(k)$ being their inputs, respectively. This procedure is visualized in Fig. 8.2 and will be presented in a more systematic way as follows.

Let's consider the following equation:

$$\begin{aligned} Y_{s_1,p}(z) &= H_{s_1,p1}(z)Y_1(z) + H_{s_1,p2}(z)Y_2(z) + H_{s_1,p3}(z)Y_3(z) \\ &= \sum_{q=1}^{3} H_{s_1,pq}(z)Y_q(z), \quad p = 1, 2, 3, \end{aligned} \tag{8.7}$$



(a)



(b)

Fig. 8.2. Illustration of the conversion from a 2×3 MIMO system to two CCI-free SIMO systems with respect to (a) $s_1(k)$ and (b) $s_2(k)$.

where $H_{s_1,pp}(z) = 0$, $\forall p$. This means that (8.7) considers only two microphone outputs for each p. The objective is to find the polynomials $H_{s_1,pq}(z)$, $p, q = 1, 2, 3$, $p \neq q$, in such a way that

$$Y_{s_1,p}(z) = F_{s_1,p}(z)S_1(z) + V_{s_1,p}(z), \quad p = 1, 2, 3, \tag{8.8}$$

which represents a SIMO system where $s_1(k)$ is the source signal, $y_{s_1,p}(k)$ ($p = 1, 2, 3$) are the outputs, $f_{s_1,p}$ are the corresponding channel impulse responses, and $v_{s_1,p}$ is the noise at the pth output. Substituting (8.5) in (8.7) for $Y_q(z)$, we deduce that

$$\begin{aligned} Y_{s_1,1}(z) = {} & [H_{s_1,12}(z)G_{21}(z) + H_{s_1,13}(z)G_{31}(z)]\, S_1(z) + \\ & [H_{s_1,12}(z)G_{22}(z) + H_{s_1,13}(z)G_{32}(z)]\, S_2(z) + \\ & H_{s_1,12}(z)V_2(z) + H_{s_1,13}(z)V_3(z), \end{aligned} \tag{8.9}$$

$$\begin{aligned} Y_{s_1,2}(z) = {} & [H_{s_1,21}(z)G_{11}(z) + H_{s_1,23}(z)G_{31}(z)]\, S_1(z) + \\ & [H_{s_1,21}(z)G_{12}(z) + H_{s_1,23}(z)G_{32}(z)]\, S_2(z) + \\ & H_{s_1,21}(z)V_1(z) + H_{s_1,23}(z)V_3(z), \end{aligned} \tag{8.10}$$

$$\begin{aligned} Y_{s_1,3}(z) = {} & [H_{s_1,31}(z)G_{11}(z) + H_{s_1,32}(z)G_{21}(z)]\, S_1(z) + \\ & [H_{s_1,31}(z)G_{12}(z) + H_{s_1,32}(z)G_{22}(z)]\, S_2(z) + \\ & H_{s_1,31}(z)V_1(z) + H_{s_1,32}(z)V_2(z). \end{aligned} \tag{8.11}$$

As shown in Fig. 8.2, one possibility is to choose

$$\begin{array}{llll} H_{s_1,12}(z) = G_{32}(z), & H_{s_1,13}(z) = -G_{22}(z), \\ H_{s_1,21}(z) = G_{32}(z), & H_{s_1,23}(z) = -G_{12}(z), \\ H_{s_1,31}(z) = G_{22}(z), & H_{s_1,32}(z) = -G_{12}(z). \end{array} \tag{8.12}$$

In this case, we find that

$$\begin{aligned} F_{s_1,1}(z) &= G_{32}(z)G_{21}(z) - G_{22}(z)G_{31}(z), \\ F_{s_1,2}(z) &= G_{32}(z)G_{11}(z) - G_{12}(z)G_{31}(z), \\ F_{s_1,3}(z) &= G_{22}(z)G_{11}(z) - G_{12}(z)G_{21}(z), \end{aligned} \tag{8.13}$$

and

$$\begin{aligned} V_{s_1,1}(z) &= G_{32}(z)V_2(z) - G_{22}(z)V_3(z), \\ V_{s_1,2}(z) &= G_{32}(z)V_1(z) - G_{12}(z)V_3(z), \\ V_{s_1,3}(z) &= G_{22}(z)V_1(z) - G_{12}(z)V_2(z). \end{aligned} \tag{8.14}$$

Since $\deg[G_{nm}(z)] = L_g - 1$, where $\deg[\cdot]$ is the degree of a polynomial, we deduce that $\deg[F_{s_1,p}(z)] \leq 2L_g - 2$. We can see from (8.13) that the polynomials $F_{s_1,1}(z)$, $F_{s_1,2}(z)$, and $F_{s_1,3}(z)$ share common zeros if $G_{12}(z)$, $G_{22}(z)$, and $G_{32}(z)$, or if $G_{11}(z)$, $G_{21}(z)$, and $G_{31}(z)$, share common zeros.

Now suppose that

$$C_2(z) = \gcd\left[G_{12}(z), G_{22}(z), G_{32}(z)\right], \tag{8.15}$$

where gcd[·] denotes the greatest common divisor of the polynomials involved. We have

$$G_{n2}(z) = C_2(z)G'_{n2}(z), \quad n = 1, 2, 3. \tag{8.16}$$

It is clear that the signal $S_2(z)$ in (8.7) can be canceled by using the polynomials $G'_{n2}(z)$ [instead of $G_{n2}(z)$ as given in (8.12)], so that the SIMO system represented by (8.8) will change to

$$Y'_{s_1,p}(z) = F'_{s_1,p}(z)S_1(z) + V'_{s_1,p}(z), \quad p = 1, 2, 3, \tag{8.17}$$

where

$$\begin{aligned} F'_{s_1,p}(z)C_2(z) &= F_{s_1,p}(z), \\ V'_{s_1,p}(z)C_2(z) &= V_{s_1,p}(z). \end{aligned}$$

It is worth noticing that $\deg\left[F'_{s_1,p}(z)\right] \leq \deg\left[F_{s_1,p}(z)\right]$ and that the polynomials $F'_{s_1,1}(z)$, $F'_{s_1,2}(z)$, and $F'_{s_1,3}(z)$ share common zeros if and only if $G_{11}(z)$, $G_{21}(z)$, and $G_{31}(z)$ share common zeros.

The second SIMO system corresponding to the second source $S_2(z)$ can be derived in a similar way. We can find the output signals:

$$Y_{s_2,p}(z) = F_{s_2,p}(z)S_2(z) + V_{s_2,p}(z), \quad p = 1, 2, 3, \tag{8.18}$$

by enforcing $F_{s_2,p}(z) = F_{s_1,p}(z)$ $(p = 1, 2, 3)$, which leads to

$$\begin{aligned} V_{s_2,1}(z) &= -G_{31}(z)V_2(z) + G_{21}(z)V_3(z), \\ V_{s_2,2}(z) &= -G_{31}(z)V_1(z) + G_{11}(z)V_3(z), \\ V_{s_2,3}(z) &= -G_{21}(z)V_1(z) + G_{11}(z)V_2(z). \end{aligned}$$

This means that the two separated SIMO systems [for s_1 and s_2, represented by equations (8.8) and (8.18)] have identical channels but different additive noise at the outputs.

Now let's see what we can do if $G_{n1}(z)$ $(n = 1, 2, 3)$ share common zeros. Suppose that $C_1(z)$ is the greatest common divisor of $G_{11}(z)$, $G_{21}(z)$, and $G_{31}(z)$. Then we have

$$G_{n1}(z) = C_1(z)G'_{n1}(z), \quad n = 1, 2, 3, \tag{8.19}$$

and the SIMO system of (8.18) becomes

$$Y'_{s_2,p}(z) = F'_{s_2,p}(z)S_2(z) + V'_{s_2,p}(z), \quad p = 1, 2, 3, \tag{8.20}$$

where

$$F'_{s_2,p}(z)C_1(z) = F_{s_2,p}(z),$$
$$V'_{s_2,p}(z)C_1(z) = V_{s_2,p}(z).$$

We see that

$$\begin{aligned}\gcd\left[F'_{s_2,1}(z), F'_{s_2,2}(z), F'_{s_2,3}(z)\right] &= \gcd\left[G_{12}(z), G_{22}(z), G_{32}(z)\right] \\ &= C_2(z),\end{aligned} \tag{8.21}$$

and in general $F'_{s_1,p}(z) \neq F'_{s_2,p}(z)$.

8.3.2 $M \times N$ MIMO System

The approach to separating signals coming from different competing sources that was explained in the previous subsection using a simple example will be generalized here to $M \times N$ MIMO systems with $M > 2$ and $M < N$. We begin with denoting $C_m(z)$ as the greatest common divisor of $G_{1m}(z), G_{2m}(z), \cdots, G_{Nm}(z)$ $(m = 1, 2, \ldots, M)$, i.e.,

$$C_m(z) = \gcd\left[G_{1m}(z), G_{2m}(z), \cdots, G_{Nm}(z)\right], \quad m = 1, 2, \ldots, M. \tag{8.22}$$

Then, $G_{nm}(z) = C_m(z)G'_{nm}(z)$ and the channel matrix $\mathbf{G}(z)$ can be rewritten as

$$\mathbf{G}(z) = \mathbf{G}'(z)\mathbf{C}(z), \tag{8.23}$$

where $\mathbf{G}'(z)$ is an $N \times M$ matrix containing the elements $G'_{nm}(z)$ and $\mathbf{C}(z)$ is an $M \times M$ diagonal matrix with $C_m(z)$ as its nonzero, diagonal components.

Let us pick up M from N microphone outputs and we have

$$P = C_N^M = \frac{\prod_{i=N-M+1}^{N} i}{\prod_{i=1}^{M} i} \tag{8.24}$$

different ways of doing this. For the pth $(p = 1, 2, \ldots, P)$ combination, we denote the index of the M selected output signals as p_m, $m = 1, 2, \ldots, M$, which together with the M inputs form an $M \times M$ MIMO sub-system.

Consider the following equations:

$$\mathbf{y}_{s,p}(z) = \mathbf{H}_{s,p}(z)\mathbf{y}_{\mathrm{a},p}(z), \quad p = 1, 2, \ldots, P, \tag{8.25}$$

where

$$\begin{aligned}\mathbf{y}_{s,p}(z) &= \left[Y_{s_1,p}(z)\ Y_{s_2,p}(z) \cdots Y_{s_M,p}(z)\right]^T, \\ \mathbf{H}_{s,p}(z) &= \begin{bmatrix} H_{s_1,p1}(z) & H_{s_1,p2}(z) & \cdots & H_{s_1,pM}(z) \\ H_{s_2,p1}(z) & H_{s_2,p2}(z) & \cdots & H_{s_2,pM}(z) \\ \vdots & \vdots & \vdots & \vdots \\ H_{s_M,p1}(z) & H_{s_M,p2}(z) & \cdots & H_{s_M,pM}(z) \end{bmatrix}, \\ \mathbf{y}_{\mathrm{a},p}(z) &= \left[Y_{p_1}(z)\ Y_{p_2}(z) \cdots Y_{p_M}(z)\right]^T.\end{aligned}$$

Let $\mathbf{G}_p(z)$ be the $M \times M$ matrix obtained from the system's channel matrix $\mathbf{G}(z)$ by keeping its rows corresponding to the M selected output signals. Then similar to (8.5), we have

$$\mathbf{y}_{\mathrm{a},p}(z) = \mathbf{G}_p(z)\mathbf{s}(z) + \mathbf{v}_{\mathrm{a},p}(z), \tag{8.26}$$

where

$$\mathbf{v}_{\mathrm{a},p}(z) = \left[V_{p_1}(z)\ V_{p_2}(z) \cdots V_{p_M}(z) \right]^T.$$

Substituting (8.26) into (8.25) yields

$$\mathbf{y}_{s,p}(z) = \mathbf{H}_{s,p}(z)\mathbf{G}_p(z)\mathbf{s}(z) + \mathbf{H}_{s,p}(z)\mathbf{v}_{\mathrm{a},p}(z). \tag{8.27}$$

In order to remove the CCI, the objective here is to find the matrix $\mathbf{H}_{s,p}(z)$ whose components are linear combinations of $G_{nm}(z)$ such that the product $\mathbf{H}_{s,p}(z)\mathbf{G}_p(z)$ would be a diagonal matrix. Consequently, we have

$$\begin{aligned} Y_{s_m,p}(z) = F_{s_m,p}(z)S_m(z) + V_{s_m,p}(z), \\ m = 1, 2, \ldots, M, \quad p = 1, 2, \ldots, P. \end{aligned} \tag{8.28}$$

If $\mathbf{C}_p(z)$ [obtained from $\mathbf{C}(z)$ in a similar way as $\mathbf{G}_p(z)$ is constructed] is not equal to the identity matrix, then $\mathbf{G}_p(z) = \mathbf{G}'_p(z)\mathbf{C}_p(z)$, where $\mathbf{G}'_p(z)$ has full column normal rank[1] (i.e. nrank $\left[\mathbf{G}'_p(z)\right] = M$, see [214] for a definition of normal rank), as we assume for separability of CCI and reverberation in a MIMO system. Thereafter, the CCI-free signals are determined as

$$\mathbf{y}'_{s,p}(z) = \mathbf{H}'_{s,p}(z)\mathbf{G}'_p(z)\mathbf{C}_p(z)\mathbf{s}(z) + \mathbf{H}'_{s,p}(z)\mathbf{v}_{\mathrm{a},p}(z), \tag{8.29}$$

and

$$Y'_{s_m,p}(z) = F'_{s_m,p}(z)S_m(z) + V'_{s_m,p}(z). \tag{8.30}$$

Obviously a good choice for $\mathbf{H}'_{s,p}(z)$ to make the product $\mathbf{H}'_{s,p}(z)\mathbf{G}'_p(z)$ a diagonal matrix is the adjoint of matrix $\mathbf{G}'_p(z)$, i.e., the (i,j)-th element of $\mathbf{H}'_{s,p}(z)$ is the (j,i)-th cofactor of $\mathbf{G}'_p(z)$. Consequently, the polynomial $F'_{s_m,p}(z)$ would be the determinant of $\mathbf{G}'_p(z)$. Since $\mathbf{G}'_p(z)$ has full column normal rank, its determinant is not equal to zero and the polynomial $F'_{s_m,p}(z)$ is not trivial.

Since

$$F'_{s_m,p}(z) = \sum_{q=1}^{M} H'_{s_m,pq}(z)G_{p_q m}(z) \tag{8.31}$$

and $H'_{s_m,pq}(z)$ $(q = 1, 2, \ldots, M)$ are co-prime, the polynomials $F'_{s_m,p}(z)$ $(p = 1, 2, \ldots, P)$ share common zeros if and only if the polynomials $G_{nm}(z)$ $(n = 1, 2, \ldots, N)$ share common zeros. Therefore, if the channels with respect to

[1] For a square matrix $M \times M$, the normal rank is full if and only if the determinant, which is a polynomial in z, is not identically zero for all z. In this case, the rank is less than M only at a finite number of points in the z plane.

any one input are co-prime for an $M \times N$ MIMO system, we can convert it into M CCI-free SIMO systems whose P channels are also co-prime, i.e., their channel matrices are irreducible.

Also, it can easily be checked that $\deg\left[F'_{s_m,p}(z)\right] \leq M(L_g - 1)$. As a result, the length of the FIR filter $f'_{s_m,p}$ would be

$$L_f \leq M(L_g - 1) + 1. \tag{8.32}$$

Before we finish this section, we would like to comment in a little bit more detail on the condition for separability of the interference caused by competing sources and the interference caused by reverberation in a MIMO system. For an $M \times M$ MIMO system or an $M \times M$ subsystem of a larger $M \times N$ $(M < N)$ MIMO system, it is now clear that the *reduced* channel matrix $\mathbf{G}'_p(z)$ needs to have full column normal rank such that the CCI and reverberation are separable. But what happens and why is the CCI unable to be separated from the reverberation if $\mathbf{G}'_p(z)$ does not have full column normal rank?

Let's first examine a 2×2 system and its reduced channel matrix is given by

$$\mathbf{G}'_p(z) = \begin{bmatrix} G'_{p,11}(z) & G'_{p,12}(z) \\ G'_{p,21}(z) & G'_{p,22}(z) \end{bmatrix}. \tag{8.33}$$

If $\mathbf{G}'_p(z)$ does not have full column normal rank, then there exist two non-zero polynomials $A_1(z)$ and $A_2(z)$ such that

$$\begin{bmatrix} G'_{p,11}(z) \\ G'_{p,21}(z) \end{bmatrix} A_1(z) = \begin{bmatrix} G'_{p,12}(z) \\ G'_{p,22}(z) \end{bmatrix} A_2(z), \tag{8.34}$$

or equivalently

$$\mathbf{G}'_p(z) \begin{bmatrix} A_1(z) \\ -A_2(z) \end{bmatrix} = \mathbf{0}. \tag{8.35}$$

As a result, in the absence of noise, we know that

$$Y_{p,1}(z) = -\frac{A_2(z)}{A_1(z)} Y_{p,2}(z), \tag{8.36}$$

which implies that the MISO systems corresponding to the two outputs are identical up to a constant filter. Therefore the 2×2 MIMO is reduced to a 2×1 MISO system where the number of inputs is greater than the number of outputs and the CCI cannot be separated from the reverberation.

For an $M \times M$ MIMO system with $M > 2$, if $\mathbf{G}'_p(z)$ does not have full column normal rank, then there are only $\left\{\text{nrank}\left[\mathbf{G}'_p(z)\right]\right\}$ independent MISO systems and the other $\left\{M - \text{nrank}\left[\mathbf{G}'_p(z)\right]\right\}$ MISO systems can be reduced. This indicates that the MIMO system has essentially more inputs than outputs and the CCI cannot be separated from the reverberation.

Extracting $C_m(z)$ $(m = 1, 2, \ldots, M)$ from the mth column of $\mathbf{G}(z)$ (if necessary) is intended to reduce the SIMO system with respect to each input.

The purpose of examining the column normal rank of $\mathbf{G}'_p(z)$ is to check the dependency of the MISO systems associated with the outputs. For the $M \times N$ MIMO systems ($M < N$), the column normal rank of $\mathbf{G}'(z)$ actually indicates how many MISO subsystems are independent. As long as nrank $[\mathbf{G}'(z)] \geq M$, there exists at least one $M \times M$ subsystem whose M MISO systems are all independent and whose CCI and reverberation are separable. Therefore the condition for separability of CCI and reverberation in an $M \times N$ MIMO system is nothing more than to require that there are more effective outputs than inputs. This condition is quite commonly met in practice, particularly in acoustic systems.

8.4 Speech Dereverberation

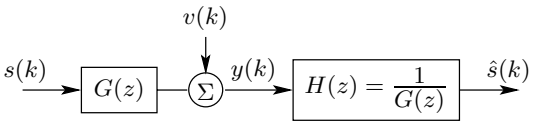
Reverberation is one of the two major causes (the other is noise) of speech degradation. It leads to temporal and spectral smearing, which would distort both the envelope and fine structure of a speech signal. As a result, speech becomes difficult to be understood in the presence of room reverberation, especially for hearing-impaired and elderly people [170] and for automatic speech recognition systems [143], [193]. This gives rise to a strong need for effective speech dereverberation algorithms in speech processing and speech communication systems.

Using the technique developed in the previous section, we can separate the co-channel interference and reverberation in an acoustic MIMO system. While the outputs are free of co-channel interference, they sound probably more reverberant since the equivalent channel impulses are prolonged. Consequently a second-step processing of dereverberation is not simply preferable, but rather imperative.

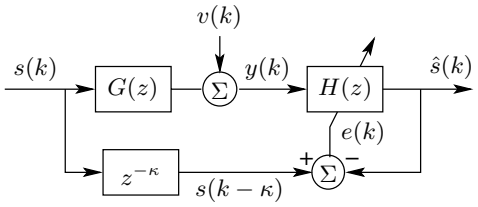
According to [126], speech dereverberation methods can be classified into the following three groups: speech-source-model-based dereverberation, separation of speech and reverberation via homomorphic transformation, and speech dereverberation by channel inversion and equalization. In the context of this chapter, while the first two classes of speech dereverberation methods can also be applied, we think that the third class is a more relevant technique. Therefore we choose to discuss only channel inverse and equalization methods for speech dereverberation in this section. Three widely used algorithms will be developed, namely the direct inverse (also called zero forcing) method, the minimum mean square error (MMSE) or least-squares (LS) method, and the multichannel inverse theorem (MINT) method. The first two methods work for SISO systems and the third for SIMO systems as illustrated in Fig. 8.3.

8.4.1 Direct Inverse

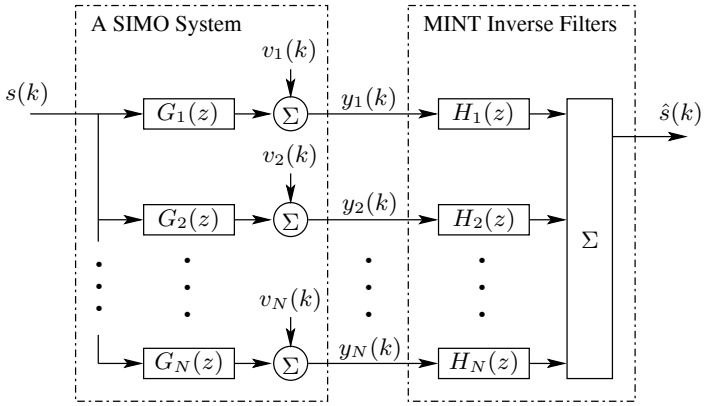
Among all existing channel inversion methods, the most straightforward is the direct inverse method. This method assumes that the acoustic channel



(a)



(b)



(c)

Fig. 8.3. Illustration of three widely-used channel equalization approaches to speech dereverberation: (a) direct inverse (or zero-forcing), (b) minimum mean square error (or least-squares), and (c) the MINT method.

impulse response is known or has already been estimated. Then as shown in Fig. 8.3(a), the equalizer filter is determined by inverting the channel transfer function $G(z)$ which is the z-transform of the channel impulse response:

$$H(z) = \frac{1}{G(z)}. \tag{8.37}$$

In practice, the inverse filter h needs to be stable and causal. It is well known that the poles of a stable, causal, and rational system must be inside the unit circle in the z-plane. As a result, a stable, causal system has a stable and

causal inverse only if both its poles and zeros are inside the unit circle. Such a system is commonly referred to as a *minimum-phase* system [177]. Although many systems are minimum phase, room acoustic impulse responses are unfortunately almost never minimum phase [171]. Consequently, while a stable inverse filter still can be found by using an all-pass filter, the inverse filter will be IIR, which is noncausal and has a long delay. In addition, inverting a transfer function is sensitive to estimation errors in the channel impulse response, particularly at those frequencies where the channel transfer function has a small amplitude. These drawbacks make direct-inverse equalizers impracticable for real-time speech dereverberation systems.

8.4.2 Minimum Mean-Square Error and Least-Squares Methods

If a reference source signal rather than an estimate of the acoustic channel impulse response is available, we can directly apply a linear equalizer to the microphone signal and adjust the equalizer coefficients such that the output can be as close to the reference as possible, as shown in Fig. 8.3(b). The error signal is defined as

$$\begin{aligned} e(k) &= s(k-\tau) - \hat{s}(k) \\ &= s(k-\tau) - h * y(k), \end{aligned} \tag{8.38}$$

where τ is the decision delay for the equalizer. Then the equalization filter h is determined as the one that either minimizes the mean square error or yields the least squares of the error signal:

$$\hat{h}_{\mathrm{MMSE}} = \arg\min_{h} E\left\{e^2(k)\right\}, \tag{8.39}$$

$$\hat{h}_{\mathrm{LS}} = \arg\min_{h} \sum_{k=0}^{K-1} e^2(k), \tag{8.40}$$

where K is the number of observed data samples. This is a typical problem in estimation theory. The solution can be found with well-known adaptive or recursive algorithms.

For minimum-phase single-channel systems, it can be shown that the MMSE/LS equalizer is the same as the direct-inverse or zero-forcing equalizer. But for non-minimum-phase acoustic systems, the MMSE/LS method essentially equalizes the channel by inverting only those components whose zeros are inside the unit circle [166]. In addition, it is clear that, for the MMSE/LS equalizer, a reference signal needs to be accessible. However, although the MMSE/LS method has these limitations, it is quite useful in practice and has been successfully applied to many speech dereverberation systems.

8.4.3 MINT Method

For a SIMO system, let's consider the polynomials $H_n(z)$ $(n = 1, 2, \ldots, N)$ and the following equation:

$$\begin{aligned}\hat{S}(z) &= \sum_{n=1}^{N} H_n(z) Y_n(z) \\ &= \left[\sum_{n=1}^{N} H_n(z) G_n(z)\right] S(z) + \sum_{n=1}^{N} H_n(z) V_n(z). \end{aligned} \tag{8.41}$$

The polynomials $H_n(z)$ should be found in such a way that $\hat{S}(z) = S(z)$ in the absence of noise by using the Bezout theorem which is mathematically expressed as follows:

$$\begin{aligned}&\gcd\left[G_1(z), G_2(z), \cdots, G_N(z)\right] = 1 \\ &\Leftrightarrow \exists\ H_1(z), H_2(z), \cdots, H_N(z)\ :\ \sum_{n=1}^{N} H_n(z) G_n(z) = 1. \end{aligned} \tag{8.42}$$

In other words, as long as the channel impulse responses g_n are co-prime (even though they may not be minimum phase), i.e., the SIMO system is irreducible, there exists a group of h filters to completely remove the reverberations and perfectly recover the source signal. The idea of using the Bezout theorem for equalizing a SIMO system was first proposed in [166] in the context of room acoustics, where the principle is more widely referred to as the MINT theory.

If the channels of the SIMO system share common zeros, i.e.,

$$C(z) = \gcd\left[G_1(z), G_2(z), \cdots, G_N(z)\right] \neq 1, \tag{8.43}$$

then we have

$$G_n(z) = C(z) G'_n(z), \quad n = 1, 2, \cdots, N, \tag{8.44}$$

and the polynomials $H_n(z)$ can be found such that

$$\sum_{n=1}^{N} H_n(z) G'_n(z) = 1. \tag{8.45}$$

In this case, (8.41) becomes

$$\hat{S}(z) = C(z) S(z) + \sum_{n=1}^{N} H_n(z) V_n(z). \tag{8.46}$$

We see that by using the Bezout theorem, the *reducible* SIMO system can be equalized up to the polynomial $C(z)$. So when there are common zeros, the MINT equalizer can only partially suppress the reverberations. For more complete cancellation of the room reverberations, we have to combat the effect of $C(z)$ using either the direct inverse or MMSE/LS methods, which depends on whether $C(z)$ is a minimum phase filter.

To find the MINT equalization filters, we write the Bezout equation (8.42) in the time domain as

$$\mathbf{G}^T\mathbf{h} = \sum_{n=1}^{N} \mathbf{G}_n^T \mathbf{h}_n = \mathbf{u}_1, \tag{8.47}$$

where

$$\begin{aligned}
\mathbf{G} &= \left[\, \mathbf{G}_1^T \; \mathbf{G}_2^T \; \cdots \; \mathbf{G}_N^T \,\right]^T, \\
\mathbf{h} &= \left[\, \mathbf{h}_1^T \; \mathbf{h}_2^T \; \cdots \; \mathbf{h}_N^T \,\right]^T, \\
\mathbf{h}_n &= \left[\, h_{n,0} \; h_{n,1} \; \cdots \; h_{n,L_h-1} \,\right]^T,
\end{aligned}$$

L_h is the length of the FIR filter h_n,

$$\mathbf{G}_n = \begin{bmatrix}
g_{n,0} & \cdots & g_{n,L_g-1} & 0 & \cdots & 0 \\
0 & g_{n,0} & \cdots & g_{n,L_g-1} & \cdots & 0 \\
\vdots & \ddots & \ddots & \ddots & \ddots & \vdots \\
0 & \cdots & 0 & g_{n,0} & \cdots & g_{n,L_g-1}
\end{bmatrix}_{L_h \times L}, \quad n = 1, 2, \ldots, N,$$

is a Sylvester matrix of size $L_h \times L$, with $L = L_g + L_h - 1$, and $\mathbf{u}_1 = \left[\, 1 \; 0 \; \cdots \; 0 \,\right]^T$ is a vector of length L.

In order to have a unique solution for (8.47), L_h must be chosen in such a way that $\mathbf{G}$ is a square matrix. In this case, we have

$$L_h = \frac{L_g - 1}{N - 1}. \tag{8.48}$$

However, this may not be practical since $(L_g - 1)/(N - 1)$ is not necessarily always an integer. Therefore, a larger L_h is usually chosen and solve (8.47) for $\mathbf{h}$ in the least squares sense as follows:

$$\hat{\mathbf{h}}_{\text{MINT}} = \mathbf{G}^{T^\dagger}\mathbf{u}_1, \tag{8.49}$$

where

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

is the pseudo-inverse of the matrix $\mathbf{G}^T$.

If a decision delay τ is taken into account, then the equalization filters turn out to be

$$\hat{\mathbf{h}}_{\text{MINT}} = \mathbf{G}^{T^\dagger}\mathbf{u}_\tau, \tag{8.50}$$

where $\mathbf{u}_\tau = \left[\, 0 \cdots 0 \; 1 \; 0 \cdots 0 \,\right]^T$ is a vector of length L with all elements being zeros except its τth element being one.

MINT equalization is an appealing approach to speech dereverberation. As long as the channels do not share any common zeroes, it can perfectly remove the effect of room reverberation even though acoustic impulse responses are not minimum phase. But in practice, the MINT method was found very sensitive to even small errors in the estimated channel impulse responses. Therefore, it is only useful when background noise is weak or well controlled.

All the approaches developed in this chapter can be implemented in subbands [229], [240]. This will reduced the computational load and sometimes may be even more robust to noise or estimation errors [83].

8.5 Conclusions

In this chapter, we continued to study the problem of separation and dereverberation using a microphone array and developed a two-stage approach. When there are multiple sound sources simultaneously in a reverberant environment, the outputs of the microphone array observe both co-channel interference and reverberation. In order to recover the source signals, spatio-temporal equalization needs to be performed to suppress or even cancel these interference signals. But instead of finding a solution in one step as we did in the previous chapter, we showed that co-channel interference and reverberation can be separated by converting an $M \times N$ MIMO system with $M < N$ into M SIMO systems that are free of co-channel interference. In the process of developing such a conversion technique, insight was highlighted about the interactions between co-channel interference and reverberation in acoustic MIMO systems. We also briefly reviewed traditional and emerging algorithms for speech dereverberation by channel inverse and equalization. They included the direct inverse (or zero forcing), the MMSE (or LS), and the MINT methods.

9

Direction-of-Arrival and Time-Difference-of-Arrival Estimation

9.1 Introduction

In the previous chapters we have studied how to use a microphone array to enhance a desired target signal and suppress unwanted noise and interference. Another major functionality of microphone array signal processing is the estimation of the location from which a source signal originates. Depending on the distance between the source and the array relatively to the array size, this estimation problem can be divided into two sub problems, i.e., direction-of-arrival (DOA) estimation and source localization.

The DOA estimation deals with the case where the source is in the array's far-field, as illustrated in Fig. 9.1. In this situation, the source radiates a plane wave having the waveform $s(k)$ that propagates through the non-dispersive medium-air. The normal to the wavefront makes an angle θ with the line joining the sensors in the linear array, and the signal received at each microphone is a time delayed/advanced version of the signal at a reference sensor. To see this, let us choose the first sensor in Fig. 9.1 as the reference point and denote the spacing between the two sensors as d. The signal at the second sensor is delayed by the time required for the plane wave to propagate through $d\cos\theta$. Therefore, the time difference (time delay) between the two sensors is given by $\tau_{12} = d\cos\theta/c$, where c is the sound velocity in air. If the angle ranges between 0° and 180° and if τ_{12} is known then θ is uniquely determined, and *vice versa*. Therefore, estimating the incident angle θ is essentially identical to estimating the time difference τ_{12}. In other words, the DOA estimation problem is the same as the so-called time-difference-of-arrival (TDOA) estimation problem in the far-field case.

Although the incident angle can be estimated with the use of two or more sensors, the range between the sound source and the microphone array is difficult (if not impossible) to determine if the source is in the array's far-field. However, if the source is located in the near-field, as illustrated in Fig. 9.2, it is now possible to estimate not only the angle from which the wave ray reaches each sensor but also the distance between the source and each microphone.

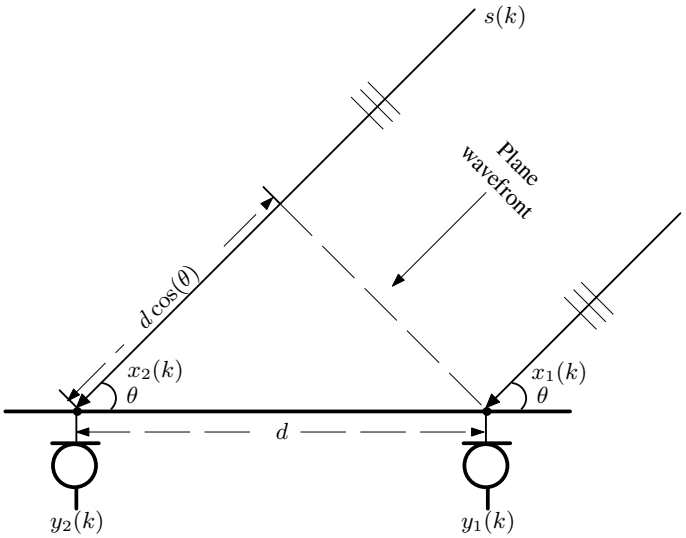


Fig. 9.1. Illustration of the DOA estimation problem in 2-dimensional space with two identical microphones: the source $s(k)$ is located in the far-field, the incident angle is θ, and the spacing between the two sensors is d.

To see this, let us consider the simple example shown in Fig. 9.2. Again, we choose the first microphone as the reference sensor. Let θ_n and r_n denote, respectively, the incident angle and the distance between the sound source and microphone n, $n = 1, 2, 3$. The TDOA between the second and first sensors is given by

$$\tau_{12} = \frac{r_2 - r_1}{c}, \tag{9.1}$$

and the TDOA between the third and first sensors is

$$\tau_{13} = \frac{r_3 - r_1}{c}. \tag{9.2}$$

Applying the cosine rule, we obtain

$$r_2^2 = r_1^2 + d^2 + 2r_1 d\cos(\theta_1) \tag{9.3}$$

and

$$r_3^2 = r_1^2 + 4d^2 + 4r_1 d\cos(\theta_1). \tag{9.4}$$

For a practical array system, the spacing d can always be measured once the array geometry is fixed. If τ_{12} and τ_{13} are available then we can calculate all the unknown parameters θ_1, r_1, r_2, and r_3 by solving the equations from (9.1) to (9.4). Further applying the sine rule, we can obtain an estimate of θ_2

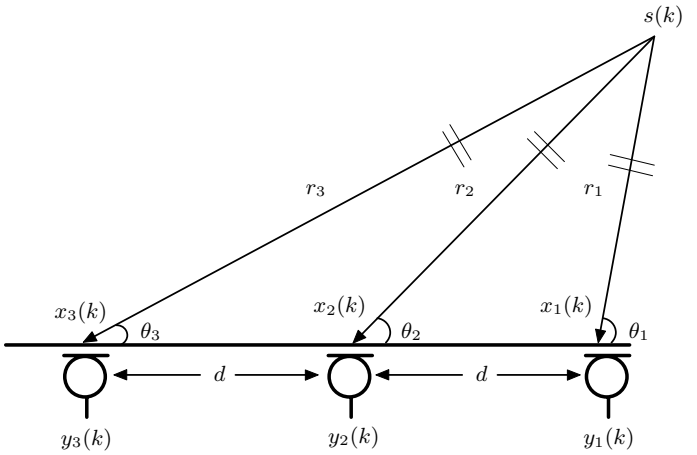


Fig. 9.2. Illustration of the source localization problem with an equispaced linear array: the source $s(k)$ is located in the near-field, and the spacing between any two neighboring sensors is d.

and θ_3. Therefore, all the information regarding the source position relatively to the array can be determined using the triangulation rule once the TDOA information is available. This basic triangulation process forms the foundation for most of the source-localization techniques, even though many algorithms may formulate and solve the problem from a different theoretical perspective [7], [26], [71], [74], [97], [116], [117], [188], [204], [221], [222], [223], [226].

Therefore, regardless if the source is located in the far-field or near-field, the most fundamental step in obtaining the source-origin information is the one of estimating the TDOA between different microphones. This estimation problem would be an easy task if the received signals were merely a delayed and scaled version of each other. In reality, however, the source signal is generally immersed in ambient noise since we are living in a natural environment where the existence of noise is inevitable. Furthermore, each observation signal may contain multiple attenuated and delayed replicas of the source signal due to reflections from boundaries and objects. This multipath propagation effect introduces echoes and spectral distortions into the observation signal, termed as reverberation, which severely deteriorates the source signal. In addition, the source may also move from time to time, resulting in a changing time delay. All these factors make TDOA estimation a complicated and challenging problem.

This chapter discusses the basic ideas underlying TDOA estimation. We will begin our discussion with scenarios where there is only a single source in the sound field. We will then explore what approaches can be used to improve the robustness of TDOA estimation with respect to noise and reverberation. Many fundamental ideas developed for the single-source TDOA estimation

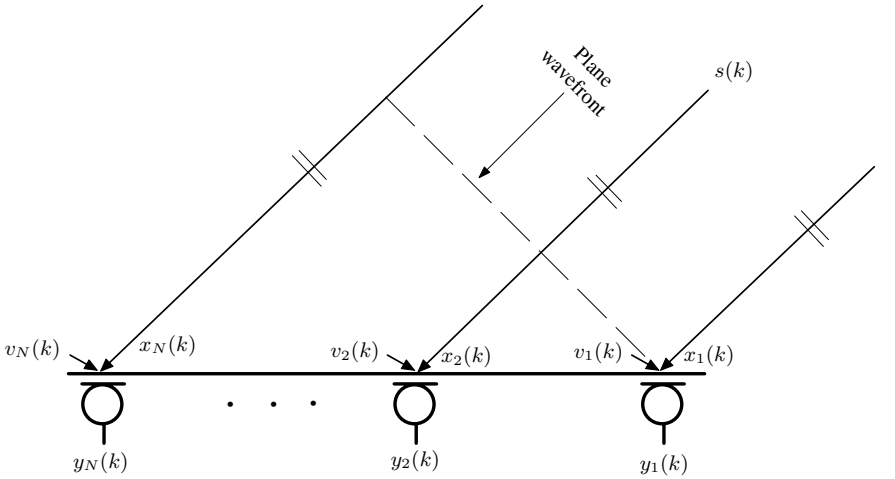


Fig. 9.3. Illustration of the ideal free-field single-source model.

can be extended to the multiple-source situation. To illustrate this, we will discuss the philosophy underlying multiple-source TDOA estimation.

9.2 Problem Formulation and Signal Models

The TDOA estimation problem is concerned with the measurement of time difference between the signals received at different microphones. Depending on the surrounding acoustic environment, we consider two situations: the free-field environment where each sensor receives only the direct-path signal, and reverberant environments where each sensor may receive a large number of reflected signals in addition to the direct path. For each situation, we differentiate the single-source case from the multiple-source scenario since the estimation principles and complexity in these two conditions may not necessarily be the same. So, in total, we consider four signal models: the single-source free-field model, the multiple-source free-field model, the single-source reverberant model, and the multiple-source reverberant model.

9.2.1 Single-Source Free-Field Model

Suppose that there is only one source in the sound field and we use an array of N microphones. In an anechoic open space as shown in Fig. 9.3, the speech source signal $s(k)$ propagates radiatively and the sound level falls off as a function of distance from the source. If we choose the first microphone as the reference point, the signal captured by the nth microphone at time k can be expressed as follows:

$$\begin{aligned} y_n(k) &= \alpha_n s\left(k - t - \tau_{n1}\right) + v_n(k) \\ &= \alpha_n s\left[k - t - \mathcal{F}_n(\tau)\right] + v_n(k) \\ &= x_n(k) + v_n(k),\ n = 1, 2, \ldots, N, \end{aligned} \tag{9.5}$$

where α_n $(n = 1, 2, \ldots, N)$, which range between 0 and 1, are the attenuation factors due to propagation effects, $s(k)$ is the unknown source signal, t is the propagation time from the unknown source to sensor 1, $v_n(k)$ is an additive noise signal at the nth sensor, which is assumed to be uncorrelated with both the source signal and the noise observed at other sensors, τ is the TDOA (also called relative delay) between sensors 1 and 2, and $\tau_{n1} = \mathcal{F}_n(\tau)$ is the TDOA between sensors 1 and n with $\mathcal{F}_1(\tau) = 0$ and $\mathcal{F}_2(\tau) = \tau$. For $n = 3, \ldots, N$, the function $\mathcal{F}_n$ depends not only on τ but also on the microphone array geometry. For example, in the far-field case (plane wave propagation), for a linear and equispaced array, we have

$$\mathcal{F}_n(\tau) = (n-1)\tau, \quad n = 2, \ldots, N, \tag{9.6}$$

and for a linear but non-equispaced array, we have

$$\mathcal{F}_n(\tau) = \frac{\sum_{i=1}^{n-1} d_i}{d_1}\tau, \quad n = 2, \ldots, N, \tag{9.7}$$

where d_i is the distance between microphones i and $i+1$ ($i = 1, \ldots, N-1$). In the near-field case, $\mathcal{F}_n$ depends also on the position of the sound source. Note that $\mathcal{F}_n(\tau)$ can be a *nonlinear* function of τ for a nonlinear array geometry, even in the far-field case (e.g., 3 equilateral sensors). In general τ is not known, but the geometry of the array is known such that the mathematical formulation of $\mathcal{F}_n(\tau)$ is well defined or given. For this model, the TDE (time-delay estimation) problem is formulated as one of determining an estimate $\hat{\tau}$ of the true time delay τ using a set of finite observation samples.

9.2.2 Multiple-Source Free-Field Model

Still in the anechoic environments, if there are multiple sources in the sound field, the signal received at the nth sensor becomes

$$\begin{aligned} y_n(k) &= \sum_{m=1}^{M} \alpha_{nm} s_m\left[k - t_m - \mathcal{F}_n(\tau_m)\right] + v_n(k) \\ &= x_n(k) + v_n(k),\ n = 1, 2, \ldots, N, \end{aligned} \tag{9.8}$$

where M is the total number of sound sources, α_{nm} $(n = 1, 2, \ldots, N,\ m = 1, 2, \ldots, M)$, are the attenuation factors due to propagation effects, $s_m(k)$ $(m = 1, 2, \ldots, M)$ are the unknown source signals, which are assumed to be mutually independent with each other, t_m is the propagation time from the unknown source m to sensor 1 (reference sensor), $v_n(k)$ is an additive noise

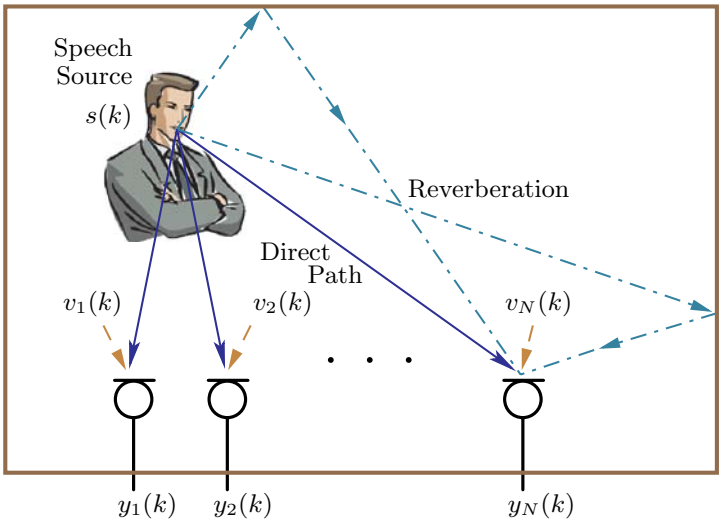


Fig. 9.4. Illustration of the single-source reverberant model.

signal at the nth sensor, which is assumed to be uncorrelated with not only all the source signals but also with the noise observed at other sensors, τ_m is the TDOA between sensors 2 and 1 due to the mth source, and $\mathcal{F}_n(\tau_m)$ is the TDOA between sensors n and 1 for the source m. For this model, the objective of TDOA estimation is to determine all the parameters τ_m, $m = 1, 2, \ldots, M$ using microphone observations.

9.2.3 Single-Source Reverberant Model

While the ideal free-field models have the merit of being simple, they do not take into account the multipath effect. Therefore, such models are inadequate to describe a real reverberant environment and we need a more comprehensive and more informative alternative to model the effect of multipath propagation, leading to the so-called reverberant models, which treat the acoustic impulse response with an FIR filter. If there is only one source in the sound filed as illustrated in Fig. 9.4, the problem can be modeled as a SIMO (single-input multiple-output) system and the nth microphone signal is given by

$$\begin{aligned} y_n(k) &= g_n * s(k) + v_n(k), \\ &= x_n(k) + v_n(k), \quad n = 1, 2, \ldots, N, \end{aligned} \tag{9.9}$$

where g_n is the channel impulse response from the source to microphone n. In vector/matrix form, (9.9) is re-written as

$$\mathbf{y}_n(k) = \mathbf{G}_n \mathbf{s}(k) + \mathbf{v}_n(k), \quad n = 1, 2, \ldots, N, \tag{9.10}$$

where

$$
\begin{aligned}
\mathbf{y}_n(k) &= \left[\, y_n(k) \cdots y_n(k-L+1) \,\right]^T, \\
\mathbf{G}_n &= \begin{bmatrix} g_{n,0} & \cdots & g_{n,L-1} & \cdots & 0 \\ \vdots & \ddots & \ddots & \ddots & \vdots \\ 0 & \cdots & g_{n,0} & \cdots & g_{n,L-1} \end{bmatrix}, \\
\mathbf{s}(k) &= \left[\, s(k)\ s(k-1) \cdots s(k-L+1) \cdots s(k-2L+2) \,\right]^T, \\
\mathbf{v}_n(k) &= \left[\, v_n(k) \cdots v_n(k-L+1) \,\right]^T,
\end{aligned}
$$

and L is the length of the longest channel impulse response of the SIMO system. Again, it is assumed that $v_n(k)$ is uncorrelated with both the source signal and the noise observed at other sensors.

In comparison with the free-field model, the TDOA τ in this reverberant model is an implicit or hidden parameter. With such a model, the TDOA can only be obtained after the SIMO system is "blindly" identified (since the source signal is unknown), which looks like a more difficult problem but is fortunately not insurmountable.

9.2.4 Multiple-Source Reverberant Model

If there are multiple sources in the sound filed, the array can be modeled as a MIMO (multiple-input multiple-output) system with M inputs and N outputs. At time k, we have

$$
\mathbf{y}(k) = \mathbf{G}\mathbf{s}_{ML}(k) + \mathbf{v}(k), \tag{9.11}
$$

where

$$
\begin{aligned}
\mathbf{y}(k) &= \left[\, y_1(k)\ y_2(k) \cdots y_N(k) \,\right]^T, \\
\mathbf{G} &= \left[\, \mathbf{G}_1\ \mathbf{G}_2 \cdots \mathbf{G}_M \,\right], \\
\mathbf{G}_m &= \begin{bmatrix} g_{1m,0} & g_{1m,1} & \cdots & g_{1m,L-1} \\ g_{2m,0} & g_{2m,1} & \cdots & g_{2m,L-1} \\ \vdots & \vdots & \ddots & \vdots \\ g_{Nm,0} & g_{Nm,1} & \cdots & g_{Nm,L-1} \end{bmatrix}_{N\times L}, \\
& \quad m = 1, 2, \ldots, M, \\
\mathbf{v}(k) &= \left[\, v_1(k)\ v_2(k) \cdots v_N(k) \,\right]^T, \\
\mathbf{s}_{ML}(k) &= \left[\, \mathbf{s}_1^T(k)\ \mathbf{s}_2^T(k) \cdots \mathbf{s}_M^T(k) \,\right]^T, \\
\mathbf{s}_m(k) &= \left[\, s_m(k)\ s_m(k-1) \cdots s_m(k-L+1) \,\right]^T.
\end{aligned}
$$

and g_{nm} $(n = 1, 2, \ldots, N,\ m = 1, 2, \ldots, M)$ is the impulse response of the channel from source m to microphone n. Similar to the multiple-source free-field model, we assume that all the source signals are mutually independent,

and $v_n(k)$ is uncorrelated with not only all the source signals but also with the noise observed at other sensors.

For this model, in order to estimate the TDOA, we have to "blindly" identify the MIMO system, which can be an extremely difficult problem.

9.3 Cross-Correlation Method

We are now ready to investigate the algorithms for TDOA estimation. Let us start with the most simple and straightforward method: cross-correlation (CC). Consider the single-source free-field model with only two sensors, i.e., $N = 2$. The cross-correlation function (CCF) between the two observation signals $y_1(k)$ and $y_2(k)$ is defined as

$$r_{y_1y_2}^{\mathrm{CC}}(p) = E\left[y_1(k)y_2(k+p)\right]. \tag{9.12}$$

Substituting (9.5) into (9.12), we can readily deduce that

$$\begin{aligned} r_{y_1y_2}^{\mathrm{CC}}(p) = \alpha_1\alpha_2 r_{ss}^{\mathrm{CC}}(p-\tau) + \alpha_1 r_{sv_2}^{\mathrm{CC}}(p+t) + \\ \alpha_2 r_{sv_1}(p-t-\tau) + r_{v_1v_2}(p). \end{aligned} \tag{9.13}$$

If we assume that $v_n(k)$ is uncorrelated with both the signal and the noise observed at the other sensor, it can be easily checked that $r_{y_1y_2}^{\mathrm{CC}}(p)$ reaches its maximum at $p = \tau$. Therefore, given the CCF, we can obtain an estimate of the TDOA between $y_1(k)$ and $y_2(k)$ as

$$\hat{\tau}^{\mathrm{CC}} = \arg\max_p r_{y_1y_2}^{\mathrm{CC}}(p), \tag{9.14}$$

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

In digital implementation of (9.14), some approximations are required because the CCF is not known and must be estimated. A normal practice is to replace the CCF defined in (9.12) by its time-averaged estimate. Suppose that at time instant k we have a set of observation samples of x_n, $\{x_n(k), x_n(k+1), \cdots, x_n(k+K-1)\}$, $n = 1, 2$, the corresponding CCF can be estimated as either

$$\hat{r}_{y_1y_2}^{\mathrm{CC}}(p) = \begin{cases} \dfrac{1}{K}\displaystyle\sum_{i=0}^{K-p-1} y_1(k+i)y_2(k+i+p), & p \geq 0 \\ \hat{r}_{y_2y_1}^{\mathrm{CC}}(-p), & p < 0 \end{cases}, \tag{9.15}$$

or

$$\hat{r}_{y_1y_2}^{\mathrm{CC}}(p) = \begin{cases} \dfrac{1}{K-p}\displaystyle\sum_{i=0}^{K-p-1} y_1(k+i)y_2(k+i+p), & p \geq 0 \\ \hat{r}_{y_2y_1}^{\mathrm{CC}}(-p), & p < 0 \end{cases}, \tag{9.16}$$

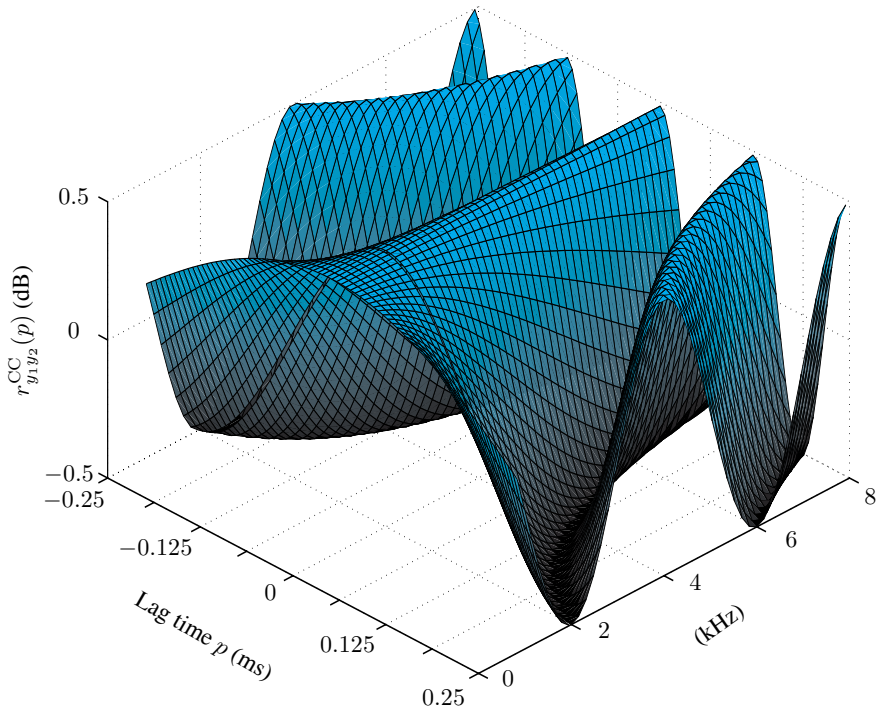


Fig. 9.5. CCF between $y_1(k)$ and $y_2(k)$: the source is a narrowband signal, the incident angle is $\theta = 0^\circ$, there is no noise, i.e., $v_n(k) = 0$, the spacing between the two sensors is $d = 8$ cm, and the sampling frequency is 16 kHz.

where K is the block size. The difference between (9.15) and (9.16) 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.

The CC method is simple to implement. However, its performance is often affected by many factors such as signal self correlation, reverberation, etc. Many efforts have been devoted to improving this method, which will be discussed in the next section. But before we finish this section, we would like to point out one potential problem that is often neglected in TDOA estimation: spatial aliasing. In Chapter 3, we have shown that spatial aliasing may cause ambiguity for the array to distinguish signals propagating from different directions. Similarly, this problem will also affect the TDOA estimation. To see this, let us consider a simple example where the source is a narrowband signal in the form of

$$s(k) = \cos(2\pi f k). \tag{9.17}$$

If we neglect both the propagation attenuation and noise effects in (9.5), we get

$$y_1(k) = \cos[2\pi f(k-t)], \tag{9.18}$$

$$y_2(k) = \cos[2\pi f(k-t-\tau)]. \tag{9.19}$$

Substituting (9.18) and (9.19) into (9.12), we easily obtain

$$r_{y_1y_2}^{\mathrm{CC}}(p) = E\left[y_1(k)y_2(k+p)\right] = \frac{1}{2}\cos[2\pi f(p-\tau)]. \tag{9.20}$$

Figure 9.5 plots the CCF for different frequencies. The spacing between the two microphones is 8 cm. Assuming that the sound velocity is 320 m/s, one can easily check, based on the analysis given in Section 3.3, that when $f > 2$ kHz, there will be spatial aliasing for beamforming. From Fig. 9.5, we see that when $f > 2$ kHz, the CCF experiences multiple peaks in the range of $[-\tau_{\max}, \tau_{\max}]$ ($\tau_{\max}$ is the maximum possible TDOA and can be determined from the spacing between the two microphones), which makes it difficult to search for the correct TDOA. In microphone-array applications, the source is usually speech, which consists of rich frequency components. In order to avoid the spatial aliasing problem and improve TDOA estimation, one should low-pass filter the microphone signal before feeding it to the estimation algorithms. The cutoff frequency can be calculated using the sensor spacing, i.e., $f_{\mathrm{c}} = c/2d$.

9.4 The Family of the Generalized Cross-Correlation Methods

The generalized cross-correlation (GCC) algorithm proposed by Knapp and Carter [145] is the most widely used approach to TDOA estimation. Same as the CC method, GCC employs the free-field model (9.5) and considers only two microphones, i.e., $N = 2$. Then the TDOA estimate between the two microphones is obtained as the lag time that maximizes the CCF between the filtered signals of the microphone outputs [which is often called the generalized CCF (GCCF)]:

$$\hat{\tau}^{\mathrm{GCC}} = \arg\max_{\tau} r_{y_1y_2}^{\mathrm{GCC}}(p), \tag{9.21}$$

where

$$\begin{aligned} r_{y_1y_2}^{\mathrm{GCC}}(p) &= F^{-1}\left[\Psi_{y_1y_2}(f)\right] \\ &= \int_{-\infty}^{\infty} \Psi_{y_1y_2}(f) e^{j2\pi fp} df \\ &= \int_{-\infty}^{\infty} \vartheta(f)\phi_{y_1y_2}(f) e^{j2\pi fp} df \end{aligned} \tag{9.22}$$

is the GCC function, $F^{-1}[\cdot]$ stands for the inverse discrete-time Fourier transform (IDTFT),

$$\phi_{y_1y_2}(f) = E\left[Y_1(f)Y_2^*(f)\right] \tag{9.23}$$

is the cross-spectrum with

$$Y_n(f) = \sum_k y_n(k)e^{-j2\pi fk}, \quad n = 1, 2,$$

$\vartheta(f)$ is a frequency-domain weighting function, and

$$\Psi_{y_1y_2}(f) = \vartheta(f)\phi_{y_1y_2}(f) \tag{9.24}$$

is the generalized cross-spectrum.

There are many different choices of the frequency-domain weighting function $\vartheta(f)$, leading to a variety of different GCC methods.

9.4.1 Classical Cross-Correlation

If we set $\vartheta(f) = 1$, it can be checked that the GCC degenerates to the cross-correlation method discussed in the previous section. The only difference is that now the CCF is estimated using the discrete Fourier transform (DFT) and the inverse DFT (IDFT), which can be implemented efficiently thanks to the fast Fourier transform (FFT).

We know from the free-field model (9.5) that

$$Y_n(f) = \alpha_n S(f)e^{-j2\pi f[t-\mathcal{F}_n(\tau)]} + V_n(f), \quad n = 1, 2. \tag{9.25}$$

Substituting (9.25) into (9.24) and noting that the noise signal at one microphone is uncorrelated with the source signal and the noise signal at the other microphone by assumption, we have

$$\Psi_{y_1y_2}^{\mathrm{CC}}(f) = \alpha_1\alpha_2 e^{-j2\pi f\tau} E\left[|S(f)|^2\right]. \tag{9.26}$$

The fact that $\Psi_{y_1y_2}^{\mathrm{CC}}(f)$ depends on the source signal can be detrimental for TDOA estimation since speech is inherently non-stationary.

9.4.2 Smoothed Coherence Transform

In order to overcome the impact of fluctuating levels of the speech source signal on TDOA estimation, an effective way is to pre-whiten the microphone outputs before their cross-spectrum is computed. This is equivalent to choosing

$$\vartheta(f) = \frac{1}{\sqrt{E\left[|Y_1(f)|^2\right]E\left[|Y_2(f)|^2\right]}}, \tag{9.27}$$

which leads to the so-called smoothed coherence transform (SCOT) method [36]. Substituting (9.25) and (9.27) into (9.24) produces the SCOT cross-spectrum:

$$\begin{aligned}\Psi_{y_1y_2}^{\mathrm{SCOT}}(f) &= \frac{\alpha_1\alpha_2 e^{-j2\pi f\tau} E\left[|S(f)|^2\right]}{\sqrt{E\left[|Y_1(f)|^2\right] E\left[|Y_2(f)|^2\right]}} \\ &= \frac{\alpha_1\alpha_2 e^{-j2\pi f\tau} E\left[|S(f)|^2\right]}{\sqrt{\alpha_1^2 E\left[|S(f)|^2\right] + \sigma_{v_1}^2(f)} \cdot \sqrt{\alpha_2^2 E\left[|S(f)|^2\right] + \sigma_{v_2}^2(f)}} \\ &= \frac{e^{-j2\pi f\tau}}{\sqrt{1+\frac{1}{\mathrm{SNR}_1(f)}} \cdot \sqrt{1+\frac{1}{\mathrm{SNR}_2(f)}}}, \end{aligned} \tag{9.28}$$

where

$$\begin{aligned}\sigma_{v_n}^2(f) &= E\left[|V_n(f)|^2\right], \\ \mathrm{SNR}_n(f) &= \frac{\alpha_n^2 E\left[|S(f)|^2\right]}{E\left[|V_n(f)|^2\right]}, \quad n = 1, 2.\end{aligned}$$

If the SNRs are the same at the two microphones, then we get

$$\Psi_{x_1x_2}^{\mathrm{SCOT}}(f) = \left[\frac{\mathrm{SNR}(f)}{1+\mathrm{SNR}(f)}\right] \cdot e^{-j2\pi f\tau}. \tag{9.29}$$

Therefore, the performance of the SCOT algorithm for TDOA estimation would vary with the SNR. But when the SNR is large enough,

$$\Psi_{x_1x_2}^{\mathrm{SCOT}}(f) \approx e^{-j2\pi f\tau}, \tag{9.30}$$

which implies that the estimation performance is independent of the power of the source signal. So, the SCOT method is theoretically superior to the CC method. But this superiority only holds when the noise level is low.

9.4.3 Phase Transform

It becomes clear by examining (9.22) that the TDOA information is conveyed in the phase rather than the amplitude of the cross-spectrum. Therefore, we can simply discard the amplitude and only keep the phase. By setting

$$\vartheta(f) = \frac{1}{|\phi_{y_1y_2}(f)|}, \tag{9.31}$$

we get the phase transform (PHAT) method [145]. In this case, the generalized cross-spectrum is given by

$$\Psi_{y_1y_2}^{\mathrm{PHAT}}(f) = e^{-j2\pi f\tau}, \tag{9.32}$$

which depends only on the TDOA τ. Substituting (9.32) into (9.22), we obtain an ideal GCC function:

$$r_{y_1y_2}^{\mathrm{PHAT}}(p) = \int_{-\infty}^{\infty} e^{j2\pi f(p-\tau)} df = \begin{cases} \infty, & p = \tau, \\ 0, & \text{otherwise.} \end{cases} \tag{9.33}$$

As a result, the PHAT method performs in general better than the CC and SCOT methods for TDOA estimation with respect to a speech sound source.

The GCC methods are computationally efficient. They induce very short decision delays and hence have a good tracking capability: an estimate is produced almost instantaneously. The GCC methods have been well studied and are found to perform fairly well in moderately noisy and non-reverberant environments [37], [128]. In order to improve their robustness to additive noise, many amendments have been proposed [25], [174], [175], [222]. However, these methods still tend to break down when room reverberation is high. This is insightfully explained by the fact that the GCC methods model the surrounding acoustic environment as an ideal free field and thus have a fundamental weakness in their ability to cope with room reverberation.

9.5 Spatial Linear Prediction Method

In this section, we explore the possibility of using multiple microphones (more than 2) to improve the TDOA estimation in adverse acoustic environments. The fundamental underlying idea is to take advantage of the redundant information provided by multiple sensors. To illustrate the redundancy, let us consider a three-microphone system. In such a system, there are three TDOAs, namely τ_{12}, τ_{13}, and τ_{23}. Apparently, these three TDOAs are not independent but are related as follows: $\tau_{13} = \tau_{12} + \tau_{23}$. Such a relationship was used in [144] and a Kalman filtering based two-stage TDE algorithm was proposed. Recently, with a similar line of thoughts, several fusion algorithms have been developed [55], [93], [172]. In what follows, we present a TDOA estimation algorithm using spatial linear prediction [14], [39], which takes advantage of the TDOA redundancy among multiple microphones in a more intuitive way.

Consider the free-field model in (9.5) with a linear array of N ($N \geq 2$) microphones. If the source is in the far-field and we neglect the noise terms, it can be easily checked that

$$y_n[k + \mathcal{F}_n(\tau)] = \alpha_n s(k - t), \quad \forall n = 1, 2, \ldots, N. \tag{9.34}$$

Therefore, $y_1(k)$ is aligned with $y_n[k + \mathcal{F}_n(\tau)]$. From this relationship, we can defined the forward spatial prediction error signal

$$e_1(k, p) = y_1(k) - \mathbf{y}_{\mathrm{a},2:N}^T(k, p)\mathbf{a}_{2:N}(p), \tag{9.35}$$

where p, again, is a dummy variable for the hypothesized TDOA τ,

$$\mathbf{y}_{\mathrm{a},2:N}(k, p) = \left[y_2[k + \mathcal{F}_2(p)] \cdots y_N[k + \mathcal{F}_N(p)] \right]^T, \tag{9.36}$$

is the aligned (subscript a) signal vector, and

$$\mathbf{a}_{2:N}(p) = \left[a_2(p)\ a_3(p) \cdots a_N(p) \right]^T$$

contains the forward spatial linear prediction coefficients. Minimizing the mean-square value of the prediction error signal

$$J_1(p) = E\left[e_1^2(k,p)\right] \tag{9.37}$$

leads to the linear system

$$\mathbf{R}_{\mathrm{a},2:N}(p)\mathbf{a}_{2:N}(p) = \mathbf{r}_{\mathrm{a},2:N}(p), \tag{9.38}$$

where

$$\begin{aligned}\mathbf{R}_{\mathrm{a},2:N}(p) &= E\left[\mathbf{y}_{\mathrm{a},2:N}(k,p)\mathbf{y}_{\mathrm{a},2:N}^T(k,p)\right] \\ &= \begin{bmatrix} \sigma_{y_2}^2 & r_{\mathrm{a},y_2y_3}(p) & \cdots & r_{\mathrm{a},y_2y_N}(p) \\ r_{\mathrm{a},y_3y_2}(p) & \sigma_{y_3}^2 & \cdots & r_{\mathrm{a},y_3y_N}(p) \\ \vdots & \vdots & \ddots & \vdots \\ r_{\mathrm{a},y_Ny_2}(p) & r_{\mathrm{a},y_Ny_3}(p) & \cdots & \sigma_{y_N}^2 \end{bmatrix}\end{aligned} \tag{9.39}$$

is the spatial correlation matrix of the aligned signals with

$$\begin{aligned}\sigma_{y_n}^2 &= E\left[y_n^2(k)\right], \quad n = 1,2,\ldots,N, \\ r_{\mathrm{a},y_iy_j}(p) &= E\left\{y_i[k+\mathcal{F}_i(p)]y_j[k+\mathcal{F}_j(p)]\right\}, \quad i,j = 1,2,\ldots,N,\end{aligned}$$

and

$$\mathbf{r}_{\mathrm{a},2:N}(p) = \left[\, r_{\mathrm{a},y_1y_2}(p)\ r_{\mathrm{a},y_1y_3}(p)\cdots r_{\mathrm{a},y_1y_N}(p)\,\right]^T.$$

Substituting the solution of (9.38), which is

$$\mathbf{a}_{2:N}(p) = \mathbf{R}_{\mathrm{a},2:N}^{-1}(p)\mathbf{r}_{\mathrm{a},2:N}(p),$$

into (9.35) gives the minimum forward prediction error

$$e_{1,\min}(k,p) = y_1(k) - \mathbf{y}_{\mathrm{a},2:N}^T(k,p)\mathbf{R}_{\mathrm{a},2:N}^{-1}(p)\mathbf{r}_{\mathrm{a},2:N}(p). \tag{9.40}$$

Accordingly, we have

$$J_{1,\min}(p) = E\left\{e_{1,\min}^2(k,p)\right\} = \sigma_{y_1}^2 - \mathbf{r}_{\mathrm{a},2:N}^T(p)\mathbf{R}_{\mathrm{a},2:N}^{-1}(p)\mathbf{r}_{\mathrm{a},2:N}(p). \tag{9.41}$$

Then we can argue that the lag time p inducing a minimum $J_{1,\min}(p)$ would be the TDOA between the first two microphones:

$$\hat{\tau}^{\mathrm{FSLP}} = \arg\min_p J_{1,\min}(p), \tag{9.42}$$

where the superscript "FSLP" stands for forward spatial linear prediction.

If there are only two microphones, i.e., $N = 2$, it can be easily checked that the FLSP algorithm is identical to the CC method. However, as the number of microphones increases, the FLSP approach can take advantage of the redundant information provided by the multiple microphones to improve the

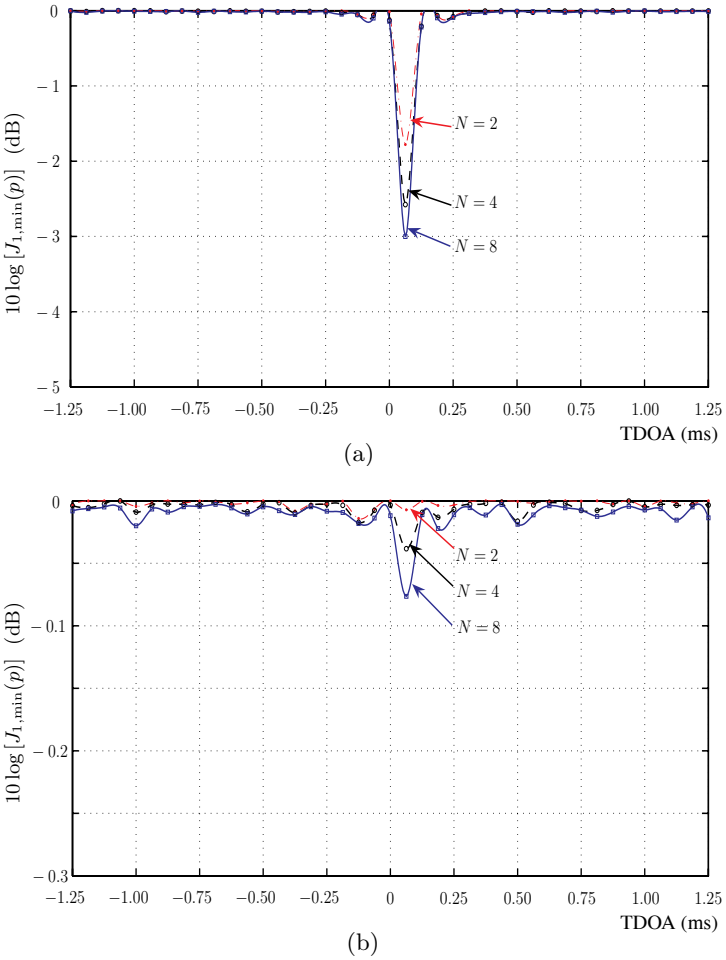


Fig. 9.6. Comparison of $J_{1,\min}(p)$ for different number of microphones. (a) SNR = 10 dB and (b) SNR = −5 dB. The source (speech) is in the array's far-field, the sampling frequency is 16 kHz, the incident angle is $\theta = 75.5°$, and the true TDOA is $\tau = 0.0625$ ms.

TDOA estimation. To illustrate this, we consider a simple simulation example where we have an equispaced linear array consisting of 10 omnidirectional microphones. The spacing between any two neighboring sensors is 8 cm. A sound source located in the far-field radiates a speech signal (female) to the array, with an incident angle of $\theta = 75.5°$. At each microphone, the signal is corrupted by a white Gaussian noise. The microphone signals are digitized with a sampling rate of 16 kHz. Figure 9.6 plots the cost function $J_{1,\min}(p)$ computed from a frame (128 ms in length) of data in two SNR conditions. When SNR = 10 dB, it is seen that the system can achieve correct estimation of the true

TDOA with only two microphones. However, as the number of microphone increases, the valley of the cost function becomes better defined, which will enable an easier search of the minimum. When SNR drops to -5 dB, this time the estimate with two microphones is incorrect. But when 4 or more microphones are employed, the system produces a correct estimate. Both situations clearly indicate that the TDOA estimation performance increases with the number of microphones

Similarly, the TDOA estimation can be developed using backward prediction or interpolation with any one of the N microphone outputs being regarded as the reference signal [39], which will be left to the reader's investigation.

9.6 Multichannel Cross-Correlation Coefficient Algorithm

It is seen from the previous section that the key to the spatial prediction based techniques is the use of the spatial correlation matrix. A more natural way of using the spatial correlation matrix in TDOA estimation is through the so-called multichannel cross-correlation coefficient (MCCC) [14], [39], which measures the correlation among the outputs of an array system and can be viewed as a seamless generalization of the classical cross-correlation coefficient to the multichannel case and where there are multiple random processes.

Following (9.36), we define a new signal vector

$$\mathbf{y}_{\mathrm{a}}(k,p) = \begin{bmatrix} y_1(k) & y_2[k+\mathcal{F}_2(p)] & \cdots & y_N[k+\mathcal{F}_N(p)] \end{bmatrix}^T. \tag{9.43}$$

Similar to (9.39), we can now write the corresponding spatial correlation matrix as

$$\begin{aligned} \mathbf{R}_{\mathrm{a}}(p) &= E\left[\mathbf{y}_{\mathrm{a}}(k,p)\mathbf{y}_{\mathrm{a}}^T(k,p)\right] \\ &= \begin{bmatrix} \sigma_{y_1}^2 & r_{\mathrm{a},y_1y_2}(p) & \cdots & r_{\mathrm{a},y_1y_N}(p) \\ r_{\mathrm{a},y_2y_1}(p) & \sigma_{y_2}^2 & \cdots & r_{\mathrm{a},y_2y_N}(p) \\ \vdots & \vdots & \ddots & \vdots \\ r_{\mathrm{a},y_Ny_1}(p) & r_{\mathrm{a},y_Ny_2}(p) & \cdots & \sigma_{y_N}^2 \end{bmatrix}. \end{aligned} \tag{9.44}$$

The spatial correlation matrix $\mathbf{R}_{\mathrm{a}}(p)$ can be factored as

$$\mathbf{R}_{\mathrm{a}}(p) = \mathbf{\Sigma}\tilde{\mathbf{R}}_{\mathrm{a}}(p)\mathbf{\Sigma}, \tag{9.45}$$

where

$$\mathbf{\Sigma} = \begin{bmatrix} \sigma_{y_1} & 0 & \cdots & 0 \\ 0 & \sigma_{y_2} & \cdots & 0 \\ \vdots & \vdots & \ddots & \vdots \\ 0 & \cdots & 0 & \sigma_{y_N} \end{bmatrix}$$

is a diagonal matrix,

$$\tilde{\mathbf{R}}_{\mathrm{a}}(p) = \begin{bmatrix} 1 & \rho_{\mathrm{a},y_1y_2}(p) & \cdots & \rho_{\mathrm{a},y_1y_N}(p) \\ \rho_{\mathrm{a},y_2y_1}(\tau) & 1 & \cdots & \rho_{\mathrm{a},y_2y_N}(p) \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{\mathrm{a},y_Ny_1}(p) & \rho_{\mathrm{a},y_Ny_2}(p) & \cdots & 1 \end{bmatrix}$$

is a symmetric matrix, and

$$\rho_{\mathrm{a},y_iy_j}(p) = \frac{r_{\mathrm{a},y_iy_j}(p)}{\sigma_{y_i}\sigma_{y_j}}, \quad i,j = 1,2,\ldots,N,$$

is the correlation coefficient between the ith and jth aligned microphone signals.

Since the matrix $\tilde{\mathbf{R}}_{\mathrm{a}}(p)$ is symmetric and positive semi-definite, and its diagonal elements are all equal to one, it can be shown that [14], [39]

$$0 \leq \det\left[\tilde{\mathbf{R}}_{\mathrm{a}}(p)\right] \leq 1, \tag{9.46}$$

where $\det(\cdot)$ stands for determinant.

If there are only two channel, i.e., $N = 2$, it can be easily checked that the squared correlation coefficient is linked to the normalized spatial correlation matrix by

$$\rho^2_{\mathrm{a},y_1y_2}(p) = 1 - \det\left[\tilde{\mathbf{R}}_{\mathrm{a}}(p)\right]. \tag{9.47}$$

Then by analogy, the squared MCCC among the N aligned signals $y_n[k + \mathcal{F}_n(p)]$, $n = 1,2,\ldots,N$, is constructed as

$$\begin{aligned} \rho^2_{\mathrm{a},y_1:y_N}(p) &= 1 - \det\left[\tilde{\mathbf{R}}_{\mathrm{a}}(p)\right] \\ &= 1 - \frac{\det\left[\mathbf{R}_{\mathrm{a}}(p)\right]}{\prod_{n=1}^{N}\sigma^2_{y_n}}. \end{aligned} \tag{9.48}$$

The MCCC has the following properties (presented without proof) [14], [39]:

1. $0 \leq \rho^2_{\mathrm{a},y_1:y_N}(p) \leq 1$;
2. if two or more signals are perfectly correlated, then $\rho^2_{\mathrm{a},y_1:y_N}(p) = 1$;
3. if all the signals are completely uncorrelated with each other, then $\rho^2_{\mathrm{a},y_1:y_N}(p) = 0$;
4. if one of the signals is completely uncorrelated with the $N-1$ other signals, then the MCCC will measure the correlation among those $N-1$ remaining signals.

Using the definition of the MCCC, we deduce an estimate of the TDOA between the first two microphone signals as

$$\hat{\tau}^{\mathrm{MCCC}} = \arg\max_p \rho^2_{\mathrm{a},y_1:y_N}(p), \tag{9.49}$$

which is equivalent to computing

$$\begin{aligned}\hat{\tau}^{\mathrm{MCCC}} &= \arg\max_p \left\{1 - \det\left[\tilde{\mathbf{R}}_{\mathrm{a}}(p)\right]\right\} \\ &= \arg\max_p \left\{1 - \frac{\det\left[\mathbf{R}_{\mathrm{a}}(p)\right]}{\prod_{n=1}^{N}\sigma_{y_n}^2}\right\} \\ &= \arg\min_p \det\left[\tilde{\mathbf{R}}_{\mathrm{a}}(p)\right] \\ &= \arg\min_p \det\left[\mathbf{R}_{\mathrm{a}}(p)\right]. \end{aligned} \tag{9.50}$$

To illustrate the TDOA estimation with the MCCC algorithm, we study the same example that was used in Section 9.5. The cost function $\det\left[\mathbf{R}_{\mathrm{a}}(p)\right]$ computed for the same frame of data used in Fig 9.6 is plotted in Fig 9.7. It is clearly seen that the algorithm achieves better estimation performance as more microphones are used.

To investigate the link between the MCCC and FSLP methods, let us revisit the spatial prediction error function given by (9.41). We define

$$\begin{aligned}\mathbf{a}(p) &= \left[\, a_1(p)\ a_2(p)\ \cdots\ a_N(p)\,\right]^T \\ &= \left[\, a_1(p)\ \mathbf{a}_{2:N}^T(p)\,\right]^T. \end{aligned} \tag{9.51}$$

Then, for $a_1(p) = -1$, the forward spatial prediction error signal (9.35) can be written as

$$e_1(k,p) = -\mathbf{y}_{\mathrm{a}}^T(k,p)\mathbf{a}(p), \tag{9.52}$$

and (9.37) can be expressed as

$$\begin{aligned}J_1(p) &= E\left[e_1^2(k,p)\right] + \mu\left[\mathbf{u}^T\mathbf{a}(p) + 1\right] \\ &= \mathbf{a}^T(p)\mathbf{R}_{\mathrm{a}}(p)\mathbf{a}(p) + \mu\left[\mathbf{u}^T\mathbf{a}(p) + 1\right], \end{aligned} \tag{9.53}$$

where μ is a Lagrange multiplier introduced to force $a_1(p)$ to have the value -1 and

$$\mathbf{u} = \left[\, 1\ 0\ \cdots\ 0\,\right]^T.$$

Taking the derivative of (9.53) with respect to $\mathbf{a}(p)$ and setting the result to zero yields

$$\frac{\partial J_1(p)}{\partial \mathbf{a}(p)} = 2\mathbf{R}_{\mathrm{a}}(p)\mathbf{a}(p) + \mu\mathbf{u} = \mathbf{0}_{N\times 1}. \tag{9.54}$$

Solving (9.54) for $\mathbf{a}(p)$ produces

$$\mathbf{a}(p) = -\frac{\mu\mathbf{R}_{\mathrm{a}}^{-1}(p)\mathbf{u}}{2}. \tag{9.55}$$

Substituting (9.55) into (9.53) leads to

$$J_1(p) = \mu\left[1 - \frac{\mathbf{u}^T\mathbf{R}_{\mathrm{a}}^{-1}(p)\mathbf{u}}{4}\mu\right], \tag{9.56}$$

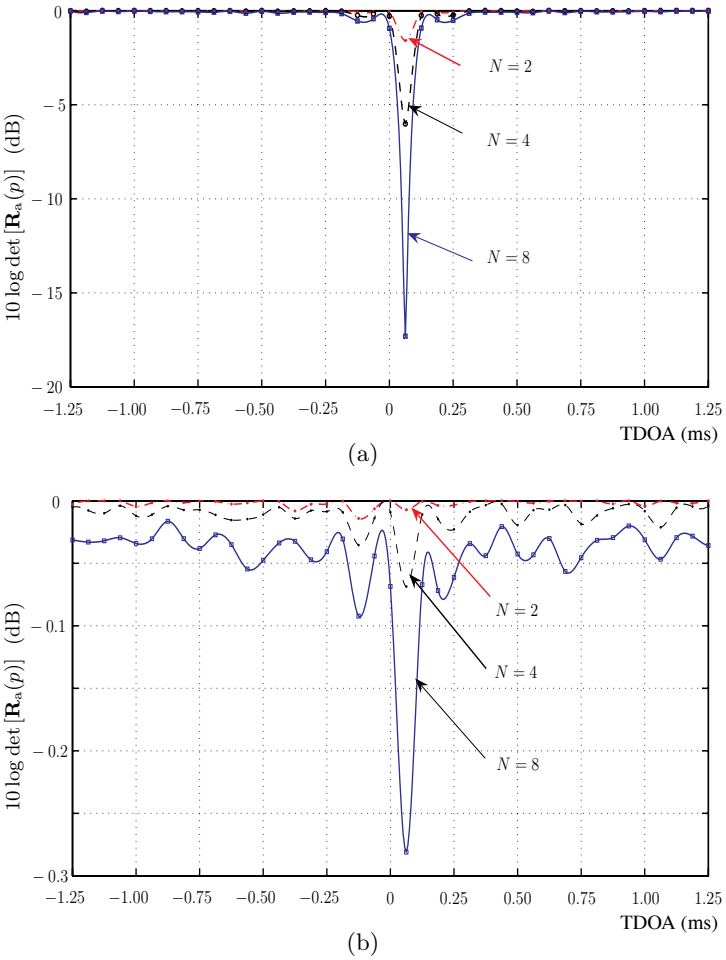


Fig. 9.7. Comparison of det $[\mathbf{R}_a(p)]$ for an equispaced linear array with different number of microphones. (a) SNR = 10 dB and (b) SNR = −5 dB. The source is in the array's far-field, the sampling frequency is 16 kHz, the incident angle is $\theta = 75.5°$, and the true TDOA is $\tau = 0.0625$ ms.

from which we know that

$$J_{1,\min}(p) = \frac{1}{\mathbf{u}^T \mathbf{R}_a^{-1}(p)\mathbf{u}}. \tag{9.57}$$

Substituting (9.45) into (9.57) and using the fact that

$$\mathbf{\Sigma}^{-1}\mathbf{u} = \frac{\mathbf{u}}{\sigma_{y_1}},$$

we have

$$J_{1,\min}(p) = \frac{\sigma_{y_1}^2}{\mathbf{u}^T \tilde{\mathbf{R}}_{\mathrm{a}}^{-1}(p)\mathbf{u}}. \tag{9.58}$$

Note that $\mathbf{u}^T\tilde{\mathbf{R}}_{\mathrm{a}}^{-1}(p)\mathbf{u}$ is the $(1,1)$th element of the matrix $\tilde{\mathbf{R}}_{\mathrm{a}}^{-1}(p)$, which is computed using the adjoint method as the $(1,1)$th cofactor of $\tilde{\mathbf{R}}_{\mathrm{a}}(p)$ divided by the determinant of $\tilde{\mathbf{R}}_{\mathrm{a}}(p)$, i.e.,

$$\mathbf{u}^T\tilde{\mathbf{R}}_{\mathrm{a}}^{-1}(p)\mathbf{u} = \frac{\det\left[\tilde{\mathbf{R}}_{\mathrm{a,2:N}}(p)\right]}{\det\left[\tilde{\mathbf{R}}_{\mathrm{a}}(p)\right]}, \tag{9.59}$$

where $\tilde{\mathbf{R}}_{\mathrm{a,2:N}}(p)$ is the lower-right submatrix of $\tilde{\mathbf{R}}_{\mathrm{a}}(p)$ by removing the first row and the first column. By substituting (9.59) into (9.58), we get

$$J_{1,\min}(p) = \sigma_{y_1}^2 \cdot \frac{\det\left[\tilde{\mathbf{R}}_{\mathrm{a}}(p)\right]}{\det\left[\tilde{\mathbf{R}}_{\mathrm{a,2:N}}(p)\right]}. \tag{9.60}$$

Therefore, the FSLP estimate of τ is found as

$$\begin{aligned}\hat{\tau}^{\mathrm{FSLP}} &= \arg\min_p J_{1,\min}(p) \\ &= \arg\min_p \frac{\det\left[\tilde{\mathbf{R}}_{\mathrm{a}}(p)\right]}{\det\left[\tilde{\mathbf{R}}_{\mathrm{a,2:N}}(p)\right]}.\end{aligned} \tag{9.61}$$

Comparing (9.50) to (9.61) reveals a clear distinction between the two methods in spite of high similarity. In practice, the FSLP method may suffer numerical instabilities since the calculation of the FSLP cost function (9.60) involves the division by $\det\left[\tilde{\mathbf{R}}_{\mathrm{a,2:N}}(p)\right]$, while the MCCC method is found to be fairly stable. If we compare Figs. 9.6 and 9.7, one can notice that the peak of the MCCC cost function is better defined than that of the FSLP function, which indicates that the MCCC algorithm is superior to the FSLP method.

It is worth pointing out that the microphone outputs can be pre-whitened before computing their MCCC as was done in the PHAT algorithm in the two-channel scenario. By doing so, the TDOA estimation algorithms become more robust to the volume variation of the speech source signal.

9.7 Eigenvector-Based Techniques

Another way to use the spatial correlation matrix for TDOA estimation is through the eigenvector-based techniques. These techniques were originally developed in radar for DOA estimation [184], [192], [198], and have been recently extended to processing a broadband speech using microphone arrays

[58]. We start with the narrowband formulation since it is much easier to comprehend. We consider the single-source free-field model in (9.5) with N microphones. For ease of analysis, we assume that the source is in the array's far-field, all the attenuation coefficients α_n are equal to 1, and the observation noises $v_n(k)$, $n = 1, 2, \ldots, N$, are mutually independent Gaussian random processes with the same variance.

9.7.1 Narrowband MUSIC

If we transform both sides of (9.5) into the frequency domain, the nth microphone output can be written as

$$\begin{aligned} Y_n(f) &= X_n(f) + V_n(f) \\ &= S(f)e^{-j2\pi[t+\mathcal{F}_n(\tau)]f} + V_n(f), \end{aligned} \tag{9.62}$$

where $Y_n(f)$, $X_n(f)$, $V_n(f)$, and $S(f)$ are, respectively, the DTFT of $y_n(k)$, $x_n(k)$, $v_n(k)$, and $s(k)$. Let us define the following frequency-domain vector:

$$\vec{\mathbf{y}} = \left[Y_1(f)\ Y_2(f) \cdots Y_N(f) \right]^T. \tag{9.63}$$

Substituting (9.62) into (9.63), we get

$$\begin{aligned} \vec{\mathbf{y}} &= \vec{\mathbf{x}} + \vec{\mathbf{v}} \\ &= \boldsymbol{\varsigma}(\tau)S(f)e^{-j2\pi tf} + \vec{\mathbf{v}}, \end{aligned} \tag{9.64}$$

where

$$\boldsymbol{\varsigma}(\tau) = \left[e^{-j2\pi\mathcal{F}_1(\tau)f} \quad e^{-j2\pi\mathcal{F}_2(\tau)f} \quad \cdots \quad e^{-j2\pi\mathcal{F}_N(\tau)f} \right]^T,$$

and $\vec{\mathbf{v}}$ is defined similarly to $\vec{\mathbf{y}}$. It follows that the output covariance matrix can be written as

$$\mathbf{R}_Y = E\left(\vec{\mathbf{y}}\vec{\mathbf{y}}^H\right) = \mathbf{R}_X + \sigma_V^2\mathbf{I}, \tag{9.65}$$

where

$$\mathbf{R}_X = \sigma_S^2\boldsymbol{\varsigma}(\tau)\boldsymbol{\varsigma}^H(\tau), \tag{9.66}$$

and $\sigma_S^2 = E[|S(f)|^2]$ and $\sigma_V^2 = E[|V_1(f)|^2] = \cdots = E[|V_N(f)|^2]$ are, respectively, the signal and noise variances. It can be easily checked that the positive semi-definite matrix $\mathbf{R}_X$ is of rank 1. Therefore, if we perform the eigenvalue decomposition of $\mathbf{R}_Y$, we obtain

$$\mathbf{R}_Y = \mathbf{B}\boldsymbol{\Lambda}\mathbf{B}^H, \tag{9.67}$$

where

$$\begin{aligned}\mathbf{\Lambda} &= \mathrm{diag}[\,\lambda_{Y,1} \quad \lambda_{Y,2} \quad \cdots \quad \lambda_{Y,N}\,] \\ &= \mathrm{diag}[\,\lambda_{X,1}+\sigma_V^2 \quad \sigma_V^2 \quad \cdots \quad \sigma_V^2\,]\end{aligned} \tag{9.68}$$

is a diagonal matrix consisting of the eigenvalues of $\mathbf{R}_Y$,

$$\mathbf{B} = \left[\,\mathbf{b}_1\ \mathbf{b}_2 \cdots \mathbf{b}_N\,\right], \tag{9.69}$$

$\mathbf{b}_n$ is the eigenvector associated with the eigenvalue $\lambda_{Y,n}$, and $\lambda_{X,1}$ is the only non-zero positive eigenvalue of $\mathbf{R}_X$.

Therefore, for $n \geq 2$, we have

$$\mathbf{R}_Y\mathbf{b}_n = \lambda_{Y,n}\mathbf{b}_n = \sigma_V^2\mathbf{b}_n. \tag{9.70}$$

We also know that

$$\mathbf{R}_Y\mathbf{b}_n = \left[\sigma_S^2\boldsymbol{\varsigma}(\tau)\boldsymbol{\varsigma}^H(\tau) + \sigma_V^2\mathbf{I}\right]\mathbf{b}_n. \tag{9.71}$$

The combination of (9.70) and (9.71) indicates that

$$\sigma_S^2\boldsymbol{\varsigma}(\tau)\boldsymbol{\varsigma}^H(\tau)\mathbf{b}_n = \mathbf{0}, \tag{9.72}$$

which is equivalent to

$$\boldsymbol{\varsigma}^H(\tau)\mathbf{b}_n = 0 \tag{9.73}$$

or

$$\mathbf{b}_n^H\boldsymbol{\varsigma}(\tau) = 0 \tag{9.74}$$

This is to say that the eigenvectors associated with the $N-1$ lowest eigenvalues of $\mathbf{R}_Y$ are orthogonal to the vector corresponding to the actual TDOA. This remarkable observation forms the cornerstone for almost all eigenvector-based algorithms. If we form the following cost function

$$J_{\mathrm{MUSIC}}(p) = \frac{1}{\displaystyle\sum_{n=2}^{N}\left|\mathbf{b}_n^H\boldsymbol{\varsigma}(p)\right|^2}, \tag{9.75}$$

where the subscript "MUSIC" stands for MUltiple SIgnal Classification (MUSIC) [198]. The lag time p that gives the maximum of $J_{\mathrm{MUSIC}}(p)$ corresponds to the TDOA τ:

$$\hat{\tau}^{\mathrm{MUSIC}} = \arg\max_p J_{\mathrm{MUSIC}}(p). \tag{9.76}$$

9.7.2 Broadband MUSIC

While the narrowband formulation of the MUSIC algorithm is straightforward to follow, it does not work well for microphone arrays because speech is nonstationary in nature. Even during the presence of speech, each frequency band may not permanently be occupied with speech, and for a large percentage of the time the band may consist of noise only. One straightforward way of circumventing this issue is to fuse the cost function given in (9.75) across all the frequency bands before searching for the TDOA. This fusion method will, in general, make the peak less well defined, thereby degrading the estimation performance. A more natural broadband MUSIC formulation has been recently developed [58]. This broadband MUSIC is derived based on the spatial correlation matrix defined in Section 9.5. Let us rewrite the alignment signal vector given in (9.43),

$$\mathbf{y}_{\mathrm{a},1:N}(k,p) = \left[\, y_1[k+\mathcal{F}_1(p)]\; y_2[k+\mathcal{F}_2(p)] \cdots y_N[k+\mathcal{F}_N(p)] \,\right]^T, \quad (9.77)$$

The spatial correlation matrix is given by

$$\begin{aligned} \mathbf{R}_{\mathrm{a}}(p) &= E\left[\mathbf{y}_{\mathrm{a},1:N}(k,p)\mathbf{y}_{\mathrm{a},1:N}^T(k,p)\right] \\ &= \mathbf{R}_s(p) + \sigma_v^2\mathbf{I}, \end{aligned} \quad (9.78)$$

where the source signal covariance matrix $\mathbf{R}_s(p)$ is given by

$$\mathbf{R}_s(p) = \begin{bmatrix} \sigma_s^2 & r_{ss,12}(p,\tau) & \cdots & r_{ss,1N}(p,\tau) \\ r_{ss,21}(p,\tau) & \sigma_s^2 & \cdots & r_{ss,2N}(p,\tau) \\ \vdots & \vdots & \ddots & \vdots \\ r_{ss,N1}(p,\tau) & r_{ss,N2}(p,\tau) & \cdots & \sigma_s^2 \end{bmatrix}, \quad (9.79)$$

and

$$r_{ss,ij}(p,\tau) = E\left\{s[k-t-\mathcal{F}_i(\tau)+\mathcal{F}_i(p)]s[k-t-\mathcal{F}_j(\tau)+\mathcal{F}_j(p)]\right\}. \quad (9.80)$$

If $p = \tau$, we easily check that

$$\mathbf{R}_s(\tau) = \sigma_s^2 \begin{bmatrix} 1 & 1 & \cdots & 1 \\ 1 & 1 & \cdots & 1 \\ \vdots & \vdots & \ddots & \vdots \\ 1 & 1 & \cdots & 1 \end{bmatrix}, \quad (9.81)$$

which is a matrix of rank 1. If $p \neq \tau$, the rank of this matrix depends on the characteristics of the source signal. If the source signal is a white process, we can see that $\mathbf{R}_s(p)$ is a diagonal matrix with $\mathbf{R}_s(p) = \mathrm{diag}\left[\,\sigma_s^2 \;\; \sigma_s^2 \;\; \cdots \;\; \sigma_s^2\,\right]$. In this particular case, $\mathbf{R}_s(p)$ is of full rank. In general, if $p \neq \tau$, $\mathbf{R}_s(p)$ is positive semi-definite, and its rank is greater than 1. Let us perform the eigenvalue decomposition of $\mathbf{R}_{\mathrm{a}}(p)$ and $\mathbf{R}_s(p)$. Let

$\lambda_{s,1}(p) \geq \lambda_{s,2}(p) \geq \cdots \geq \lambda_{s,N}(p)$ denote the N eigenvalues of $\mathbf{R}_s(p)$. Then the N eigenvalues of $\mathbf{R}_{\mathrm{a}}(p)$ are given by

$$\lambda_{y,n}(p) = \lambda_{s,n}(p) + \sigma_v^2. \tag{9.82}$$

Further let $\mathbf{b}_1(p), \mathbf{b}_2(p), \cdots, \mathbf{b}_N(p)$ denote their associated eigenvectors (since $\mathbf{R}_{\mathrm{a}}(p)$ is symmetric and Toeplitz, all the eigenvectors are real-valued), then

$$\mathbf{R}_{\mathrm{a}}(p)\mathbf{B}(p) = \mathbf{B}(p)\mathbf{\Lambda}(p), \tag{9.83}$$

where

$$\mathbf{B}(p) = \left[\, \mathbf{b}_1(p)\ \mathbf{b}_2(p) \cdots \mathbf{b}_N(p) \,\right], \tag{9.84}$$

$$\mathbf{\Lambda}(p) = \mathrm{diag}\left[\, \lambda_{y,1}(p)\ \lambda_{y,2}(p) \cdots \lambda_{y,N}(p) \,\right]. \tag{9.85}$$

When $p = \tau$, we already know that $\mathbf{R}_{\mathrm{s}}(\tau)$ is of rank 1. Therefore, for $n \geq 2$, we have

$$\mathbf{R}_{\mathrm{a}}(\tau)\mathbf{b}_n(\tau) = \left[\mathbf{R}_{\mathrm{s}}(\tau) + \sigma_v^2\mathbf{I}\right]\mathbf{b}_n(\tau) = \sigma_v^2\mathbf{b}_n(\tau), \tag{9.86}$$

which implies

$$\mathbf{b}_n^T(p)\mathbf{R}_{\mathrm{a}}(p)\mathbf{b}_n(p) = \begin{cases} \sigma_v^2, & p = \tau \\ \lambda_{y,n}(p) \geq \sigma_v^2, & p \neq \tau \end{cases}. \tag{9.87}$$

Therefore, if we form the following function

$$J_{\mathrm{BMUSIC}}(p) = \frac{1}{\sum_{n=2}^{N} \mathbf{b}_n^T(p)\mathbf{R}_{\mathrm{a}}(p)\mathbf{b}_n(p)}, \tag{9.88}$$

the peak of this cost function will correspond to the true TDOA τ:

$$\hat{\tau}^{\mathrm{BMUSIC}} = \arg\max_p J_{\mathrm{BMUSIC}}(p). \tag{9.89}$$

Although the forms of the broadband and narrowband MUSIC algorithms look similar, they are different in many aspects, such as

- the broadband algorithm can take either broadband or narrowband signals as its inputs, while the narrowband algorithm can only work for narrowband signals;
- in the narrowband case, we only need to perform the eigenvalue decomposition once, but in the broadband situation we will have to compute the eigenvalue decomposition for all the spatial correlation matrices $\mathbf{R}_{\mathrm{a}}(p)$, $-\tau_{\max} \leq p \leq \tau_{\max}$;
- in the narrowband case, when $p = \tau$, the objective function $J_{\mathrm{MUSIC}}(p)$ approaches infinity, so the peak is well defined. However, in the broadband situation, the maximum of the cost function $J_{\mathrm{BMUSIC}}(p)$ is $1/[(N-1)\sigma_v^2]$, which indicates that the peak may be less well-defined.

9.8 Minimum Entropy Method

So far, we have explored the use of the cross-correlation information between different channels for TDOA estimation. The correlation coefficient, regardless if it is computed between two or multiple channels, is a second-order-statistics (SOS) measure of dependence between random Gaussian variables. However for non-Gaussian source signals such as speech, higher order statistics (HOS) may have more to say about their dependence. This section discusses the use of HOS for TDOA estimation through the concept of entropy.

Entropy is a statistical (apparently HOS) measure of randomness or uncertainty of a random variable; it was introduced by Shannon in the context of communication theory [203]. For a random variable y with a probability density function (PDF) $p(y)$ (note here we choose not to distinguish random variables and their realizations), the entropy is defined as [52]

$$\begin{aligned} H(y) &= -\int p(y) \ln p(y) dy \\ &= -E\left[\ln p(y)\right]. \end{aligned} \tag{9.90}$$

The entropy (in the continuous case) is a measure of the structure contained in the PDF [146]. As far as the multivariate random variable $\mathbf{y}_{\mathrm{a}}(k,p)$ given by (9.43) is concerned, the joint entropy is

$$H\left[\mathbf{y}_{\mathrm{a}}(k,p)\right] = -\int p\left[\mathbf{y}_{\mathrm{a}}(k,p)\right] \ln p\left[\mathbf{y}_{\mathrm{a}}(k,p)\right] d\mathbf{y}_{\mathrm{a}}(k,p). \tag{9.91}$$

It was then argued in [19] that the time lag p that gives the minimum of $H\left[\mathbf{y}_{\mathrm{a}}(k,p)\right]$ corresponds to the TDOA between the two microphones:

$$\hat{\tau}^{\mathrm{ME}} = \arg\min_{p} H\left[\mathbf{y}_{\mathrm{a}}(k,p)\right], \tag{9.92}$$

where the superscript "ME" refers to the minimum entropy method.

9.8.1 Gaussian Source Signal

If the source is Gaussian, so are the microphone outputs in the absence of noise. Suppose that the aligned microphone signals are zero mean and joint Gaussian random signals. Their joint PDF is then given by

$$p\left[\mathbf{y}_{\mathrm{a}}(k,p)\right] = \frac{\exp\left[-\eta_{\mathrm{a}}(k,p)/2\right]}{\sqrt{(2\pi)^N \det\left[\mathbf{R}_{\mathrm{a}}(p)\right]}}, \tag{9.93}$$

where

$$\eta_{\mathrm{a}}(k,p) = \mathbf{y}_{\mathrm{a}}^T(k,p)\mathbf{R}_{\mathrm{a}}^{-1}(p)\mathbf{y}_{\mathrm{a}}(k,p). \tag{9.94}$$

By substituting (9.93) into (9.91), the joint entropy can be computed [19] as

$$H\left[\mathbf{y}_{\mathrm{a}}(k,p)\right] = \frac{1}{2}\ln\left\{(2\pi e)^N \det\left[\mathbf{R}_{\mathrm{a}}(p)\right]\right\}. \tag{9.95}$$

Consequently, (9.92) becomes

$$\hat{\tau}^{\mathrm{ME}} = \arg\min_{p} \det\left[\mathbf{R}_{\mathrm{a}}(p)\right]. \tag{9.96}$$

It is clear from (9.50) and (9.96) that minimizing the entropy is equivalent to maximizing the MCCC for Gaussian source signals.

9.8.2 Speech Source Signal

Speech is a complicated random process and there is no rigorous mathematical formula for its entropy. But in speech research, it was found that speech can be fairly well modeled by a Laplace distribution [85], [186].

The univariate Laplace distribution with mean zero and variance σ_y^2 is given by

$$p(y) = \frac{\sqrt{2}}{2\sigma_y} e^{-\sqrt{2}|y|/\sigma_y}, \tag{9.97}$$

and the corresponding entropy is [52]

$$H(y) = 1 + \ln\left(\sqrt{2}\ \sigma_y\right). \tag{9.98}$$

Suppose that $\mathbf{y}_{\mathrm{a}}(k,p)$ has a multivariate Laplace distribution with mean $\mathbf{0}$ and covariance matrix $\mathbf{R}_{\mathrm{a}}(p)$ [147], [67]:

$$p\left[\mathbf{y}_{\mathrm{a}}(k,p)\right] = \frac{2\left[\eta_{\mathrm{a}}(k,p)/2\right]^{Q/2} K_Q\left[\sqrt{2\eta_{\mathrm{a}}(k,p)}\right]}{\sqrt{(2\pi)^N \det\left[\mathbf{R}_{\mathrm{a}}(p)\right]}}, \tag{9.99}$$

where $Q = (2-N)/2$ and $K_Q(\cdot)$ is the modified Bessel function of the third kind (also called the modified Bessel function of the second kind) given by

$$K_Q(a) = \frac{1}{2}\left(\frac{a}{2}\right)^Q \int_0^\infty z^{-Q-1} \exp\left(-z - \frac{a^2}{4z}\right) dz, \quad a > 0. \tag{9.100}$$

The joint entropy is

$$H\left[\mathbf{y}_{\mathrm{a}}(k,p)\right] = \frac{1}{2}\ln\left\{\frac{(2\pi)^N \det\left[\mathbf{R}_{\mathrm{a}}(p)\right]}{4}\right\} - \frac{Q}{2} E\left\{\ln\left[\frac{\eta_{\mathrm{a}}(k,p)}{2}\right]\right\} - E\left\{\ln K_Q\left[\sqrt{2\eta_{\mathrm{a}}(k,p)}\right]\right\}. \tag{9.101}$$

The two quantities $E\left\{\ln\left[\eta_{\mathrm{a}}(k,p)/2\right]\right\}$ and $E\left\{\ln K_Q\left[\sqrt{2\eta_{\mathrm{a}}(k,p)}\right]\right\}$ do not seem to have a closed form. So a numerical scheme needs to be developed to estimate them. One possibility to do this is the following. Assume that all

processes are ergodic. As a result, ensemble averages can be replaced by time averages. If there are K samples for each element of the observation vector $\mathbf{y}_{\mathrm{a}}(k,p)$, the following estimators were proposed in [19]:

$$E\left\{\ln\left[\eta_{\mathrm{a}}(k,p)/2\right]\right\} \approx \frac{1}{K}\sum_{k=0}^{K-1}\ln\left[\eta_{\mathrm{a}}(k,p)/2\right], \tag{9.102}$$

$$E\left\{\ln K_Q\left[\sqrt{2\eta_{\mathrm{a}}(k,p)}\right]\right\} \approx \frac{1}{K}\sum_{k=0}^{K-1}\ln K_Q\left[\sqrt{2\eta_{\mathrm{a}}(k,p)}\right]. \tag{9.103}$$

The simulation results presented in [19] show that the ME algorithm performs in general comparably to or better than the MCCC algorithm. Apparently the ME algorithm is computationally intensive. But the idea of using entropy expands our horizon of knowledge in pursuit of new TDOA estimation algorithms.

9.9 Adaptive Eigenvalue Decomposition Algorithm

The adaptive eigenvalue decomposition (AED) algorithm approaches the TDOA estimation problem from a different point of view as compared to the methods discussed in the previous sections. Similar to the GCC family, the AED considers only the scenario with a single source and two microphones, but it adopts the real reverberant model instead of the free-field model. It first identifies the two channel impulse responses from the source to the two sensors, and then measures the TDOA by detecting the two direct paths. Since the source signal is unknown, the channel identification has to be a blind method.

Following the single source reverberant model (9.9) and the fact that, in the absence of additive noise,

$$y_1(k) * g_2 = x_1(k) * g_2 = s(k) * g_1 * g_2 = x_2(k) * g_1 = y_2(k) * g_1, \tag{9.104}$$

we deduce the following cross-relation in vector/matrix form at time k:

$$\mathbf{y}^T(k)\mathbf{w} = \mathbf{y}_1^T(k)\mathbf{g}_2 - \mathbf{y}_2^T(k)\mathbf{g}_1 = 0, \tag{9.105}$$

where

$$\begin{aligned}
\mathbf{y}(k) &= \left[\,\mathbf{y}_1^T(k)\ \mathbf{y}_2^T(k)\,\right]^T,\\
\mathbf{w} &= \left[\,\mathbf{g}_2^T\ -\mathbf{g}_1^T\,\right]^T,\\
\mathbf{g}_n &= \left[\,g_{n,0}\ g_{n,1}\ \cdots\ g_{n,L-1}\,\right]^T,\quad n=1,2.
\end{aligned}$$

Multiplying (9.105) by $\mathbf{y}(k)$ from the left-hand side and taking expectation yields

$$\mathbf{R}_{yy}\mathbf{w} = \mathbf{0}_{2L\times 1}, \tag{9.106}$$

where $\mathbf{R}_{yy} = E\left[\mathbf{y}(k)\mathbf{y}^T(k)\right]$ is the covariance matrix of the two microphone signals. This indicates that the vector $\mathbf{w}$, which consists of the two impulse responses, is in the null space of $\mathbf{R}_{yy}$. More specifically, $\mathbf{w}$ is an eigenvector of $\mathbf{R}_{yy}$ corresponding to the eigenvalue 0. If $\mathbf{R}_{yy}$ is rank deficient by 1, $\mathbf{w}$ can be uniquely determined up to a scaling factor, which is equivalent to saying that the two-channel SIMO system can be blindly identified. Using what has been proved in [238], we know that such a two-channel acoustic SIMO system is blindly identifiable using only the second-order statistics (SOS) of the microphone outputs if and only if the following two conditions hold:

- the polynomials formed from $\mathbf{g}_1$ and $\mathbf{g}_2$ are co-prime, i.e., their channel transfer functions share no common zeros;
- the autocorrelation matrix $\mathbf{R}_{ss} = E[\mathbf{s}(k)\mathbf{s}^T(k)]$ of the source signal is of full rank (such that the SIMO system can be fully excited).

In practice, noise always exists and the covariance matrix $\mathbf{R}_{yy}$ is positive definite rather than positive semi-definite. As a consequence, $\mathbf{w}$ is found as the normalized eigenvector of $\mathbf{R}_{yy}$ corresponding to the smallest eigenvalue:

$$\hat{\mathbf{w}} = \arg\min_{\mathbf{w}} \mathbf{w}^T\mathbf{R}_{yy}\mathbf{w} \quad \text{subject to} \quad \|\mathbf{w}\| = 1. \tag{9.107}$$

In the AED algorithm, solving (9.107) is carried out in an adaptive manner using a constrained LMS algorithm:

$$\begin{aligned}
&\text{Initialize}\\
&\hat{\mathbf{g}}_n(0) = \left[\begin{array}{cccc}\frac{\sqrt{2}}{2} & 0 & \cdots & 0\end{array}\right]^T, \quad n = 1, 2,\\
&\hat{\mathbf{w}}(0) = \left[\begin{array}{cc}\hat{\mathbf{g}}_2^T(0) & -\hat{\mathbf{g}}_1^T(0)\end{array}\right]^T,\\
&\\
&\text{Compute,} \quad \text{for } k = 0, 1, \ldots\\
&e(k) = \hat{\mathbf{w}}^T(k)\mathbf{y}(k),\\
&\hat{\mathbf{w}}(k+1) = \frac{\hat{\mathbf{w}}(k) - \mu e(k)\mathbf{y}(k)}{\|\hat{\mathbf{w}}(k) - \mu e(k)\mathbf{y}(k)\|},
\end{aligned} \tag{9.108}$$

where the adaptation step size μ is a small positive constant.

After the AED algorithm converges, the time difference between the direct paths of the two identified channel impulse responses $\hat{\mathbf{g}}_1$ and $\hat{\mathbf{g}}_2$ is measured as the TDOA estimate:

$$\hat{\tau}^{\text{AED}} = \arg\max_l |\hat{g}_{1,l}| - \arg\max_l |\hat{g}_{2,l}|. \tag{9.109}$$

9.10 Adaptive Blind Multichannel Identification Based Methods

The AED algorithm provides us a new way to look at the TDOA estimation problem, which was found particularly robust in a reverberant environment. It applies the more realistic real-reverberant model to a two-microphone acoustic system at a time and attempts to blindly identify the two-channel impulse responses, from which the embedded TDOA information of interest is then extracted. Clearly the blind two-channel identification technique plays a central role in such an approach. The more accurately the two impulse responses are blindly estimated, the more precisely the TDOA can be inferred. But for a two-channel system, the zeros of the two channels can be close especially when their impulse responses are long, which leads to an ill-conditioned system that is difficult to identify. If they share some common zeros, the system becomes unidentifiable (using only second-order statistics) and the AED algorithm may not be better than the GCC methods. It was suggested in [120] that this problem can be alleviated by employing more microphones. When more microphones are employed, it is less likely for all channels to share a common zero. As such, blind identification deals with a more well-conditioned SIMO system and the solutions can be globally optimized over all channels. The resulting algorithm is referred as the adaptive blind multichannel identification (ABMCI) based TDOA estimation.

The generalization of blind SIMO identification from two channels to multiple (> 2) channels is not straightforward and in [118] a systematic way was proposed. Consider a SIMO system with N channels whose outputs are described by (9.10). Each pair of the system outputs has a cross-relation in the absence of noise:

$$\mathbf{y}_i^T(k)\mathbf{g}_j = \mathbf{y}_j^T\mathbf{g}_i, \quad i,j = 1,2,\ldots,N. \tag{9.110}$$

When noise is present or the channel impulse responses are improperly modeled, the cross-relation does not hold and an *a priori* error signal can be defined as follows:

$$e_{ij}(k+1) = \frac{\mathbf{y}_i^T(k+1)\hat{\mathbf{g}}_j(k) - \mathbf{y}_j^T(k+1)\hat{\mathbf{g}}_i(k)}{\|\hat{\mathbf{g}}(k)\|}, \quad i,j = 1,2,\ldots,N, \tag{9.111}$$

where $\hat{\mathbf{g}}_i(k)$ is the model filter for the ith channel at time k and

$$\hat{\mathbf{g}}(k) = \left[\hat{\mathbf{g}}_1^T(k)\ \hat{\mathbf{g}}_2^T(k)\cdots\hat{\mathbf{g}}_N^T(k)\right]^T.$$

The model filters are normalized in order to avoid a trivial solution whose elements are all zeros. Based on the error signal defined here, a cost function at time $k+1$ is given by

$$J(k+1) = \sum_{i=1}^{N-1}\sum_{j=i+1}^{N} e_{ij}^2(k+1). \tag{9.112}$$

The multichannel LMS (MCLMS) algorithm updates the estimate of the channel impulse responses as follows:

$$\hat{\mathbf{g}}(k+1) = \hat{\mathbf{g}}(k) - \mu \nabla J(k+1), \tag{9.113}$$

where μ is again a small positive step size. As shown in [118], the gradient of $J(k+1)$ is computed as

$$\nabla J(k+1) = \frac{\partial J(k+1)}{\partial \hat{\mathbf{g}}(k)} = \frac{2\left[\bar{\mathbf{R}}_{y+}(k+1)\hat{\mathbf{g}}(k) - J(k+1)\hat{\mathbf{g}}(k)\right]}{\|\hat{\mathbf{g}}(k)\|^2}, \tag{9.114}$$

where

$$\bar{\mathbf{R}}_{y+}(k) = \begin{bmatrix} \sum\limits_{n\neq 1} \bar{\mathbf{R}}_{y_n y_n}(k) & -\bar{\mathbf{R}}_{y_2 y_1}(k) & \cdots & -\bar{\mathbf{R}}_{y_N y_1}(k) \\ -\bar{\mathbf{R}}_{y_1 y_2}(k) & \sum\limits_{n\neq 2} \bar{\mathbf{R}}_{y_n y_n}(k) & \cdots & -\bar{\mathbf{R}}_{y_N y_2}(k) \\ \vdots & \vdots & \ddots & \vdots \\ -\bar{\mathbf{R}}_{y_1 y_N}(k) & -\bar{\mathbf{R}}_{y_2 y_N}(k) & \cdots & \sum\limits_{n\neq N} \bar{\mathbf{R}}_{y_n y_n}(k) \end{bmatrix},$$

and

$$\bar{\mathbf{R}}_{y_i y_j}(k) = \mathbf{y}_i(k)\mathbf{y}_j^T(k), \quad i,j = 1,2,\ldots,N.$$

If the model filters are always normalized after each update, then a simplified MCLMS algorithm is obtained

$$\hat{\mathbf{g}}(k+1) = \frac{\hat{\mathbf{g}}(k) - 2\mu\left[\bar{\mathbf{R}}_{y+}(k+1)\hat{\mathbf{g}}(k) - J(k+1)\hat{\mathbf{g}}(k)\right]}{\left\|\hat{\mathbf{g}}(k) - 2\mu\left[\bar{\mathbf{R}}_{y+}(k+1)\hat{\mathbf{g}}(k) - J(k+1)\hat{\mathbf{g}}(k)\right]\right\|}. \tag{9.115}$$

A number of other adaptive blind SIMO identification algorithms were also developed with faster convergence and lower computational complexity, e.g., [119], [122]. But we would like to refer the reader to [125] and references therein for more details.

After the adaptive algorithm converges, the TDOA τ is determined as

$$\hat{\tau}^{\mathrm{ABMCI}} = \arg\max_l |\hat{g}_{1,l}| - \arg\max_l |\hat{g}_{2,l}|. \tag{9.116}$$

More generally, the TDOA between any two microphones can be inferred as

$$\hat{\tau}_{ij}^{\mathrm{ABMCI}} = \arg\max_l |\hat{g}_{i,l}| - \arg\max_l |\hat{g}_{j,l}|, \quad i,j = 1,2,\ldots,N, \tag{9.117}$$

where we have assumed that in every channel the direct path is always dominant. This is generally true for acoustic waves, which would be considerably attenuated by wall reflection. But sometimes two or more reverberant signals via multipaths of equal delay could add coherently such that the direct-path component no longer dominates the impulse response. Therefore a more robust

way to pick the direct-path component is to identify the Q ($Q > 1$) strongest elements in the impulse responses and choose the one with the smallest delay [120]:

$$\hat{\tau}_{ij}^{\text{ABMCI}} = \min\left[\arg\max_{l}{}^{q}|\hat{g}_{i,l}|\right] - \min\left[\arg\max_{l}{}^{q}|\hat{g}_{j,l}|\right], \quad i,j = 1,2,\ldots,N,\ q = 1,2,\ldots,Q, \tag{9.118}$$

where $\max^q$ computes the qth largest element.

9.11 TDOA Estimation of Multiple Sources

So far, we have assumed that there is only one source in the sound field. In many applications such as teleconferencing and telecollaboration, there may be multiple sound sources active at the same time. In this section, we consider the problem of TDOA estimation for the scenarios where there are more than one source in the array's field of view. Fundamentally, the TDOA estimation in such situations consists of two steps, i.e., determining the number of sources, and estimating the TDOA due to each sound source. Here we assume that the number of sources is known *a priori*, so we focus our discussion on the second step only.

Many algorithms discussed in Sections 9.3–9.8 can be used or extended for TDOA estimation of multiple sources. Let us take, for example, the CC method. When there are two sources, using the signal model given in (9.8), we can write the CCF between $y_1(k)$ and $y_2(k)$ as

$$\begin{aligned} r_{y_1y_2}^{\text{CC}}(p) = {} & \alpha_{11}\alpha_{21}r_{s_1s_1}^{\text{CC}}(p-\tau_1) + \alpha_{11}\alpha_{22}r_{s_1s_2}^{\text{CC}}(t_1+p-t_2-\tau_2) + \\ & \alpha_{11}r_{s_1v_2}^{\text{CC}}(p+t_1) + \alpha_{12}\alpha_{21}r_{s_2s_1}^{\text{CC}}(p+t_2-t_1-\tau_1) + \\ & \alpha_{12}\alpha 22r_{s_2s_2}^{\text{CC}}(p-\tau_2) + \alpha_{12}r_{s_2v_2}^{\text{CC}}(p+t_2) + \\ & \alpha_{2,1}r_{v_1s_1}^{\text{CC}}(p-t_1-\tau_1) + \alpha_{22}r_{v_1s_2}^{\text{CC}}(p-t_2-\tau_2) + \\ & r_{v_1v_2}^{\text{CC}}(p). \end{aligned} \tag{9.119}$$

Noting that the source signals are assumed to be mutually independent with each other and the noise signal at one sensor is assumed to be uncorrelated with the source signals and the noise at the other microphones, we get

$$r_{y_1y_2}^{\text{CC}}(p) = \alpha_{11}\alpha_{21}r_{s_1s_1}^{\text{CC}}(p-\tau_1) + \alpha_{12}\alpha_{22}r_{s_2s_2}^{\text{CC}}(p-\tau_2). \tag{9.120}$$

The two correlation functions $r_{s_1s_1}^{\text{CC}}(p-\tau_1)$ and $r_{s_2s_2}^{\text{CC}}(p-\tau_2)$ will reach their respective maximum at $p = \tau_1$ and $p = \tau_2$. Therefore, we should expect to see two large peaks in $r_{y_1y_2}^{\text{CC}}(p)$, each corresponding to the TDOA of one source. The same result applies to all the GCC methods [26], [27].

To illustrate the TDOA estimation of two sources using the correlation based method, we consider the simulation example used in Section 9.5 except

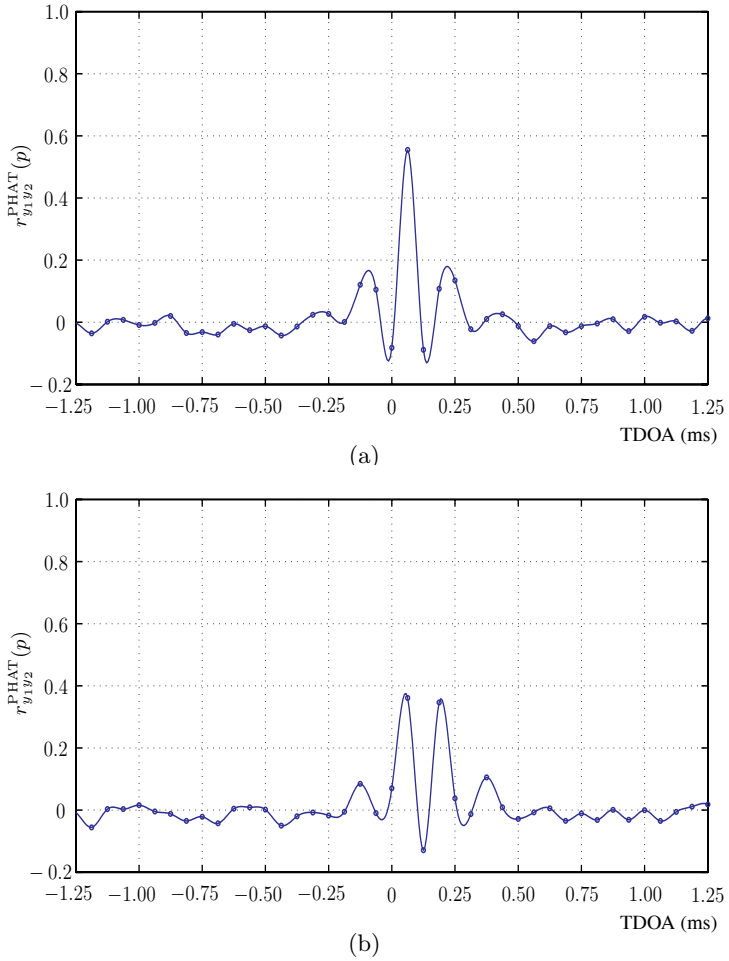


Fig. 9.8. The CCF computed using the PHAT algorithm: (a) there is only one source at $\theta = 75.5^\circ$ and (b) there are two source at $\theta_1 = 75.5^\circ$ and $\theta_2 = 41.4^\circ$ respectively. The microphone noise is white Gaussian with SNR = 10 dB. The sampling frequency is 16 kHz.

that now we have two sources in the far-field and their incident angles are $\theta_1 = 75.5^\circ$ and $\theta_2 = 41.4^\circ$ respectively. Figure 9.8 plots the CCF computed using the PHAT algorithm. We see from Fig. 9.8(b) that there are two large peaks corresponding to the two true TDOAs. However, if we compare Figs. 9.8(b) and (a), one can see that the peaks in the two-source situation are not defined as well as the peak for the single-source scenario. This result should not come as a surprise. From (9.120), we see that the two correlation functions $r^{\mathrm{CC}}_{s_1 s_1}(p - \tau_1)$ and $r^{\mathrm{CC}}_{s_2 s_2}(p - \tau_2)$ interfere with each other. So, one source will behave like noise to the other source, thereby making the TDOA estimation more difficult.

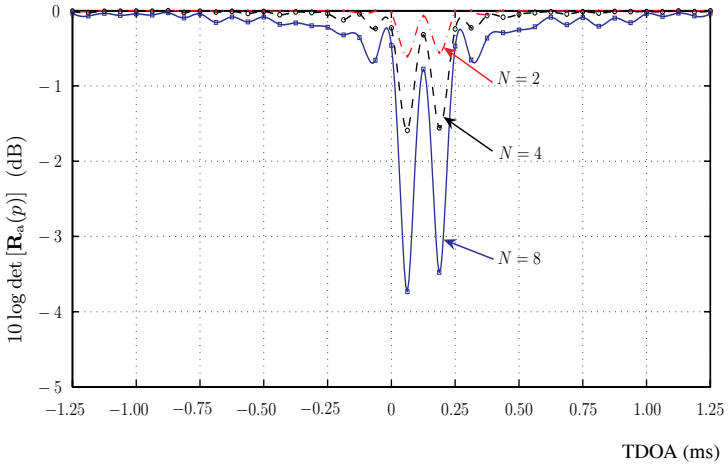


Fig. 9.9. Comparison of det $[\mathbf{R}_a(p)]$ for an equispaced linear array with different number of microphones. There are two source at $\theta_1 = 75.5°$ and $\theta_2 = 41.4°$ respectively. The microphone noise is white Gaussian with SNR = 10 dB. The sampling frequency is 16 kHz.

This problem will become worse as the number of sources in the array's field increases.

Similar to the single-source situation, we can improve the TDOA estimation of multiple sources by increasing the number of microphones. Figure 9.9 plots the cost function computed from the MCCC method with different number of microphones. It is seen that the estimation performance improves with the number of sensors.

The GCC, spatial prediction, MCCC, and entropy based techniques can be directly used to estimate TDOA for multiple sources. The extension of the narrowband MUSIC to the multiple-source situation is also straightforward. Consider the signal model in (9.8) where we neglect the attenuation difference, we have

$$Y_n(f) = \sum_{m=1}^{M} S_m(f) e^{-j2\pi[t_m + \mathcal{F}_n(\tau_m)]f} + V_n(f). \tag{9.121}$$

Following the notation used in Section 9.7.1, we can write $\vec{\mathbf{y}}$ as

$$\vec{\mathbf{y}} = \boldsymbol{\Omega}\vec{\mathbf{s}} + \vec{\mathbf{v}}, \tag{9.122}$$

where

$$\boldsymbol{\Omega} = \left[\boldsymbol{\varsigma}(\tau_1)\ \boldsymbol{\varsigma}(\tau_2) \cdots \boldsymbol{\varsigma}(\tau_M) \right],$$

is a matrix of size $N \times M$, and

$$\vec{\mathbf{s}} = \left[S_1(f)e^{j2\pi t_1 f} \; S_2(f)e^{j2\pi t_2 f} \; \cdots \; S_M(f)e^{j2\pi t_M f} \right]^T.$$

The covariance matrix $\mathbf{R}_Y$ has the form

$$\mathbf{R}_Y = E\left(\vec{\mathbf{y}}\vec{\mathbf{y}}^H\right) = \mathbf{\Omega}\mathbf{R}_S\mathbf{\Omega}^H + \sigma_V^2\mathbf{I}, \tag{9.123}$$

where

$$\mathbf{R}_S = E\left(\vec{\mathbf{s}}\vec{\mathbf{s}}^H\right). \tag{9.124}$$

It is easily seen that the rank of the product matrix $\mathbf{\Omega}\mathbf{R}_S\mathbf{\Omega}^H$ is of M. Therefore, if we perform the eigenvalue decomposition of $\mathbf{R}_Y$ and sort its eigenvalues in descending order, we get

$$\mathbf{\Omega}\mathbf{R}_S\mathbf{\Omega}^H\mathbf{b}_n = \mathbf{0}, \quad n = M+1, \ldots, N, \tag{9.125}$$

where, again, $\mathbf{b}_n$ is the eigenvector associated with the nth eigenvalue of $\mathbf{R}_Y$. This result indicates that

$$\mathbf{b}_n^H\boldsymbol{\varsigma}(\tau_m) = 0, \quad m = 1, 2, \ldots, M, \quad M+1 \leq n \leq N. \tag{9.126}$$

Following the same line of analysis in Section 9.7.1, after the eigenvalue decomposition of $\mathbf{R}_Y$, we can construct the narrowband MUSIC cost function as

$$J_{\text{MUSIC}}(p) = \frac{1}{\sum\limits_{n=M+1}^{N} \left|\mathbf{b}_n^H\boldsymbol{\varsigma}(p)\right|^2}. \tag{9.127}$$

The M largest peaks of $J_{\text{MUSIC}}(p)$ should correspond to the TDOAs τ_m, $m = 1, 2, \ldots, M$.

As we have pointed out earlier, the narrowband MUSIC may not be very useful for microphone array due to the nonstationary nature of speech. The extension of the broadband MUSIC (presented in Section 9.7.2) to multiple-source situation, however, is not straightforward. To see this, let us assume that there are M sources. With some mathematical manipulation, the spatial covariance matrix $\mathbf{R}_{\text{a}}(p)$ can be written as

$$\mathbf{R}_{\text{a}}(p) = \sum_{m=1}^{M} \mathbf{R}_{s_m}(p) + \sigma^2\mathbf{I}. \tag{9.128}$$

Now even when $p = \tau_m$ and $\mathbf{R}_{s_m}(p)$ becomes a matrix of rank 1, the superimposed signal matrix, $\sum_{m=1}^{M} \mathbf{R}_{s_m}(p)$, may still be of rank N. Therefore, the signal and noise subspaces are overlapped and we cannot form a broadband MUSIC algorithm for multiple sources. But in one particular case where all the sources are white, we can still use the estimator in (9.89). In general, for

multiple source TDOA estimation, we would recommend to use the MCCC approach.

Another possible approach for TDOA estimation of multiple sources is to blindly identify the impulse responses of a MIMO system. However, blind MIMO identification is much more difficult than blind SIMO identification, and might be even unsolvable. The research on this problem remains at the state of feasibility investigations. To finish this section, let us mention that recently some algorithms based on the MIMO model of (9.11) have been proposed in [157], [158].

9.12 Conclusions

This chapter presented the problem of DOA and TDOA estimation. We have chosen to focus exclusively on the principles of TDOA estimation since the problem of the DOA estimation is essentially the same as the TDOA estimation. We have discussed the basic idea of TDOA estimation based on the generalized cross-correlation criterion. In practice, the estimation problem can be seriously complicated by noise and reverberation. In order to improve the robustness of TDOA estimation with respect to distortions, we have discussed two basic approaches: exploiting the fact that we can have multiple microphones and using a more practical reverberant signal model, which resulted to a wide range of algorithms such as the spatial prediction, multichannel cross-correlation, minimum entropy, and adaptive blind channel identification techniques. Also discussed in this chapter were the principles for TDOA estimation of multiple sources.

10

Unaddressed Problems

10.1 Introduction

Microphone array signal processing is a technical domain where traditional speech and array processing meet. The primary goal is to enhance and extract information carried by acoustic waves received by a number of microphones at different positions. Due to the random, broadband, and nonstationary essence of speech and the presence of room reverberation, microphone array signal processing is not only a very broad but also a very complicated topic. Most, if not all, of the array signal processing algorithms need to be re-developed and specially tailored for the problems with the use of microphone arrays. Therefore, one cannot expect that one book could and should cover all these problems. As a matter of fact, in this area and every year, a great number of Ph.D. dissertations and numerous journal papers are produced. A key thing that we want to demonstrate to the readers is how an algorithm can be developed to properly process broadband speech signals no matter whether they propagate from far-field or near-field sources. We selected those problems for which we achieved promising results in our research as examples. But the unaddressed problems in microphone array signal processing are also important. They are either still open for research or better discussed in other books. In the following, we will briefly describe the state of the art of three unaddressed problems and provide useful references to help the reader for further detailed studies.

10.2 Speech Source Number Estimation

Microphone arrays, as a branch of array signal processing, offer an effective approach to extending the sense of hearing of human beings. Genetically, enhancement of acoustic signals from desired sound sources and separation of an interested acoustic signal (either speech or non-speech audio) from other competing interference are the primary goals. But whether these goals can be

satisfactorily achieved depends not only on speech enhancement and separation algorithms themselves (as evidenced by the great efforts made in most, if not all, parts of this book for their advancement), but also on an array's capability of characterizing its surrounding acoustic environment. Such a characterization, sometimes termed as auditory scene analysis (ASA), includes determination of the number of active sound sources, localization and tracking of these sources, and the like technologies. Speech source number estimation is an important problem since many of the algorithms for processing microphone array signals make the assumption that the number of sources is known *a priori* and may give misleading results if the wrong number of sources is used. A good example is the failure of a blind source separation algorithm when the wrong number of sources is assumed. While acoustic source localization and tracking have been discussed in the previous chapter, no section was devoted to the problem of speech source number estimation in this book. This is not because speech source number estimation is easy to solve, but on the contrary, because it is a real challenge that is still open for research in practice.

Determining the number of sources is a traditional problem in array signal processing for radar, sonar, communications, and geophysics. A common formulation is to compute the spatial correlation matrix of the *narrowband* outputs of the sensor array [135]. The spatial correlation matrix can be decomposed into the signal-plus-noise and noise-only subspaces using eigenvalue decomposition. The number of sources is equal to the dimension of the signal-plus-noise subspace, which can be estimated using either decision theoretic approaches (e.g., the sphericity test [235]) or information theoretic approaches (e.g., the Akaike information criterion (AIC) [3], and the minimum description length (MDL) [190], [201]) – the reader can also refer to [227], [236], [237], [239], and the references therein for more information. These subspace analysis methods perform reasonably well, but only for narrowband signals. In microphone arrays, speech is broadband and nonstationary. The preliminary results from our own research on this problem indicates that a straightforward application of the traditional source number estimation approaches to microphone arrays by choosing an arbitrary frequency for testing produces little success, if it is not completely useless. A possible direction for improvement is to re-define the original subspace analysis framework such that processing at multiple frequency bins can be carried out, and meanwhile exploit the knowledge about unique speech characteristics to help address such questions as how many and which frequency bins should be examined.

10.3 Cocktail Party Effect and Blind Source Separation

It has been recognized for some time that a human has the ability of focusing on one particular voice or sound amid a cacophony of distracting conversations and background noise. This interesting psychoacoustic phenomenon is

referred to as the *cocktail party effect* or *attentional selectivity*. The cocktail party problem was first investigated by Colin Cherry in his pioneering work published in 1953 [45] and has since been studied in a large variety of diverse fields: psychoacoustics, neuroscience, signal processing, and computer science (in particular human-machine interface). Due to the apparently differing interests and theoretical values in these different domains, the cocktail party effect is explored from different perspectives. These efforts can be broadly categorized as addressing the following two questions:

1. how do the human auditory system and the brain solve the cocktail party problem?
2. can we replicate the ability of the cocktail party effect for man-made intelligent systems?

While a comprehensive understanding of the cocktail party effect that is gained from the first efforts will certainly help tackle the second problem (which can be referred to as the computational cocktail party problem), it does not mean that we have to replicate every aspect of the human auditory system or we have to exactly follow every step of the acoustic perception procedure in solving the computational cocktail party problem. However, although only a simplified solution is pursued and the problem has been continuously investigated for a number of decades, no existing systems or algorithms can convincingly allow us to claim or just foresee a victory. In particular, we do not have all the necessary know-hows in order to provide a recipe in this book that a computer programmer can readily follow to build an intelligent acoustic interface working properly in a reverberant, cocktail-party-like environment.

Microphone array beamforming is one of the focuses of this book. A beamformer is a spatial filter that enhances the signal coming from one direction while suppressing interfering speech or noise signals coming from other directions. Apparently a primary requirement for beamforming is that the directions of the sound sources (at least the source of interest) need to be known in advance or pre-estimated from the microphone observations. Therefore traditional microphone array beamforming is a typical example of the class of the computational auditory scene analysis (CASA) approach aimed to solve the computational cocktail party problem. Proceeding by steps, the CASA first detects and classifies sound sources by their low-level spatial locations in addition to spectro-temporal structures, and then performs an unblind, supervised decomposition of the auditory scene.

Blind source separation (BSS) by independent component analysis (ICA) is another class of approaches to the computational cocktail party problem, but is not covered in this book. BSS/ICA assumes that an array of microphones records linear mixtures of unobserved, *statistically independent* source signals. A linear de-mixing system is employed to process the microphone signals such that an independence measure of the separated signals is maximized. Since there is no available information about the way in which the source signals are mixed, the de-mixing procedure is carried out in a blind

(i.e., unsupervised) manner. ICA was first introduced by Herault, Jutten, and Ans in 1985 [104] (a paper in French) and has quickly blossomed into an important area of the ever-expanding discipline of statistical signal processing. In spite of the swift popularity of ICA, its proven effectiveness is mainly limited in the cases of instantaneous mixtures. When convolutive mixtures are concerned as encountered in almost all speech-related applications, a common way is to use the discrete Fourier transform and transform the time-domain convolutive mixtures into a number of instantaneous mixtures in the frequency domain [65], [202], [207]. ICA is then performed independently at each frequency bin with respect to the instantaneous mixtures. It is noteworthy that independent subband source signals in an instantaneous mixture can at best be blindly separated up to a scale and a permutation. This results in the possibility that a recovered fullband, time-domain signal is not a consistent estimate of one of the source signals over all frequencies, which is known as the permutation inconsistency problem [129], [202]. The degradation of speech quality caused by the permutation ambiguity is only slightly noticeable when the length of the mixing channels is short. The impact becomes more evident when the channels are longer in reverberant environments. Although a number of methods were proposed to align permutations of the de-mixing filters over all the frequency bins [130], [169], [180], [194], [196], [202], this is still an open problem under active research. In addition, in the human cocktail party effect, we only separate the source signal of interest from the competing signals. But the BSS/ICA try to calculate estimates of all the source signals at a time. Therefore, we choose not to include the development of BSS in this book but would like to refer the interested reader to a very recent review of convolutive BSS [182] and the references therein for a deeper, more informative exploration.

10.4 Blind MIMO Identification

In traditional antenna array signal processing, source signals are narrowband and the arrays work in fairly open space. As a result, the channel is relatively flat. Even when multipath exists, the delays between the reflections and the signal coming from the direct path are short. For example, in wireless communications, the channel impulse responses are at longest tens of samples. However, as we mentioned in various places in this book, speech is broadband by nature and a microphone array is used most of the time in an enclosure. Moreover, the human ear has an extremely wide dynamic range and is much more sensitive to weak tails of the channel impulse responses. Consequently, it is not uncommon to model an acoustic channel with an FIR filter of thousands of samples long in microphone array signal processing. Therefore, while system identification may have already been regarded as an off-the-shelf technique in antenna array processing for wireless communication, estimating a very long acoustic impulse response is a real challenge when source signals are

accessible (e.g., multichannel acoustic echo cancellation [10]), and otherwise can be fundamentally intractable or even unsolvable. But unfortunately, for a majority of microphone array applications, source signals are not known and a blind MIMO identification algorithm has to be developed. Needless to explain, the challenges are great, but so are the potential rewards. If we can blindly identify an acoustic MIMO system in practice, the solutions to many difficult acoustic problems become immediately obvious [124].

The innovative idea of identifying a system without reference signals was first proposed by Sato in [195]. Early studies of blind channel identification and equalization focused primarily on higher (than second) order statistics (HOS) based methods. Because HOS cannot be accurately computed from a small number of observations, slow convergence is the critical drawback of all existing HOS methods. In addition, a cost function based on the HOS is barely concave and an HOS algorithm can be misled to a local minimum by corrupting noise in the observations. Therefore, after it was recognized that the problem can be solved in the light of only second-order statistics (SOS) of system outputs [212], the focus of the blind channel identification research has shifted to SOS methods. Using SOS to blindly identify a system requires that the number of sensors would be greater than the number of sources. Hence for a microphone array only the SIMO and MIMO models are concerned.

Blind SIMO identification using only SOS is relatively simple and two necessary and sufficient conditions (one on the channel diversity and the other on the input signals) were clearly given in [238] and as follows:

1. the polynomials formed from the acoustic channel impulse responses are co-prime, i.e., the channel transfer functions do not share any common zeroes;
2. the autocorrelation matrix of the solo input signal is of full rank (such that the SIMO system can be fully excited).

There has been a rich literature on this technique. Not only batch methods [6], [96], [114], [155], [168], [206], [213], [238], but also a number of adaptive algorithms [13], [118], [119], [122] were developed.

On the contrary, blind MIMO identification is still an open research problem. A necessary condition for identifiability using only SOS on the channel impulse responses resembles that for a SIMO system: the transfer functions with respect to the same source signal do not share any common zeros (i.e., the MIMO system is irreducible). But the conditions on the source signals that are sufficient for identifiability using only SOS depend on whether the acoustic channels are memoryless or convolutive. For a memoryless MIMO system, it was shown in [9], [115], and [212] that the uncorrelated source signals must be colored and must have distinct power spectra. But for a convolutive MIMO system, while either colored inputs with distinct power spectra or white, nonstationary inputs can guarantee blind identifiability, no practically realizable algorithm has yet been invented.

Even though this subject is important and may be of interest to a lot of readers, it has been comprehensively discussed in one of the previous books of the same authors [125]. Nevertheless, channel identification is more relevant to microphone array signal processing from a MIMO perspective than from a spatial-filtering perspective. Therefore, we choose not to repeat this subject in this book.

10.5 Conclusions

As a wrapping up, three unaddressed problems, namely, speech source number estimation, cocktail party effect and blind source separation, and blind MIMO identification, were briefly reviewed. The state of the art of these problems was described and we explained why they were not covered in this book. A fairly comprehensive, though not necessarily exhaustive, list of references was supplied to help the interested readers know where they can find useful information on these topics.

References

1. S. Affes, S. Gazor, and Y. Grenier, "Robust adaptive beamforming via LMS-like target tracking," in *Proc. IEEE ICASSP*, 1994, pp. IV-269–272.
2. S. Affes, *Formation de Voie Adaptative en Milieux Réverbérants*. PhD Thesis, Telecom Paris University, France, 1995.
3. H. Akaike, "A new look at the statistical model identification," *IEEE Trans. Autom. Control*, vol. AC-19, pp. 716–723, Dec. 1974.
4. V. R. Algazi and M. Suk, "On the frequency weighted least-square design of finite duction filters," *IEEE Trans. Circuits Syst.*, vol. CAS-22, pp. 943–953, Dec. 1975.
5. S. P. Applebaum, "Adaptive arrays," *IEEE Trans. Antennas Propagat.*, vol. AP-24, pp. 585–598, Sept. 1976.
6. L. A. Baccala and S. Roy, "A new blind time-domain channel identification method based on cyclostationarity," *IEEE Signal Process. Lett.*, vol. 1, pp. 89–91, June 1994.
7. W. Bangs and P. Schultheis, "Space-time processing for optimal parameter estimation," in *Signal Processing*, J. Griffiths, P. Stocklin, and C. Van Schooneveld, eds., New York: Academic Press, 1973, pp. 577–590.
8. B. G. Bardsley and D. A. Christensen, "Bean pattern from pulsed ultrasonic transducers using linear systems theory," *J. Acoust. Soc. Am.*, vol. 69, pp.25–30, Jan. 1981.
9. A. Belouchrani, K. Abed-Meraim, J.-F. Cardoso, and E. Moulines, "A blind source separation technique using second-order statistics," *IEEE Trans. Signal Process.*, vol. 45, pp. 434–444, Feb. 1997.
10. J. Benesty, T. Gaensler, D. R. Morgan, M. M. Sondhi, and S. L. Gay, *Advances in Network and Acoustic Echo Cancellation*. Berlin, Germany: Springer-Verlag, 2001.
11. J. Benesty and Y. Huang, eds., *Adaptive Signal Processing: Applications to Real-World Problems*. Berlin, Germany: Springer-Verlag, 2003.
12. J. Benesty and T. Gaensler, "New insights into the RLS algorithm," *EURASIP J. Applied Signal Process.*, vol. 2004, pp. 331–339, Mar. 2004.
13. J. Benesty, Y. Huang, and J. Chen, "An exponentiated gradient adaptive algorithm for blind identification of sparse SIMO systems," in *Proc. IEEE ICASSP*, 2004, vol. II, pp. 829–832.

14. J. Benesty, J. Chen, and Y. Huang, "Time-delay estimation via linear interpolation and cross-correlation," *IEEE Trans. Speech Audio Process.*, vol. 12, pp. 509–519, Sept. 2004.
15. J. Benesty, J. Chen, Y. Huang, and S. Doclo, "Study of the Wiener filter for noise reduction," in *Speech Enhancement*, J. Benesty, S. Makino, and J. Chen, eds., Berlin, Germany: Springer-Verlag, 2005.
16. J. Benesty, S. Makino, and J. Chen, eds., *Speech Enhancement*. Berlin, Germany: Springer-Verlag, 2005.
17. J. Benesty and T. Gaensler, "Computation of the condition number of a non-singular symmetric Toeplitz matrix with the Levinson-Durbin algorithm," *IEEE Trans. Signal Process.*, vol. 54, pp. 2362–2364, June 2006.
18. J. Benesty, J. Chen, Y. Huang, and J. Dmochowski, "On microhone-array beamforming from a MIMO acoustic signal processing perspective," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, pp. 1053–1065, Mar. 2007.
19. J. Benesty, Y. Huang, and J. Chen, "Time delay estimation via minimum entropy," *IEEE Signal Process. Lett.*, vol. 14, pp. 157–160, Mar. 2007.
20. J. Benesty, M. M. Sondhi, and Y. Huang, eds., *Springer Handbook of Speech Processing*. Berlin, Germany: Springer-Verlag, 2007.
21. J. Benesty, J. Chen, and Y. Huang, "A minimum speech distortion multichannel algorithm for noise reduction," in *Proc. IEEE ICASSP*, to appear, 2008.
22. M. Berouti, R. Schwartz, and J. Makhoul, "Enhancement of speech corrupted by acoustic noise," in *Proc. IEEE ICASSP*, 1979, pp. 208–211.
23. J. Bitzer, K. U. Simmer, and K.-D. Kammeyer, "Theoretical noise reduction limits of the generalized sidelobe canceller (GSC) for speech enhancement," in *Proc. IEEE ICASSP*, 1999, pp. 2965–2968.
24. S. F. Boll, "Suppression of acoustic noise in speech using spectral subtraction," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-27, pp. 113–120, Apr. 1979.
25. M. S. Brandstein, "A pitch-based approach to time-delay estimation of reverberant speech," in *Proc. IEEE WASPAA*, Oct. 1997.
26. M. S. Brandstein and H. F. Silverman, "A practical methodology for speech source localization with microphone arrays," *Comput., Speech, Language*, vol. 2, pp. 91–126, Nov. 1997.
27. M. Brandstein and D. B. Ward, eds., *Microphone Arrays: Signal Processing Techniques and Applications*. Berlin, Germany: Springer-Verlag, 2001.
28. B. R. Breed and J. Strauss, "A short proof of the equivalence of LCMV and GSC beamforming," *IEEE Signal Process. Lett.*, vol. 9, pp. 168–169, June 2002.
29. C. Breining, P. Dreiscitel, E. Hänsler, A. Mader, B. Nitsch, H. Puder, T. Schertler, G. Schmidt, and J. Tilp, "Acoustic echo control - an application of very-high-order adaptive filters," *IEEE Signal Process. Mag.*, vol. 16, pp. 42-69, July 1999.
30. M. Buck, T. Haulick, and H.-J. Pfleiderer, "Self-calibrating microphone arrays for speech signal acquisition: a systematic approach," *Signal Process.*, vol. 86, pp. 1230–1238, June 2006.
31. K. M. Buckley and L. J. Griffiths, "An adaptive generalized sidelobe canceller with derivative constraints," *IEEE Trans. Antennas Propagat.*, vol. AP-34, pp. 311–319, Mar. 1986.
32. K. M. Buckley, "Broad-band beamforming and the generalized sidelobe canceller," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, pp. 1322–1323, Oct. 1986.

33. K. M. Buckley, "Spatial/spectral filtering with linearly constrained minimum variance beamformers," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, pp. 249–266, Mar. 1987.
34. W. S. Burdic, *Underwater Acoustic System Analysis*. Englewood Cliffs, NJ: Prentice-Hall, 1984.
35. J. Capon, "High resolution frequency-wavenumber spectrum analysis," *Proc. IEEE*, vol. 57, pp. 1408–1418, Aug. 1969.
36. G. C. Carter, A. H. Nuttall, and P. G. Cable, "The smoothed coherence transform," *Proc. IEEE*, vol. 61, pp. 1497–1498, Oct. 1973.
37. B. Champagne, S. Bédard, and A. Stéphenne, "Performance of time-delay estimation in presence of room reverberation," *IEEE Trans. Speech Audio Process.*, vol. 4, pp. 148–152, Mar. 1996.
38. G. Chen, S. N. Koh, and I. Y. Soon, "Enhanced Itakura measure incorporating masking properties of human auditory system," *Signal Process.*, vol. 83, pp. 1445–1456, July 2003.
39. J. Chen, J. Benesty, and Y. Huang, "Robust time delay estimation exploiting redundancy among multiple microphones," *IEEE Trans. Speech Audio Process.*, vol. 11, pp. 549–557, Nov. 2003.
40. J. Chen, J. Benesty, and Y. Huang, "Time delay estimation in room acoustic environments: an overview," *EURASIP J. Applied Signal Process.*, vol. 2006, Article ID 26503, 19 pages, 2006.
41. J. Chen, J. Benesty, Y. Huang, and S. Doclo, "New insights into the noise reduction Wiener filter," *IEEE Trans. Audio, Speech, Language Process.*, vol. 14, pp. 1218–1234, July 2006.
42. J. Chen, J. Benesty, and Y. Huang, "On the optimal linear filtering techniques for noise reduction," *Speech Communication*, vol. 49, pp. 305–316, Apr. 2007.
43. J. Chen, J. Benesty, Y. Huang, and E. J. Diethorn, "Fundamentals of noise reduction," in *Springer Handbook of Speech Processing*, J. Benesty, M. M. Sondhi, and Y. Huang, eds., Berlin, Germany: Springer-Verlag, 2007.
44. J. Chen, J. Benesty, and Y. Huang, "A minimum distortion noise reduction algorithm with multiple microphones," *IEEE Trans. Audio, Speech, Language Process.*, to appear, 2008.
45. E. C. Cherry, "Some experiments on the recognition of speech, with one and with two ears," *J. Acoust. Soc. Am.*, vol. 25, pp. 975–979, Sept. 1953.
46. E. C. Cherry and W. L. Taylor, "Some further experiments upon the recognition of speech, with one and with two ears," *J. Acoust. Soc. Am.*, vol. 26, pp. 554–559, July 1954.
47. I. Chiba, T. Takahashi, and Y. Karasawa, "Transmitting null beam forming with beam space adaptive array antennas," in *Proc. IEEE 44th VTC*, 1994, pp. 1498–1502.
48. I. Cohen, "Analysis of two-channel generalized sidelobe canceller (GSC) with post-filtering," *IEEE Trans. Speech Audio Process.*, vol. 11, pp. 684–699, Nov. 2003.
49. R. T. Compton, Jr., "Pointing accuracy and dynamic range in a steered beam adaptive array," *IEEE Trans. Aerospace, Electronic Systems*, vol. AES-16, pp. 280–287, May 1980.
50. R. T. Compton, Jr., "The effect of random steering vector errors in the Applebaum adaptive array," *IEEE Trans. Aerospace, Electronic Systems*, vol. AES-18, pp. 392–400, July 1982.

51. R. T. Compton, Jr., *Adaptive Antennas: Concepts and Performance*. Englewood Cliffs, NJ: Prentice-Hall, 1988.
52. T. M. Cover and J. A. Thomas, *Elements of Information Theory*. New York: John Wiley & Sons, Inc., 1991.
53. H. Cox, "Resolving power and sensitivity to mismatch of optimum array processors," *J. Acoust. Soc. Am.*, vol. 54, pp. 771–785, Mar. 1973.
54. H. Cox, R. M. Zeskind, and M. M. Owen, "Robust adaptive beamforming," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, pp. 1365–1376, Oct. 1987.
55. J. DiBiase, H. Silverman, and M. Brandstein, "Robust localization in reverberant rooms," in *Microphone Arrays: Signal Processing Techniques and Applications*, M. Branstein and D. Ward, eds., Berlin, Germany: Springer, 2001.
56. E. J. Diethorn, "Subband noise reduction methods for speech enhancement," in *Audio Signal Processing for Next-Generation Multimedia Communication Systems*, Y. Huang and J. Benesty, eds., Boston, MA, USA: Kluwer, 2004, pp. 91–115.
57. J. P. Dmochowski, J. Benesty, and S. Affes, "Direction of arrival estimation using the parameterized spatial correlation matrix," *IEEE Trans. Audio, Speech, Language Process.*, vol. 15, pp. 1327–1339, May 2007.
58. J. P. Dmochowski, J. Benesty, and S. Affes, "Broadband MUSIC: opportunities and challenges for multiple source localization," in *Proc. IEEE WASPAA*, 2007, pp. 18–21.
59. S. Doclo and M. Moonen, "GSVD-based optimal filtering for single and multimicrophone speech enhancement," *IEEE Trans. Signal Process.*, vol. 50, pp. 2230–2244, Sept. 2002.
60. S. Doclo, *Multi-Microphone Noise Reduction and Dereverberation Techniques for Speech Applications*. PhD Thesis, Katholieke Universiteit Leuven, Belgium, 2003.
61. S. Doclo and M. Moonen, "Design of far-field and near-field broadband beamformers using eigenfilters," *Signal Process.*, vol. 83, pp. 2641–2673, Dec. 2003.
62. S. Doclo and M. Moonen, "On the output SNR of the speech-distortion weighted multichannel Wiener filter," *IEEE Signal Process. Lett.*, vol. 12, pp. 809–811, Dec. 2005.
63. D. E. Dudgeon, "Fundamentals of digital array porcessing," *Proc. IEEE*, vol. 65, pp. 898–904, June 1977.
64. O. J. Dunn and V. A. Clark, *Applied Statistics: Analysis of Variance and Regression*. New York: Wiley, 1974.
65. F. Ehlers and H. G. Schuster, "Blind separation of convolutive mixtures and an application in automatic speech recognition in a noisy environment," *IEEE Trans. Signal Process.*, vol. 45, pp. 2608–2612, Oct. 1997.
66. Y. C. Eldar, A. Nehorai, and P. S. La Rosa, "A competitive mean-squared error approach to beamforming," *IEEE Trans. Signal Process.*, vol. 55, pp. 5143–5154, Nov. 2007.
67. T. Eltoft, T. Kim, and T.-W. Lee, "On the multivariate Laplace distribution," *IEEE Signal Process. Lett.*, vol. 13, pp. 300–303, May 2006.
68. Y. Ephraim and D. Malah, "Speech enhancement using a minimum mean-square error short-time spectral amplitude estimator," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-32, pp. 1109–1121, Dec. 1984.

69. Y. Ephraim and H. L. Van Trees, "A signal subspace approach for speech enhancement," *IEEE Trans. Speech Audio Process.*, vol. 3, pp. 251–266, July 1995.
70. W. Etter and G. S. Moschytz, "Noise reduction by noise-adaptive spectral magnitude expansion," *J. Audio Eng. Soc.*, vol. 42, pp. 341–349, May 1994.
71. D. R. Fischell and C. H. Coker, "A speech direction finder," in *Proc. IEEE ICASSP*, 1984, pp. 19.8.1–19.8.4.
72. J. L. Flanagan, J. D. Johnson, R. Zahn, and G. W. Elko, "Computer-steered microphone arrays for sound transduction in large rooms," *J. Acoust. Soc. Amer.*, vol. 75, pp. 1508–1518, Nov. 1985.
73. J. L. Flanagan, D. A. Berkley, G. W. Elko, J. E. West, and M. M. Sondhi, "Autodirective microphone systems," *Acustica*, vol. 73, pp. 58–71, Feb. 1991.
74. J. L. Flanagan, A. C. Surendran, and E. Jan, "Spatially selective sound capture for speech and audio processing," *Speech Communication*, vol. 13, pp. 207–222, Jan. 1993.
75. B. Friedlander and B. Porat, "Performance analysis of a null-steering algorithm based on direction-of-arrival estimation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-37, pp. 461–466, Apr. 1989.
76. O. L. Frost, III, "An algorithm for linearly constrained adaptive array processing," *Proc. IEEE*, vol. 60, pp. 926–935, Aug. 1972.
77. K. Fukunaga, *Introduction to Statistical Pattern Recognition*. San Diego, CA: Academic Press, 1990.
78. S. Gannot, D. Burshtein, and E. Weinstein, "Iterative and sequential Kalman filter-based speech enhancement algorithms," *IEEE Trans. Speech Audio Process.*, vol. 6, pp. 373–385, July 1998.
79. S. Gannot, D. Burshtein, and E. Weinstein, "Signal enhancement using beamforming and nonstationarity with applications to speech," *IEEE Trans. Signal Process.*, vol. 49, pp. 1614–1626, Aug. 2001.
80. S. Gannot, D. Burshtein, and E. Weinstein, "Analysis of the power spectral deviation of the general transfer function GSC," *IEEE Trans. Signal Process.*, vol. 52, pp. 1115–1121, Apr. 2004.
81. S. Gannot and I. Cohen, "Adaptive beamforming and postfiltering," in *Springer Handbook of Speech Processing*, J. Benesty, M. M. Sondhi, and Y. Huang, eds., Berlin, Germany: Springer-Verlag, 2007.
82. S. Gannot and A. Yeredor, "The Kalman filter," in *Springer Handbook of Speech Processing*, J. Benesty, M. M. Sondhi, and Y. Huang, eds., Berlin, Germany: Springer-Verlag, 2007.
83. N. D. Gaubitch, M. R. P. Thomas, and P. A. Naylor, "Subband method for multichannel least squares equalization of room transfer functions," in *Proc. IEEE WASPAA*, 2007, pp. 14–17.
84. S. L. Gay and J. Benesty, *Acoustic Signal Processing for Telecommunication*. Boston, MA: Kluwer Academic Publishers, 2001.
85. S. Gazor and W. Zhang, "Speech probability distribution," *IEEE Signal Process. Lett.*, vol. 10, pp. 204–207, July 2003.
86. J. D. Gibson, B. Koo, and S. D. Gray, "Filtering of colored noise for speech enhancement and coding," *IEEE Trans. Signal Process.*, vol. 39, pp. 1732–1742, Aug. 1991.
87. L. C. Godara and A. Cantoni, "Uniqueness and linear independence of steering vectors in array space," *J. Acoust. Soc. Amer.*, vol. 70, pp. 467–475, 1981.

88. L. C. Godara, "Application of antenna arrays to mobile communications, part II: beam-forming and direction-of-arrival considerations," *Proc. IEEE*, vol. 85, pp. 1195–1245, Aug. 1997.
89. G. H. Golub and C. F. Van Loan, *Matrix Computations*. Baltimore, MD: The Johns Hopkins University Press, 1996.
90. J. W. Goodman, *Introduction of Fourier Optics*. New York: McGraw-Hill, 1968.
91. A. Graham, *Kronecker Products and Matrix Calculus: with Applications*. New York: John Wiley & Sons, Inc., 1981.
92. J. E. Greenberg and P. M. Zurek, "Adaptive beamformer performance in reverberation," in *Proc. IEEE WASPAA*, 1991, pp. 101–102.
93. S. M. Griebel and M. S. Brandstein, "Microphone array source localization using realizable delay vectors," in *Proc. IEEE WASPAA*, 2001, pp. 71–74.
94. L. J. Griffiths and C. W. Jim, "An alternative approach to linearly constrained adaptive beamforming," *IEEE Trans. Antennas Propagat.*, vol. AP-30, pp. 27–34, Jan. 1982.
95. L. J. Griffiths and K. M. Buckley, "Quiescent pattern control in linearly constrained adaptive arrays," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-35, pp. 917–926, July 1987.
96. M. I. Gürelli and C. L. Nikias, "A new eigenvector-based algorithm for multichannel blind deconvolution of input colored signals," in *Proc. IEEE ICASSP*, 1993, vol. 4, pp. 448-451.
97. W. R. Hahn and S. A. Tretter, "Optimum processing for delay-vector estimation in passive signal arrays," *IEEE Trans. Inform. Theory*, vol. IT-19, pp. 608–614, May 1973.
98. E. Hänsler and G. Schmidt, *Acoustic Echo and Noise Control: A Practical Approach*. Hoboken, NJ: John Wiley & Sons, 2004.
99. E. Hänsler and G. Schmidt, eds., *Topics in Acoustic Echo and Noise Control*. Berlin, Germany: Springer-Verlag, 2006.
100. J. H. L. Hansen, "Speech enhancement employing adaptive boundary detection and morphological based spectral constraints," in *Proc. IEEE ICASSP*, 1991, pp. 901–904.
101. A. Härmä, "Acoustic measurement data from the varechoic chamber," Technical Memorandum, Agere Systems, Nov. 2001.
102. M. H. Hayes, *Statistical Digital Signal Processing and Modeling*. New York: John Wiley & Sons, 1996.
103. S. Haykin, *Adaptive Filter Theory*. Fourth Edition, Upper Saddle River, NJ: Prentice-Hall, 2002.
104. J. Herault, C. Jutten, and B. Ans, "Detection de grandeurs primitives dans un message composite par une architecture de calul neuromimetique un apprentissage non supervise," in *Proc. GRETSI*, 1985.
105. W. Herbordt and W. Kellermann, "Adaptive beamforming for audio signal acquisition," in *Adaptive Signal Processing: Applications to Real-World Problems*, J. Benesty and Y. Huang, eds., Berlin, Germany: Springer-Verlag, 2003.
106. W. Herbordt, *Combination of Robust Adaptive Beamforming with Acoustic Echo Cancellation for Acoustic Human/Machine Interfaces*. PhD Thesis, Erlangen-Nuremberg University, Germany, 2004.
107. K. Hermus, P. Wambacq, and H. Van hamme, "A review of signal subspace speech enhancement and its application to noise robust speech recognition," *EURASIP J. Advances Signal Process.*, vol. 2007, Article ID 45821, 15 pages, 2007.

108. M. W. Hoffman and K. M. Buckley, "Robust time-domain processing of broadband microphone array data," *IEEE Trans. Speech Audio Process.*, vol. 3, pp. 193–203, May 1995.
109. O. Hoshuyama, A. Sugiyama, and A. Hirano, "A robust adaptive beamformer for microphone arrays with a blocking matrix using constrained adaptive filters," *IEEE Trans. Signal Process.*, vol. 47, pp. 2677–2684, Oct. 1999.
110. P. W. Howells, "Explorations in fixed and adaptive resolution at GE and SURC," *IEEE Trans. Antennas Propagat.*, vol. AP-24, pp. 575–584, Sept. 1976.
111. Y. Hu and P. C. Loizou, "A subspace approach for enhancing speech corrupted by colored noise," *IEEE Signal Process. Lett.*, vol. 9, pp. 204–206, July 2002.
112. Y. Hu and P. C. Loizou, "A subspace approach for enhancing speech corrupted by colored noise," in *Proc. IEEE ICASSP*, 2002, pp. I-573–I-576.
113. Y. Hu and P. C. Loizou, "A generalized subspace approach for enhancing speech corrupted by colored noise," *IEEE Trans. Speech Audio Process.*, vol. 11, pp. 334–341, July 2003.
114. Y. Hua, "Fast maximum likelihood for blind identification of multiple FIR channels," *IEEE Trans. Signal Process.*, vol. 44, pp. 661–672, Mar. 1996.
115. Y. Hua and J. K. Tugnait, "Blind identifiability of FIR-MIMO systems with colored input using second order statistics," *IEEE Signal Process. Lett.*, vol. 7, pp. 348–350, Dec. 2000.
116. Y. Huang, J. Benesty, and G. W. Elko, "Microphone arrays for video camera steering," in *Acoustic Signal Processing for Telecommunication*, S. L. Gay and J. Benesty, eds., Boston, MA: Kluwer Academic Publishers, chap. 11, pp. 239–259, 2000.
117. Y. Huang, J. Benesty, G. W. Elko, and R. M. Mersereau, "Real-time passive source localization: an unbiased linear-correction least-squares approach," *IEEE Trans. Speech Audio Process.*, vol. 9, pp. 943–956, Nov. 2001.
118. Y. Huang and J. Benesty, "Adaptive multi-channel least mean square and Newton algorithms for blind channel identification," *Signal Process.*, vol. 82, pp. 1127–1138, Aug. 2002.
119. Y. Huang and J. Benesty, "A class of frequency-domain adaptive approaches to blind multi-channel identification," *IEEE Trans. Signal Process.*, vol. 51, pp. 11–24, Jan. 2003.
120. Y. Huang and J. Benesty, "Adaptive multichannel time delay estimation based on blind system identification for acoustic source localization," in *Adaptive Signal Processing: Applications to Real-World Problems*, J. Benesty and Y. Huang, eds., Berlin, Germany: Springer, 2003.
121. Y. Huang and J. Benesty, eds., *Audio Signal Processing for Next-Generation Multimedia Communication Systems*. Boston, MA: Kluwer Academic Publishers, 2004.
122. Y. Huang, J. Benesty, and J. Chen, "Optimal step size of the adaptive multichannel LMS algorithm for blind SIMO identification," *IEEE Signal Process. Lett.*, vol. 12, pp. 173–176, Mar. 2005.
123. Y. Huang, J. Benesty, and J. Chen, "A blind channel identification-based two-stage approach to separation and dereverberation of speech signals in a reverberant environment," *IEEE Trans. Speech Audio Process.*, vol. 13, pp. 882–895, Sept. 2005.
124. Y. Huang, J. Benesty, and J. Chen, "Identification of acoustic MIMO systems: challenges and opportunities," *Signal Process.*, vol. 86, pp. 1278–1295, June 2006.

125. Y. Huang, J. Benesty, and J. Chen, *Acoustic MIMO Signal Processing*. Berlin, Germany: Springer-Verlag, 2006.
126. Y. Huang, J. Benesty, and J. Chen, "Dereverberation," in *Springer Handbook of Speech Processing*, J. Benesty, M. M. Sondhi, and Y. Huang, eds., Berlin, Germany: Springer, 2007.
127. A. Hyvärinen, J. Karhunen, and E. Oja, *Independent Component Analysis*. London, England: John Wiley & Sons, 2001.
128. J. P. Ianniello, "Time delay estimation via cross-correlation in the presence of large estimation errors," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-30, pp. 998–1003, Dec. 1982.
129. M. Z. Ikram and D. R. Morgan, "Exploring permutation inconsistency in blind separation of speech signals in a reverberant environments," in *Proc. IEEE ICASSP*, 2000, pp. 1041–1044.
130. M. Z. Ikram and D. R. Morgan, "Permutation inconsistency in blind speech separation: investigation and solutions," *IEEE Trans. Speech Audio Process.*, vol. 13, pp. 1–13, Jan. 2005.
131. F. Itakura and S. Saito, "A statistical method for esimation fo speech spectral density and formant frequencies," *Electron. Commun. Japan*, vol. 53A, pp. 36–43, 1970.
132. S. H. Jensen, P. C. Hansen, S. D. Hansen, and J. A. Sorensen, "Reduction of broad-band noise in speech by truncated QSVD," *IEEE Trans. Speech Audio Process.*, vol. 3, pp. 439–448, Nov. 1995.
133. C. W. Jim, "A comparison of two LMS constrained optimal array structures," *Proc. IEEE*, vol. 65, pp. 1730–1731, Dec. 1977.
134. D. H. Johnson, "The application of spectral estimation methods to bearing estimation problems," *Proc. IEEE*, vol. 70, pp. 1018–1028, Sept. 1982.
135. D. H. Johnson and D. E. Dudgeon, *Array Signal Processing–Concepts and Techniques*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
136. T. Kailath, "A view of three decades of linear filtering theory," *IEEE Trans. Inform. Theory*, vol. IT-20, pp. 146–181, Mar. 1974.
137. R. E. Kalman, "A new approach to linear filtering and prediction problems," *Trans. ASME, J. Basic Eng.*, ser. D, vol. 82, pp. 35–45, Mar. 1960.
138. R. E. Kalman and R. S. Bucy, "New results in linear filtering and prediction theory," *Trans. ASME, J. Basic Eng.*, ser. D, vol. 83, pp. 95–108, Mar. 1961.
139. R. E. Kalman, "New methods and results in linear filtering and prediction theory," in *Proc. Symp. on Engineering Applications of Probability and Random Functions*, 1961.
140. S. Kay, "Some results in linear interpolation theory," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, pp. 746–749, June 1983.
141. S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ: Prentice-Hall, 1993.
142. W. Kellermann, "A self-steering digital microphone array," in *Proc. IEEE ICASSP*, 1991, vol. 5, pp. 3581–3584.
143. B. E. D. Kingsbury and N. Morgan, "Recognizing reverberant speech with RASTA-PLP," in *Proc. IEEE ICASSP*, 1997, vol. 2, pp. 1259–1262.
144. R. L. Kirlin, D. F. Moore, and R. F. Kubichek, "Improvement of delay measurements from sonar arrays via sequential state estimation," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-29, pp. 514–519, June 1981.

145. C. H. Knapp and G. C. Carter, "The generalized correlation method for estimation of time delay," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 24, pp. 320–327, Aug. 1976.
146. I. Kojadinovic, "On the use of mutual information in data analysis: an overview," in *International Symposium on Applied Stochastic Models and Data Analysis*, 2005.
147. S. Kotz, T. J. Kozubowski, and K. Podgórski, "An asymmetric multivariate Laplace distribution," Technical Report No. 367, Department of Statistics and Applied Probability, University of California at Santa Barbara, 2000.
148. H. Krim and M. Viberg, "Two decades of array signal processing research: the parametric approach," *IEEE Signal Process. Mag.*, vol. 13, pp. 67–94, July 1996.
149. R. T. Lacoss, "Data adaptive spectral analysis methods," *Geophysics*, vol. 36, pp. 661–675, Aug. 1971.
150. B. Lee, K. Y. Lee, and S. Ann, "An EM-based approach for parameter enhancement with an application to speech signals," *Signal Process.*, vol. 46, pp. 1–14, Sept. 1995.
151. H. Lev-Ari and Y. Ephraim, "Extension of the signal subspace speech enhancement approach to colored noise," *IEEE Signal Process. Lett.*, vol. 10, pp. 104–106, Apr. 2003.
152. N. Levinson, "The Wiener rms (root-mean-square) error criterion in filter design and prediction," *J. Math. Phy.*, vol. 25, pp. 261–278, Jan. 1947.
153. J. S. Lim and A. V. Oppenheim, "Enhancement and bandwidth compression of noisy speech," *Proc. IEEE*, vol. 67, pp. 1586–1604, Dec. 1979.
154. J. S. Lim, ed., *Speech Enhancement*. Englewood Cliffs, NJ: Prentice-Hall, 1983.
155. H. Liu, G. Xu, and L. Tong, "A deterministic approach to blind equalization," in *Proc. 27th Asilomar Conf. on Signals, Systems, and Computers*, 1993, vol. 1, pp. 751–755.
156. P. Loizou, *Speech Enhancement: Theory and Practice*. Boca Raton, FL: CRC Press, 2007.
157. A. Lombard, H. Buchner, and W. Kellermann, "Multidimensional localization of multiple sound sources using blind adaptive MIMO system identification," in *Proc. IEEE Int. Conf. on Multisensor Fusion and Integration for Intelligent Systems (MFI)*, 2006.
158. A. Lombard, H. Buchner, and W. Kellermann, "Improved wideband blind adaptive system identification using decorrelation filters for the localization of multiple speakers," in *Proc. IEEE ISCAS*, 2007.
159. S. Makino, T.-W Lee, and H. Sawada, eds., *Blind Speech Separation*. Berlin, Germany: Springer-Verlag, 2007.
160. C. Marro, Y. Mahieux, and K. U. Simmer, "Analysis of noise reduction and dereverberation techniques based on microphone arrays with postfiltering," *IEEE Trans. Speech Audio Process.*, vol. 6, pp. 240–259, May 1998.
161. R. J. McAulay and M. L. Malpass, "Speech enhancement using a soft-decision noise suppression filter," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-28, pp. 137–145, Apr. 1980.
162. I. A. McCowan, *Robust Speech Recognition using Microphone Arrays*. PhD Thesis, Queensland University of Technology, Australia, 2001.
163. J. Meyer and G. W. Elko, "A highly scalable spherical microphone array based on an orthonormal decomposition of the soundfield," in *Proc. IEEE ICASSP*, 2002, pp. 1781–1784.

164. J. Meyer and G. W. Elko, "Spherical microphone arrays for 3D sound recording," in *Audio Signal Processing for Next-Generation Multimedia Communication Systems*, Y. Huang and J. Benesty, eds., Boston, MA: Kluwer Academic Publishers, 2004.
165. U. Mittal and N. Phamdo, "Signal/noise KLT based approach for enhancing speech degraded by colored noise," *IEEE Trans. Speech Audio Process.*, vol. 8, pp. 159–167, Mar. 2000.
166. M. Miyoshi and Y. Kaneda, "Inverse filtering of room acoustics," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-36, pp. 145–152, Feb. 1988.
167. R. A. Monzingo and T. W. Miller, *Introduction to Adaptive Arrays*. Raleigh, NC: SciTech, 2004.
168. E. Moulines, P. Duhamel, J.-F. Cardoso, and S. Mayrargue, "Subspace methods for the blind identification of multichannel FIR filters," *IEEE Trans. Signal Process.*, vol. 43, pp. 516–525, Feb. 1995.
169. N. Murata, S. Ikeda, and A. Ziehe, "An approach to blind source separation based on temporal structure of speech signals," *Neurocomputing*, vol. 41, pp. 1–24, Oct. 2001.
170. A. K. Nábělek, "Communication in noisy and reverberant environments," in *Acoustical Factors Affecting Hearing Aid Performance*, G. A. Studebaker and I. Hochberg, eds., 2nd ed., Needham Height, MA: Allyn and Bacon, 1993.
171. S. T. Neely and J. B. Allen, "Invertibility of a room impulse response," *J. Acoust. Soc. Am.*, vol. 68, pp. 165–169, July 1979.
172. T. Nishiura, T. Yamada, S. Nakamura, and K. Shikano, "Localization of multiple sound sources based on a CSP analysis with a microphone array," in *Proc. IEEE ICASSP*, 2000, pp. 1053–1055.
173. M. Okuda, M. Ikehara, and S. Takahashi, "Fast and stable least-squares approach for the design of linear phase FIR filters," *IEEE Trans. Signal Process.*, vol. 46, pp. 1485–1493, June 1998.
174. M. Omologo and P. Svaizer, "Acoustic event localization using a crosspower-spectrum phase based technique," in *Proc. IEEE ICASSP*, 1994, vol. 2, pp. 273–276.
175. M. Omologo and P. Svaizer, "Acoustic source location in noisy and reverberant environment using CSP analysis," in *IEEE ICASSP*, 1996, vol. 2, pp. 921–924.
176. A. Oppenheim, A. Willsky, and H. Nawab, *Signals and Systems*. Upper Saddle River, NJ: Prentice Hall, 1996.
177. A. V. Oppenheim, R. W. Schafer, and J. R. Buck, *Dicrete-Time Signal Processing*. Second Edition, Upper Saddle River, NJ: Prentice Hall, 1998.
178. N. Owsley, "Sonar array processing," in *Array Signal Processing*, S. Haykin, ed., Englewood Cliffs, NJ: Prentice-Hall, 1984.
179. K. K. Paliwal and A. Basu, "A speech enhancement method based on Kalman filtering," in *Proc. IEEE ICASSP*, 1987, pp. 177–180.
180. L. C. Parra and C. Spence, "Convolutive blind separation of non-stationary sources," *IEEE Trans. Speech Audio Process.*, vol. 8, pp. 320–327, May 2000.
181. K. Pearson, "Mathematical contributions to the theory of evolution.–III. Regression, heredity and panmixia," *Philos. Trans. Royal Soc. London*, Ser. A, vol. 187, pp. 253–318, 1896.
182. M. S. Pedersen, J. Larsen, U. Kjems, and L. C. Parra, "Convolutive blind source separation methods," in *Springer Handbook of Speech Processing*, J. Benesty, M. M. Sondhi, and Y. Huang, eds., Berlin, Germany: Springer-Verlag, 2007.

183. B. Picinbono and J.-M. Kerilis, "Some properties of prediction and interpolation errors," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-36, pp. 525–531, Apr. 1988.
184. V. F. Pisarenko, "The retrieval of harmonics from a covariance functions," *Geophys. J. Royal Astron. Soc.*, vol. 33, pp. 347–366, 1973.
185. S. R. Quackenbush, T. P. Barnwell, and M. A. Clements, *Objective Measures of Speech Quality*. Englewood Cliffs, NJ: Prentice-Hall, 1988.
186. L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*. Englewood Cliffs, NJ: Prentice-Hall, 1978.
187. L. R. Rabiner and B.-H. Juang, *Fundamentals of Speech Recognition*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
188. D. V. Rabinkin, R. J. Ranomeron, J. C. French, and J. L. Flanagan, "A DSP implementation of source location using microphone arrays," in *Proc. SPIE*, vol. 2846, 1996, pp. 88–99.
189. A. Rezayee and S. Gazor, "An adaptive KLT approach for speech enhancement," *IEEE Trans. Speech Audio Process.*, vol. 9, pp. 87–95, Feb. 2001.
190. J. Rissanen, "Modeling by shortest data description," *Automatica*, vol. 14, pp. 465–471, Sept. 1978.
191. J. L. Rodgers and W. A. Nicewander, "Thirteen ways to look at the correlation coefficient," *The Amer. Statistician*, vol. 42, pp. 59–66, Feb. 1988.
192. R. Roy, A. Paulraj, and T. Kailath, "ESPRIT–a subspace rotation appraoch to estimation of parameters of cisoids in noise," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, pp. 1340–1342, Oct. 1986.
193. S. Sandhu and O. Ghitza, "A comparative study of MEL cepstra and EIH for phone classification under adverse conditions," in *Proc. IEEE ICASSP*, 1995, vol. 1, pp. 409–412.
194. H. Saruwatari, S. Kurita, K. Takeda, F. Itakura, T. Nishikawa, and K. Shikano, "Blind source separation combining independent component analysis and beamforming," *EURASIP J. Applied Signal Process.*, vol. 2003, pp. 1135–1146, Nov. 2003.
195. Y. Sato, "A method of self-recovering equalization for multilevel amplitude-modulation," *IEEE Trans. Commun.*, vol. COM-23, pp. 679–682, June 1975.
196. H. Sawada, R. Mukai, S. Araki, and S. Makino, "A robust and precise method for solving the permutation problem of fequency-domain blind source separation," *IEEE Trans. Speech Audio Process.*, vol. 12, pp. 530–538, Sept. 2004.
197. S. A. Schelkunoff, "A mathematical theory of linear arrays," *Bell Syst. Tech. J.*, vol. 22, pp. 80–107, Jan. 1943.
198. R. O. Schmidt, "Multiple emitter location and signal parameter estimation," *IEEE Trans. Antennas Propagat.*, vol. AP-34, pp. 279–280, Mar. 1986.
199. M. R. Schroeder, "Apparatus for suppressing noise and distortion in communication signals," U.S. Patent No. 3,180,936, filed Dec. 1, 1960, issued Apr. 27, 1965.
200. M. R. Schroeder, "Processing of communication signals to reduce effects of noise," U.S. Patent No. 3,403,224, filed May 28, 1965, issued Sept. 24, 1968.
201. G. Schwarz, "Estimating the dimension of a model," *The Annals of Statistics*, vol. 6, pp. 461–464, Mar. 1978.
202. C. Servière, "Feasibility of source separation in frequency domain," in *Proc. IEEE ICASSP*, 1998, vol. 4, pp. 2085–2088.
203. C. E. Shannon, "A mathematical theory of communication," *Bell Syst. Tech. J.*, vol. 27, pp. 379–423, 623–656, 1948.

204. H. F. Silverman, "Some analysis of microphone arrays for speech data analysis," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 35, pp. 1699–1712, Dec. 1987.
205. B. L. Sim, Y. C. Tong, J. S. Chang, and C. T. Tan, "A parametric formulation of the generalized spectral subtraction method," *IEEE Trans. Speech, Audio Process.*, vol. 6, pp. 328–337, July 1998.
206. D. Slock, "Blind fractionally-spaced equalization, prefect reconstruction filerbanks, and multilinear prediction," in *Proc. IEEE ICASSP*, 1994, vol. 4, pp. 585–588.
207. P. Smaragdis, "Efficient blind separation of convolved sound mixtures," in *Proc. IEEE WASPAA*, 1997.
208. M. M. Sondhi, C. E. Schmidt, and L. R. Rabiner, "Improving the quality of a noisy speech signal," *Bell Syst. Techn. J.*, vol. 60, pp. 1847–1859, Oct. 1981.
209. A. Spriet, M. Moonen, and J. Wouters, "Spatially pre-processed speech distortion weighted multi-channel Wiener filtering for noise reduction," *Signal Process.*, vol. 84, pp. 2367–2387, Dec. 2004.
210. P. Stoica and R. L. Moses, *Introduction to Spectral Analysis*. Upper Saddle River, NJ: Prentice-Hall, 1997.
211. C. Sydow, "Broadband beamforming for a microphone array," *J. Acoust. Soc. Am.*, vol. 96, pp. 845–849, Aug. 1994.
212. L. Tong, G. Xu, and T. Kailath, "A new approach to blind identification and equalization of multipath channels," in *Proc. 25th Asilomar Conf. on Signals, Systems, and Computers*, 1991, vol. 2, pp. 856–860.
213. L. Tong and S. Perreau, "Multichannel blind identification: from subspace to maximum likelihood methods," *Proc. IEEE*, vol. 86, pp. 1951–1968, Oct. 1998.
214. P. P. Vaidyanathan, *Multirate Systems and Filter Bank*. Englewood Cliffs, NJ: Prentice-Hall, 1993.
215. D. Van Compernolle, "Switching adaptive filters for enhancing noisy and reverberant speech from microphone array recordings," in *Proc. IEEE ICASSP*, 1990, pp. 833–836.
216. B. D. Van Veen and K. M. Buckley, "Beamforming: a versatile approach to spatial filtering," *IEEE Acoust., Speech, Signal Process. Mag.*, vol. 5, pp. 4–24, Apr. 1988.
217. H. L. Van Trees, *Optimum Array Processing. Part IV of Detection, Estimation, and Modulation Theory*. New York: John Wiley & Sons, Inc., 2002.
218. P. Vary and R. Martin, *Digital Speech Transmission: Enhancement, Coding and Error Concealment*. Chichester, England: John Wiley & Sons Ltd, 2006.
219. A. M. Vural, "A comparative performance study of adaptive array processors," in *Proc. IEEE ICASSP*, 1977, vol. 1, pp. 695–700.
220. A. M. Vural, "Effects of perturbations on the performance of optimum/adaptive arrays," *IEEE Trans. Aerospace, Electronic Systems*, vol. AES-15, pp. 76–87, Jan. 1979.
221. C. Wang and M. S. Brandstein, "A hybrid real-time face tracking system," in *Proc. IEEE ICASSP*, 1998, vol. 6, pp. 3737–3741.
222. H. Wang and P. Chu, "Voice source localization for automatic camera pointing system in videoconferencing," in *Proc. IEEE WASPAA*, 1997.
223. D. B. Ward and G. W. Elko, "Mixed nearfield/farfield beamforming: a new technique for speech acquisition in a reverberant environment," in *Proc. IEEE WASPAA*, 1997.

224. D. B. Ward, R. C. Williamson, and R. A. Kennedy, "Broadband microphone arrays for speech acquisition," *Acoustics Australia*, vol. 26, pp. 17–20, Apr. 1998.
225. W. C. Ward, G. W. Elko, R. A. Kubli, and W. C. McDougald, "The new varechoic chamber at AT&T Bell Labs," in *Proc. Wallance Clement Sabine Centennial Symposium*, 1994, pp. 343–346.
226. M. Wax and T. Kailath, "Optimum localization of multiple sources by passive arrays," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-31, pp. 1210–1218, Oct. 1983.
227. M. Wax and T. Kailath, "Detection of signals by information theoretic criteria," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-33, pp. 387–392, Apr. 1985.
228. M. R. Weiss, E. Aschkenasy, and T. W. Parsons, "Processing speech signals to attenuate interference," in *Proc. IEEE Symposium on Speech Recognition*, 1974, pp. 292–295.
229. S. Weiss, G. W. Rice, and R. W. Stewart, "Multichannel equalization in subbands," in *Proc. IEEE WASPAA*, 1999, pp. 203–206.
230. S. Werner, J. A. Apolinário, Jr., and M. L. R. de Campos, "On the equivalence of RLS implementations of LCMV and GSC processors," *IEEE Signal Process. Lett.*, vol. 10, pp. 356–359, Dec. 2003.
231. B. Widrow, J. R. Glover, J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeidler, E. Dong, and R. C. Goodlin, "Adaptive noise cancelling: principles and applications," *Proc. IEEE*, vol. 63, pp. 1692–1716, Dec. 1975.
232. B. Widrow and S. D. Stearns, *Adaptive Signal Processing*. Englewood Cliffs, NJ: Prentice-Hall, 1985.
233. N. Wiener and E. Hopf, "On a class of singular integral equations," *Proc. Prussian Acad., Math.-Phys. Ser.*, p. 696, 1931.
234. N. Wiener, *Extrapolation, Interpolation and Smoothing of Stationary Time Series*. New York: John Wiley & Sons, 1949.
235. D. B. Williams and D. H. Johnson, "Using the sphericity test for source detection with narrow band passive arrays," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, pp. 2008–2014, Nov. 1990.
236. D. B. Williams, "Counting the degrees of freedom when using AIC and MDL to detect signals," *IEEE Trans. Signal Process.*, vol. 42, pp. 3282–3284, Nov. 1994.
237. K. M. Wong, Q.-T. Zhang, J. P. Reilly, and P. C. Yip, "On information theoretic criteria for determining the number of signals in high resolution array processing," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. 38, pp. 1959–1971, Nov. 1990.
238. G. Xu, H. Liu, L. Tong, and T. Kailath, "A least-squares approach to blind channel identification," *IEEE Trans. Signal Process.*, vol. 43, pp. 2982–2993, Dec. 1995.
239. W. Xu and M. Kaveh, "Analysis of the performance and sensitivity of eigendecomposition-based detectors," *IEEE Trans. Signal Process.*, vol. 43, pp. 1413–1426, June 1995.
240. K. Yamada, J. Wang, and F. Itakura, "Recovering of broad band reverberant speech signal by sub-band MINT method," in *Proc. IEEE ICASSP*, 1991, pp. 969–972.
241. C. L. Zahm, "Effects of errors in the direction of incidence on the performance of an adaptice array," *Proc. IEEE*, vol. 60, pp. 1008–1009, Aug. 1972.

242. R. Zelinski, "A microphone array with adaptive post-filtering for noise reduction in reverberant rooms," in *Proc. IEEE ICASSP*, 1988, pp. 2578–2581.
243. X. Zhang and J. H. L. Hansen, "CSA-BF: a constrained switched adaptive beamformer for speech enhancement and recognition in real car environments," *IEEE Trans. Speech Audio Process.*, vol. 11, pp. 733–745, Nov. 2003.

Index